*Федеральное государственное бюджетное образовательное учреждение*

*высшего профессионального образования*

|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | ***«Московский государственный технический университет  имени Н.Э. Баумана»***  ***(МГТУ им. Н.Э. Баумана)*** |

ФАКУЛЬТЕТ Информатика и системы управления

КАФЕДРА Программное обеспечение ЭВМ и информационные технологии

**РАСЧЁТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

**к курсовому проекту на тему:**

Разработка редактора композиций трехмерных графических

примитивов

Студент **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** Луговой Д.М.

(Подпись, дата) (И.О.Фамилия)

Руководитель курсового проекта **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** Майков К.А.

(Подпись, дата) (И.О.Фамилия)

Москва, 2019

Оглавление

[Введение 4](#_Toc25830676)

[1. Аналитический раздел 5](#_Toc25830677)

[1.1 Алгоритмы удаления невидимых линий и поверхностей 5](#_Toc25830678)

[Алгоритм Робертса 5](#_Toc25830679)

[Алгоритм Варнока 6](#_Toc25830680)

[Алгоритм, использующий z-буфер 7](#_Toc25830681)

[Алгоритм трассировки лучей 8](#_Toc25830682)

[Критерии сравнения 8](#_Toc25830683)

[Выбор оптимального алгоритма 9](#_Toc25830684)

[1.2 Алгоритмы закраски 10](#_Toc25830685)

[Простая закраска 10](#_Toc25830686)

[Закраска по Гуро 10](#_Toc25830687)

[Закраска по Фонгу 10](#_Toc25830688)

[Выбор алгоритма закраски 11](#_Toc25830689)

[2. Конструкторский раздел 12](#_Toc25830690)

[2.1 Аффинные преобразования 13](#_Toc25830691)

[2.2 Камера и перспективная проекция 13](#_Toc25830692)

[2.3 Отбрасывание невидимых граней 13](#_Toc25830693)

[2.4 Отсечение по пирамиде видимости 13](#_Toc25830694)

[2. 5 Алгоритм Z-буфера 13](#_Toc25830695)

[2.6 Модель освещения 13](#_Toc25830696)

[Модель Ламберта 13](#_Toc25830697)

[Модель Фонга 14](#_Toc25830698)

[2.7 Построение теней 16](#_Toc25830699)

# Введение

В настоящее время компьютерная графика используется во многих сферах деятельности. Одной из основных областей ее применения является 3D-моделирование, так как построение трехмерного изображения, близкого к реальному, является важной составляющей многих программных продуктов.

Целью данного курсового проекта является разработка редактора композиций, состоящих из графических примитивов, геометрические и спектральные характеристики которых задает пользователь. Для формирования более полного представления о полученной сцене должны присутствовать возможности передвижения камеры и изменения пространственных и спектральных характеристик источника освещения.

В рамках реализации проекта должны быть решены следующие задачи:

* Изучение и анализ существующих алгоритмов компьютерной графики, использующихся для создания реалистичной модели взаимно перекрывающихся объектов, и выбор наиболее подходящего для решения поставленной задачи.
* Проектирование архитектуры программы и ее интерфейса.
* Реализация выбранных алгоритмов и структур данных.
* Проведение исследования на основе разработанной программы.

# 1. Аналитический раздел

## 1.1 Алгоритмы удаления невидимых линий и поверхностей

Фундаментальной задачей при построении реалистичного изображения является задача удаления объектов или их частей, которые перекрываются другими объектами, то есть являются невидимыми с точки зрения наблюдателя. Выделяют две группы алгоритмов для ее решения:

* Алгоритмы, работающие в объектном пространстве. Данные алгоритмы имеют привязку к мировой или физической системе координат. Получаемые результаты ограничиваются только точностью вычислений, однако требуют большого объема вычислений, зависящего от требуемой точности и сложности поступающей на вход сцены. В эту группу входят алгоритм Робертса, алгоритм со списком приоритетов и т.д.
* Алгоритмы, работающие в пространстве изображения. Данные алгоритмы предполагают привязку к системе координат экрана или картинной плоскости, на которую производится проецирование изображаемых объектов. Объем требуемых вычислений значительно меньше, чем у алгоритмов первой группы, и зависит от разрешающей способности экрана и количества объектов на сцене. Основными представителями данной группы являются алгоритм Варнока, алгоритм Z-буфера и алгоритм трассировки лучей.

Для выбора наиболее оптимального алгоритма из перечисленных необходимо осуществить их краткий обзор, отобрать критерии для сравнения и выявить алгоритм, который удовлетворяет всем или большинству критериев.

### Алгоритм Робертса

Алгоритм Робертса представляет собой первое известное решение задачи об удалении невидимых линий. В соответствии с алгоритмом, вначале из каждого тела удаляются те ребра или грани, которые перекрываются самим телом. Затем каждое из видимых ребер каждого тела сравнивается с каждым из оставшихся тел для определения того, какая его часть перекрывается этими телами.

Преимущество данного алгоритма в том, что математические методы, используемые в нем просты, мощны и точны.

Недостатком этого алгоритма является большая, по сравнению с алгоритмами, работающими в пространстве изображения, трудоемкость, которая пропорциональна квадрату количества объектов на сцене.

### Алгоритм Варнока

Основной идеей данного алгоритма является принцип "разделяй и властвуй", заключающийся в разбиении области рисунка на более мелкие подобласти. Для каждой подобласти определяются связанные с ней многоугольники и те из них, видимость которых тривиальна, изображаются на экране. В случае невозможности однозначно определить видимость части многоугольника разбиение области повторяется, и для каждой из вновь полученных подобластей рекурсивно применяется процедура определения видимости. Предполагается, что с уменьшением размеров области количество перекрывающих ее многоугольников сокращается. В результате получаются области, содержащие не более одного многоугольника, либо разбиение продолжается до тех пор, пока размер области не станет равен одному пикселю. В этом случае для полученного пикселя вычисляется значение глубины каждого многоугольника (координата Z), и визуализируется тот из них, у которого значение этой координаты больше.

Достоинством данного алгоритма является простота реализации и высокая эффективность в случае, если размеры перекрываемых областей невелики.

Его недостатком является использование рекурсивных вызовов, что значительно снижает скорость выполнения в случае больших размеров перекрываемых областей.

### Алгоритм, использующий z-буфер

Данный алгоритм удаления невидимых поверхностей является одним из самых простых и широко используемых. Его идея заключается в использовании двух буферов: буфера кадра и буфера глубины, также называемого Z-буфером. Буфер кадра используется для хранения спектральных характеристик каждого пикселя в пространстве изображения. В буфере глубины запоминается значение координаты Z (глубины) каждого видимого пикселя в пространстве изображения. В процессе работы значение глубины каждого нового пикселя, заносимого в буфер кадра, сравнивается с глубиной того пикселя, который уже занесен в Z-буфер. Если это сравнение показывает, что новый пиксель расположен ближе к наблюдателю, чем пиксель, уже находящийся в буфере кадра, то новый пиксель заносится в буфер кадра и производится корректировка Z-буфера: в него заносится глубина нового пикселя. Если же значение глубины нового пикселя меньше, чем хранящееся в буфере, то осуществляется переход к следующей точке.

Основными достоинствами данного алгоритма являются простота его реализации, корректная обработка случаев взаимных пересечений объектов, линейная зависимость трудоемкости от числа объектов на сцене, а также отсутствие необходимости предварительной сортировки объектов по глубине, то есть они могут обрабатываться в произвольном порядке.

К недостаткам данного алгоритма относят необходимость выделения памяти под два буфера, каждый из которых имеет размер равный количеству пикселей на экране.

### Алгоритм трассировки лучей

В этом методе для каждого пикселя картинной плоскости определяется ближайшая к нему грань, для чего через этот пиксель выпускается луч, находятся все его пересечения с гранями и среди них выбирается ближайшая.

К достоинствам данного алгоритма можно отнести возможность получения изображения гладких объектов без аппроксимации их примитивами (например, треугольниками). Вычислительная сложность метода линейно зависит от сложности сцены. Благодаря отслеживанию пути, пройденного лучом, появляется возможность реализовать глобальную модель освещения, учитывающую отражения и преломления света. Качество полученного изображения получается очень реалистичным, этот метод отлично подходит для создания фотореалистичных картин.

Главным недостатком алгоритма трассировки является необходимость создавать огромное число лучей, проходящих через сцену, которые могут раздваиваться на отраженный и преломленный лучи. Это приводит к существенному снижению скорости работы программы.

### Критерии сравнения

При реализации редактора композиций из графических примитивов необходимо обеспечить плавную смену кадров при перемещении камеры и изменении объектов, поэтому основными критериями сравнения алгоритмов являются зависимость трудоемкости алгоритма от числа объектов на экране и использование рекурсивных вызовов.

Так как в наше время среднестатистический компьютер имеет 8 ГБ оперативной памяти [1], то факт задействования дополнительной памяти под буферы не является весомым критерием и может быть опущен.

В связи с ограниченностью времени выполнения курсового проекта значимым критерием отбора алгоритма является простота его реализации, а также то, насколько широко используется алгоритм в современных приложениях.

### Выбор оптимального алгоритма

По критерию использования рекурсивных вызовов оптимальными алгоритмами являются алгоритмы Робертса и Z-буфера, однако алгоритм Робертса имеет квадратичную зависимость от числа объектов на сцене, в то время как алгоритм Z-буфера – линейную.

Алгоритмы Варнока и Z-буфера имеют наименьшую сложность реализации, так как работают непосредственно с пикселями экрана и не требуют решения задачи нахождения пересечения граней объектов.

Наибольшее распространение в современном программном обеспечении получили алгоритмы трассировки лучей и Z-буфера: алгоритм Z-буфера используется в большинстве графических движков, в то время как алгоритм трассировки лучей используется в ПО для создания фотореалистичных сцен.

Как видно из таблицы 1.1, алгоритмом, соответствующим всем выделенным критериям, является алгоритм Z-буфера. Поэтому именно этот алгоритм предлагается для реализации в рамках данного курсового проекта.

Таблица 1 – Сравнение алгоритмов удаления невидимых линий и поверхностей

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Алгоритм | Робертса | Варнока | Z-буфера | Трассировки лучей |
| Использование рекурсивных вызовов | - | + | - | + |
| Зависимость трудоемкости от числа объектов (N) |  | N, но зависит от положения объектов | N | N, но зависит от модели освещения |
| Сложность реализации | Средняя | Низкая | Низкая | Средняя |
| Распространенность в современном ПО | Нет | Нет | Широкая | Широкая |

## 1.2 Алгоритмы закраски

Существуют три основных алгоритма, позволяющих закрасить полигональную модель: простая закраска, закраска по Гуро и закраска по Фонгу.

### Простая закраска

Суть данного алгоритма заключается в том, что для каждой грани объекта находится вектор нормали и с его помощью в соответствии с выбранной моделью освещения вычисляется значение интенсивности, с которой закрашивается вся грань. Данный метод закраски обладает большим быстродействием, однако видны все переходы между гранями.

### Закраска по Гуро

Данный метод принципиально отличается от простой закраски тем, что разные точки грани закрашиваются с разными значениями интенсивности. Для это в каждой вершине грани находится вектор нормали и вычисляется значение интенсивности. Затем найденные значения интенсивности интерполируются по всем точкам грани. С помощью этого метода получаются достаточно реалистичные изображения, однако все объекты кажутся матовыми.

### Закраска по Фонгу

Закраска Фонга по своей идее похожа на закраску Гуро, отличие состоит в том, что в методе Гуро по всем точкам интерполируется значение интенсивности, а в методе Фонга – вектора нормалей, и с их помощью для каждой точки находится значение интенсивности. Эта закраска требует больших вычислительных затрат, однако она позволяет разрешить многие проблемы метода Гуро. Достигается лучшая локальная аппроксимация кривизны поверхности и, следовательно, получается более реалистичное изображение. В частности, правдоподобнее выглядят зеркальные блики.

### Выбор алгоритма закраски

На рисунке 1 наглядно показаны различия рассмотренных методов закраски.



Рисунок 1 – Методы закраски (слева направо: плоская, Гуро, Фонга)

Алгоритм закраски Фонга требует большего числа вычислений по сравнению с другими, однако он дает наиболее реалистичное изображение, в частности зеркальных бликов. В данном курсовом проекте пользователь должен иметь возможность задавать спектральные характеристики объектов, поэтому для получения более реалистичной сцены предлагается использовать метод закраски Фонга.

# 2. Конструкторский раздел

На вход поступает список объектов, которые могут отличаться по геометрическим и спектральным характеристикам. Общий алгоритм программы представляет собой следующую последовательность действий:

1. Для каждого объекта сцены:
   1. Для каждого полигона объекта:
      1. Преобразовать точки полигона из пространства модели в мировое пространство с помощью матриц аффинных преобразований.
      2. Преобразовать точки полигона из пространства модели в пространство камеры с помощью матрицы вида.
      3. Привести все точки к однородным координатам и с помощью матрицы перспективной проекции осуществить переход к пространству отсечения.
      4. Вычислить нормаль полигона, определить видимость полигона с помощью скалярного произведения вектора нормали на вектор, направленный от позиции камеры к полигону.
      5. В случае, если грань лицевая, выполнить отсечение полигона по пирамиде видимости, в противном случае перейти к следующему полигону.
      6. Если после отсечения полигон не был отброшен, то триангулировать его.
      7. Для каждого треугольника полигона:

1.1.7.1) Растеризовать полигон в соответствии с алгоритмом Z-буфера.

2) Вывести заполненный буфер кадра на экран.

## 2.1 Аффинные преобразования

## 2.2 Камера и перспективная проекция

## 2.3 Отбрасывание невидимых граней

## 2.4 Отсечение по пирамиде видимости

## 2. 5 Алгоритм Z-буфера

## 2.6 Модель освещения

Человек видит объекты окружающего мира за счет того, что в соответствии с физическими законами существуют некоторые внешние источники, которые излучают световую энергию, а материальные тела эту энергию поглощают или отражают. Именно отраженная часть света дает представление о цвете объекта. При этом свет может многократно отражаться от различных объектов и проходить сквозь них, если объект частично либо полностью прозрачен. Модель освещения, учитывающая эти факторы, называется глобальной и требует значительных вычислительных затрат.

Существуют более простые, локальные модели освещения, которые учитывают только свет от начального источника. Выделяют две основные модели локального освещения: модель Ламберта и модель Фонга.

### Модель Ламберта

Данная модель является простейшей моделью освещения, так как учитывает только идеальное диффузное отражение света от тела. На рисунке 1 показано, что согласно этой модели, освещенность в точке определяется только плотностью света в точке поверхности, а она линейно зависит от косинуса угла падения. При этом положение наблюдателя не имеет значение, т.к. диффузно отраженный свет рассеивается равномерно по всем направлениям.

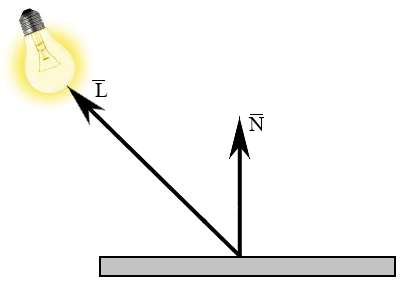


Рисунок 2 – Модель освещения Ламберта

Пусть

– вектор от точки до источника

– вектор нормали

I – результирующая интенсивность света в точке

I0 – интенсивность источника

k – коэффициент диффузного освещения

Формула расчёта интенсивности имеет следующий вид:

(1)

Из формулы (1) следует главный недостаток модели Ламберта – одинаковая интенсивность во всех точках, принадлежащих одной грани.

### Модель Фонга

Модель Фонга является классической моделью локального освещения. Она включает в себя модель Ламберта, но также учитывает тот факт, что кроме диффузного отражения на материале может появляться блик, то есть зеркальное отражение. Местонахождение блика на объекте, освещенном по модели Фонга, определяется из закона равенства углов падения и отражения, как показано на рисунке 2.

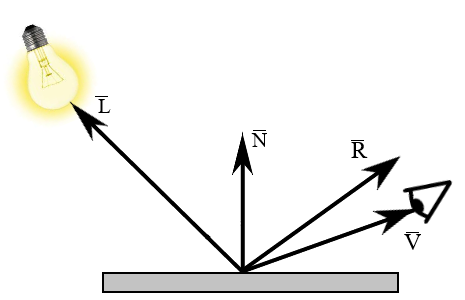


Рисунок 3 – Модель освещения Фонга

Падающий и отраженный лучи лежат в одной плоскости с нормалью к отражающей поверхности в точке падения, и эта нормаль делит угол между лучами на две равные части. Таким образом, отраженная составляющая освещенности в точке зависит от того, насколько близки направления на наблюдателя и отраженного луча. Также в модели освещения Фонга используется понятие рассеянного освещения – это константа, которая прибавляется к интенсивности в точке для придания сцене большего реализма.

Таким образом, согласно модели Фонга интенсивность к точке складывается из 3 компонент: диффузной, зеркальной и рассеянной, как показано на   
рисунке 4.

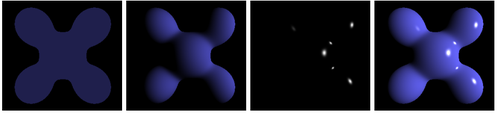


Рисунок 4 – Составляющие модели Фонга (слева направо: рассеянная, диффузная, зеркальная, суммарная)

Пусть

– вектор от точки до источника

– вектор отраженного луча

– вектор нормали

– вектор нормали

I – результирующая интенсивность света в точке

I0 – интенсивность источника

Ip – интенсивность рассеянного освещения

кd – коэффициент диффузного освещения

кз – коэффициент зеркального освещения

ка – коэффициент рассеянного освещения

α - коэффициент блеска

Формула для расчета интенсивности для модели освещения Фонга имеет вид:

(2)

## 2.7 Построение теней

Одним из основных факторов, влияющих на корректное восприятие построенного изображения, является генерация реалистичных теней. С их помощью человек лучше понимает положение различных объектов в пространстве относительно друг друга, у него складывается более полное представление созданной сцены.

Многие приложения не поддерживают динамическое построение теней, либо закрепляя объекты на месте, просчитав тени только при инициализации сцены, либо полностью убирая тени, тем самым создавая сложную для восприятия картинку. Это объясняется сложностью сохранения плавности изображения, так как просчет реалистичных динамических теней требует значительных ресурсов компьютера и сильно снижает кадровую частоту.