

Министерство образования и науки Российской Федерации

федеральное государственное автономное образовательное учреждение
высшего образования

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Институт Кибернетики

Направление подготовки 09.03.01 Информатика и вычислительная техника

Кафедра Автоматики и Компьютерных Систем

БАКАЛАВРСКАЯ РАБОТА

Тема работы

**РАЗРАБОТКА МОДУЛЯ ГЕНЕРАЦИИ И ВИЗУАЛИЗАЦИИ 3D
ЛАНДШАФТА**

УДК 004.925.84:551.4

Студент

Группа	ФИО	Подпись	Дата
3-8В2В3	Шумихин Василий Сергеевич		

Руководитель

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент кафедры АиКС	Скирневский И.П.	аспирант		

КОНСУЛЬТАНТЫ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент кафедры менеджмента	Рахимов Т.Р.	к.э.н., доцент		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Инженер	Маланова Н.В.	к.т.н.		

ДОПУСТИТЬ К ЗАЩИТЕ:

И.о. зав. кафедрой	ФИО	Ученая степень, звание	Подпись	Дата
АиКС	Суходоев М.С.	к.т.н.		

Перечень подлежащих исследованию, проектированию и разработке вопросов	Процедура исследования состоит из следующих этапов: 1) Изучение используемых алгоритмов; 2) Анализ аналогов; 3) Проектирование логики программного модуля; 4) Программная реализация; 5) Составление программной документации;
Перечень графического материала	Скриншоты пользовательского интерфейса редактора, скриншоты из программы.

Консультанты по разделам выпускной квалификационной работы	
Раздел	Консультант
«Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»	Рахимов Тимур Рустамович
«Социальная ответственность»	Маланова Наталья Викторовна

Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику	
---	--

Задание выдал руководитель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент кафедры АиКС	Скирневский Игорь Петрович	аспирант		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
3-8В2В3	Шумихину Василий Сергеевич		

Министерство образования и науки Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»



Студенту:

Группа	ФИО
3-8В2В3	Шумихину Василию Сергеевичу

Институт	Кибернетики	Кафедра	Вычислительной техники
Уровень образования	Бакалавриат	Направление/специальность	09.03.01 Информатика и вычислительная техника

Исходные данные к разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»:

1. <i>Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих</i>	Работа с информацией, представленной в российских и иностранных научных публикациях, аналитических материалах, статистических бюллетенях и изданиях, нормативно-правовых документах
2. <i>Нормы и нормативы расходования ресурсов</i>	
3. <i>Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования</i>	

Перечень вопросов, подлежащих исследованию, проектированию и разработке:

1. <i>Оценка коммерческого потенциала, перспективности и альтернатив проведения НИ с позиции ресурсоэффективности и ресурсосбережения</i>	<ul style="list-style-type: none"> • Определение потенциальных потребителей результатов исследования, сегментирование рынка. • Проведение анализа конкурентов. • SWOT-анализ. • Определение альтернатив проведения исследования
2. <i>Планирование и формирование бюджета научных исследований</i>	<ul style="list-style-type: none"> • Определение этапов исследования • Определение трудоемкости работ • Расчет бюджета исследования для альтернативных вариантов
3. <i>Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования</i>	<ul style="list-style-type: none"> • Определение сравнительной эффективности исследования

Перечень графического материала (с точным указанием обязательных чертежей):

1. <i>Методология QuaD</i>
2. <i>Матрица SWOT</i>
3. <i>Альтернативы проведения НИ</i>
4. <i>График проведения и бюджет НИ</i>
5. <i>Оценка ресурсной, финансовой и экономической эффективности НИ</i>

Дата выдачи задания для раздела по линейному графику

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент кафедры менеджмента	Рахимов Т. Р.	к.э.н., доцент		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
3-8В2В3	Шумихин Василий Сергеевич		

Министерство образования и науки Российской Федерации
 Федеральное государственное автономное образовательное учреждение
 высшего образования



**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
 ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Институт	Институт дистанционного обучения
Направление подготовки (специальность)	Информатика и вычислительная техника
Уровень образования	Бакалавриат
Кафедра	Автоматики и компьютерных систем
Период выполнения	(осенний / весенний семестр 2016/2017 учебного года)

Студенту:

Группа	ФИО
3-8В2В3	Шумихину Василию Сергеевичу

Тема работы:

Утверждена приказом проректора-директора (директора) (дата, номер)	

Форма представления работы:

Бакалаврская работа

(бакалаврская работа, дипломный проект/работа, магистерская диссертация)

ЗАДАНИЕ

Исходные данные к разделу «Социальная ответственность»:	
Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области его применения.	Алгоритм генерации и визуализации 3D ландшафта.
Перечень вопросов, подлежащих исследованию, проектированию и разработке:	

<p>1. Производственная безопасность</p> <p>1.1. Анализ вредных и опасных факторов, которые может создать объект исследования.</p> <p>1.2. Анализ вредных и опасных факторов, которые могут возникнуть на рабочем месте при проведении исследований.</p> <p>1.3. Обоснование мероприятий по защите исследователя от действия опасных и вредных факторов.</p> <p>2. Экологическая безопасность</p> <p>2.1. Анализ влияния объекта исследования на окружающую среду.</p> <p>2.2. Анализ влияния процесса исследования на окружающую среду.</p> <p>2.3. Обоснование мероприятий по защите окружающей среды.</p> <p>3. Безопасность в чрезвычайных ситуациях</p> <p>3.1. Анализ вероятных ЧС, которые может инициировать объект исследований.</p> <p>3.2. Анализ вероятных ЧС, которые могут возникнуть на рабочем месте при проведении исследований.</p> <p>3.3. Обоснование мероприятий по предотвращению ЧС и разработка порядка действия в случае возникновения ЧС.</p> <p>4. Правовые и организационные вопросы обеспечения безопасности</p> <p>4.1. Специальные (характерные для проектируемой рабочей зоны) правовые нормы трудового законодательства.</p> <p>4.2. Организационные мероприятия при компоновке рабочей зоны.</p>	<p>1. Производственная безопасность</p> <p>1.1. Объект исследование не создает вредных и опасных факторов.</p> <p>1.2. Вредные факторы:</p> <ul style="list-style-type: none"> • Микроклимат; • Шум; • Электромагнитные поля; • Освещение. <p>Опасные факторы:</p> <ul style="list-style-type: none"> • Опасность пожара; • Опасность поражения током. <p>2. Экологическая безопасность</p> <p>2.1. Объект исследования не влияет на окружающую среду.</p> <p>2.2. Загрязнение литосферы.</p> <p>3. Безопасность в чрезвычайных ситуациях</p> <p>3.1. Объект исследования не может инициировать ЧС.</p> <p>3.2. Пожарная безопасность.</p>
---	---

Дата выдачи задания для раздела по линейному графику	
---	--

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Инженер	Маланова Н. В.	к.т.н.		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
3-8В2В3	Шумихин Василий Сергеевич		

СОДЕРЖАНИЕ

ЗАПЛАНИРОВАННЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ООП	9
РЕФЕРАТ	10
ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ, СОКРАЩЕНИЯ, НОРМАТИВНЫЕ ССЫЛКИ	12
ВВЕДЕНИЕ	13
1. АНАЛИЗ И ПОСТАНОВКА ЗАДАЧ	15
1.1. Анализ предметной области	15
1.2. Анализ аналогов	16
1.3. Выбор среды разработки и технологии программирования	18
2. ИСПОЛЬЗУЕМЫЕ АЛГОРИТМЫ	20
2.1. Алгоритм моделирования трехмерного ландшафта	20
2.1.1. Входные параметры алгоритма	22
2.1.1.1. Карта высот	22
2.1.1.2. Уровень детализации	23
2.1.2. Выходные данные алгоритма	24
2.1.3. Алгоритм Quadtree	25
2.1.3.1. Ограниченный квадрант	25
2.1.3.2. Поверхность на основе дерева квадрантов	27
2.1.3.3. Оптимизация	29
2.2. Алгоритмы интерполяции	30
2.2.1. Биквадратная интерполяция	30
2.2.2. Бикубическая интерполяция	31
3. АРХИТЕКТУРНОЕ ПРОЕКТИРОВАНИЕ	32
3.1. Функциональные требования	32
3.2. Архитектурные решения	32
3.3. Варианты использования	33
3.4. Диаграмма классов	34
4. ПРОГРАММНАЯ РЕАЛИЗАЦИЯ	35

4.1. Основные определения среды визуализации.....	35
4.1.1. Программирование в Unity3d	35
4.1.2. Порядок выполнения функций событий	35
4.1.3. Компонент физики твердого тела	36
4.1.4. Компоненты для физического взаимодействия.....	36
4.1.5. Игрок	37
4.2. Построение дерева квадрантов.....	38
4.2.1. Импорт карты высот.....	38
4.2.2. Генерация дерева квадрантов.....	38
4.3. Реализация основного функционала.....	39
4.3.1. Функционал скрипта QuadTerrain.....	39
4.3.2. Функционал скрипта QuadManager	41
4.3.3. Движение по поверхности ландшафта	42
5. РЕЗУЛЬТАТЫ РАБОТЫ.....	44
6. ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ	48
7. СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ.....	65
ЗАКЛЮЧЕНИЕ	80
СПИСОК ЛИТЕРАТУРЫ.....	81
ПРИЛОЖЕНИЕ А	83

ЗАПЛАНИРОВАННЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ООП

Код результатов	Результат обучения (выпускник должен быть готов)
<i>Профессиональные компетенции</i>	
P1	Применять базовые и специальные естественнонаучные и математические знания в области информатики и вычислительной техники, достаточные для комплексной инженерной деятельности.
P2	Применять базовые и специальные знания в области современных информационных технологий для решения инженерных задач.
P3	Ставить и решать задачи комплексного анализа, связанные с созданием аппаратно-программных средств информационных и автоматизированных систем, с использованием базовых и специальных знаний, современных аналитических методов и моделей.
P4	Разрабатывать программные и аппаратные средства (системы, устройства, блоки, программы, базы данных и т. п.) в соответствии с техническим заданием и с использованием средств автоматизации проектирования.
P5	Проводить теоретические и экспериментальные исследования, включающие поиск и изучение необходимой научно-технической информации, математическое моделирование, проведение эксперимента, анализ и интерпретация полученных данных, в области создания аппаратных и программных средств информационных и автоматизированных систем.
P6	Внедрять, эксплуатировать и обслуживать современные программно-аппаратные комплексы, обеспечивать их высокую эффективность, соблюдать правила охраны здоровья, безопасность труда, выполнять требования по защите окружающей среды.
<i>Универсальные компетенции</i>	
P7	Использовать базовые и специальные знания в области проектного менеджмента для ведения комплексной инженерной деятельности.
P8	Владеть иностранным языком на уровне, позволяющем работать в иноязычной среде, разрабатывать документацию, презентовать и защищать результаты комплексной инженерной деятельности.
P9	Эффективно работать индивидуально и в качестве члена группы, состоящей из специалистов различных направлений и квалификаций, демонстрировать ответственность за результаты работы и готовность следовать корпоративной культуре организации.
P10	Демонстрировать знания правовых, социальных, экономических и культурных аспектов комплексной инженерной деятельности.
P11	Демонстрировать способность к самостоятельному обучению в течение всей жизни и непрерывному самосовершенствованию в инженерной профессии.

РЕФЕРАТ

Выпускная квалификационная работа состоит из 89 страниц, 22 рисунка, 20 таблиц, 16 источников.

Ключевые слова: трехмерная графика, трехмерное моделирование, ландшафт, визуализация, движок Unity.

Объектом исследования являются алгоритмы генерации и визуализации трехмерного ландшафта.

Цель работы – разработать программный модуль генерации и визуализации трехмерного ландшафта.

В результате исследования был разработан программный модель генерации и визуализации трехмерного ландшафта.

Основные конструктивные, технологические и технико-эксплуатационные характеристики: аппроксимация местности по карте высот, визуализация местности.

Степень внедрения: разработанный программный модуль находится на этапе тестирования.

Область применения: программный модель может использоваться в разработке игровых приложения, приложений симуляции, специализированного программного обеспечения для строителей.

ОПРЕДЕЛЕНИЯ И СОКРАЩЕНИЯ

Нормативные акты

ГОСТ 19.401 – 78 Единая система программной документации. Текст программы. Требования к содержанию и оформлению.

ГОСТ 19.402 – 78 Единая система программной документации. Описание программы.

ГОСТ 19.404 – 79 Единая система программной документации. Пояснительная записка.

ГОСТ 19.502 – 78 Единая система программной документации. Описание применения. Требования к описанию и оформлению.

ГОСТ 19.504 – 79 Единая система программной документации. Руководство программиста. Требования к содержанию и оформлению.

ГОСТ 19.101 – 77 Единая система программной документации. Виды программ и программных документов.

Определения

3D-моделирование (трехмерное моделирование) – это процесс создания трехмерной модели объекта. Задача 3D-моделирования – разработать визуальный объемный образ желаемого объекта.

Ландшафт – это достаточно большой, обозримый простым глазом участок поверхности, отличающийся от соседних участков характерными индивидуальными чертами.

Программное обеспечение (ПО) – это все или часть программ, процедур, правил и соответствующей документации системы обработки информации.

Рендеринг (визуализация) – это термин в компьютерной графике, обозначающий процесс получения изображения по модели с помощью компьютерной программы.

Игровой движок (game engine) – это центральный программный компонент интерактивных приложений с графикой, обрабатываемой в

реальном времени. Как правило, основными модулями игрового движка являются среда визуализации (graphics engine, графический движок) и движок физики (physics engine), реализующие, соответственно, графическое отображение приложения и физическое взаимодействие объектов в приложении.

ВВЕДЕНИЕ

В настоящее время компьютерная графика используется во многих областях, таких как наука, медицина, архитектура, производство кино, реклама и компьютерные игры. Конструкторы, разрабатывая новые модели автомобилей и самолетов, используют трехмерные графические объекты, чтобы представить окончательный вид изделия. Архитекторы создают на экране монитора объемное изображение здания, и это позволяет им увидеть, как оно впишется в ландшафт.

Один из трудоемких процессов – это 3D-моделирование. Область 3D-моделирования можно разделить на две подобласти. Первая это генерация трехмерных изображений в реальном времени, а вторая – генерация высоко реалистичных трехмерных изображений. Высокореалистичные изображения требуют очень больших объемов вычислений, в то время как в первой подобласти применяется система условных допущений, позволяющая применять более простые алгоритмы, и тем самым значительно уменьшить объем вычислений, что, однако, сказывается на качестве изображения.

В некоторых задачах решают проблему построения реалистичных ландшафтов. Особенно это касается специализированного ПО для симуляции, строительства и игр жанров RPG (role-play game) и MMORPG (massively multiplayer online role-playing game). Срок разработки такого ПО может занимать от 2 до 6 лет. Более того стоимость разработки подобных проектов очень высока. Один из способов удешевить стоимость изготовления – это автоматизировать некоторые процессы.

Ландшафты можно создавать вручную, но этот процесс занимает достаточное продолжительное время. Можно автоматизировать процесс создания грубой модели ландшафта с последующей обработкой.

Данная работа посвящена разработке модуля генерации и визуализации 3D ландшафта. Разработка модуля происходит в рамках этапа разработки программного обеспечения для интерактивного редактирования местности.

Создание модуля подразумевает решение следующих задач:

- 1) Провести анализ существующих программных продуктов, используемых в данной сфере;
- 2) Провести анализ возможностей выбранной среды визуализации;
- 3) Разработать архитектуру программного модуля;
- 4) Реализовать генерацию и визуализацию, что подразумевает:
 - a) Добавление исходных данных о ландшафте;
 - b) Аппроксимация исходных данных для получения модели;
 - c) Визуализация полученной модели местности.
- 5) Создать рабочий прототип приложения для демонстрации результатов.

В первом разделе работы производится анализ и постановка задачи. В разделе рассматриваются вопросы актуальности выбранной тематики и практическая значимость проекта, выбора среды разработки, модели представления. В заключение раздела обозначается цель и ставятся задачи выполняемой работы.

Во втором разделе описываются используемые алгоритмы, их применение и входные параметры.

Третий раздел посвящен архитектурному проектированию. Приводятся функциональные требования, варианты использования и диаграмма классов.

Четвертый раздел раскрывает подробности программной реализации генерации и визуализации ландшафта.

В пятом разделе показаны результаты работы.

Шестой раздел посвящен финансовым расчетам затрат на ведение проектной деятельности, а также рассмотрению альтернативных путей планирования процесса разработки.

Седьмой раздел описывает факторы производственной среды, влияющие на процесс разработки программного обеспечения при выполнении данной работы.

1. АНАЛИЗ И ПОСТАНОВКА ЗАДАЧИ

1.1. Анализ предметной области

В последние десятилетия наблюдается повышенный интерес к рендерингу в реальном времени и моделированию поверхностей. В частности, на сегодняшний день остается актуальной задача интерактивной визуализации реалистичного ландшафта. Под понятием реалистичный ландшафт подразумевается, то, что ландшафт, полученный каким-то не естественным способом, будет выглядеть похоже по ряду признаков на естественный ландшафт. К таким признакам относят:

- гладкий переход между поверхностями не подверженные эрозии;
- эрозия влияет на рельеф с точки зрения воздействия воды на природный материал;
- однородный рельеф;
- единый геологический фундамент;
- иметь уникальную структуру.

Другими словами, полученная территория должна представлять собой не делимое по внешним признакам территорию с особыми факторами развития.

Интерактивная визуализация масштабного рельефа имеет ряд проблем с эффективностью. Чтобы наилучшим образом использовать производительность рендеринга, необходимо максимально уменьшить сложность сцены, не приводя к ухудшению визуального представления. Следовательно, для контроля за нагрузкой вводится количественный параметр, с помощью которого можно оценить сложность сцены – такой параметр называется уровень детализации (level-of-detail, LOD). Согласно LOD можно осуществляется извлечение треугольных сеток в основную память с интерактивной частотой кадров, а загрузка новых данных по требованию. Кроме того, различные уровни рельефа можно отнести к разным уровням детализации. Чтобы уменьшить сложность рендеринга без негативного

влияния на визуальную точность, триангуляция ландшафта должна быть адаптивной к характеристикам поверхности местности: распределять все меньше и больше треугольники для областей с гладкой поверхностью и использовать все больше и меньше треугольники для сложного рельефа местности. Кроме того, модель триангуляции может обеспечивать средства для быстрого извлечения представлений поверхности с переменной точностью, так что несколько LOD могут поддерживаться во время выполнения.

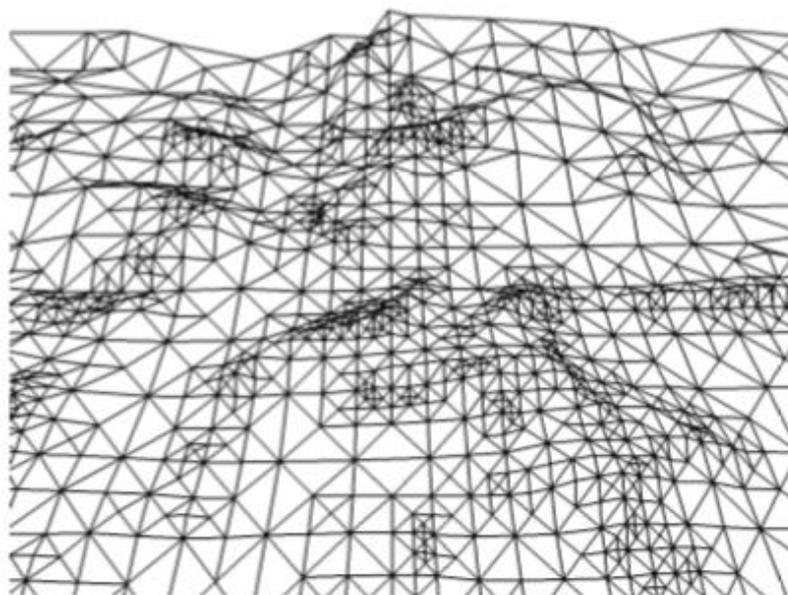


Рисунок 1 – Пример адаптивного ландшафта

1.2. Анализ аналогов

Существующие пакеты генерации ландшафта подразделяются на генераторы и системы моделирования, в состав которых включены генераторы. Нужно отметить, что на сегодняшний день системы второго типа более распространены. Оба типа систем не доступны большинству разработчиков и исследователей в виду либо высокой стоимости, либо неэффективности системы. Малая эффективность связана с тем, что результат работы алгоритмов существующих пакетов генерации мало предсказуемы, а потому для получения итоговой модели, удовлетворяющей заданным условиям, генерацию необходимо повторять многократно, а сгенерированный ландшафт подвергается трудоемкой дизайнерской постобработке.

На рынке представлены готовые решения, такие как Vue Esprit и Terragen.

Vue Esprit – это профессиональная программа для моделирования окружающей среды, ориентированный на 3D-художников. Есть возможность экспортировать в различные 3D-форматы, такие как 3ds, lwo, obj, dxf и как рендер в форматы tiff, tga, pict, bmp, gif, jpeg, png. Эта программа для ландшафтного моделирование и анимации имеет следующие достоинства:

- импорт карты высот;
- малый объем дистрибутива;
- моделирование неба и облаков.
- Так же она имеет ряд недостатков:
- не предусмотрена возможность повтора или отмены действий;
- невозможно создать растительность;
- невозможность импорта 3D-моделей;
- невозможность построения ландшафта в режиме реального времени;
- невозможность использования API программы для встраивая в другие ПО.

Распространяется по цене 349 долларов США за базовый набор и за 699 за полную комплектацию.

Terragen – программа ландшафтного моделирования и анимации, позволяющая создавать фотореалистичные пейзажи. Существуют как бесплатная, так и коммерческая версии программы. Бесплатная версия имеет некоторые ограничения по сравнению с коммерческой.

Программа позволяет генерировать и модифицировать ландшафт как двухмерную карту высот. Ввиду того, что карта высот генерируется случайным образом, получить одинаковые пейзажи почти невозможно.

Достоинства:

- Генерирование фотореалистичных ландшафтов, их импорт и экспорт в Raw-формате.

- Фотореалистичное моделирование водной поверхности.
- Импорт карты высот для моделирования ландшафтов.
- Моделирование неба и облаков.
- Анимация.
- Подключаемые плагины.
- Малый объём дистрибутива.

Недостатки:

- Невозможность импорта 3D-моделей.
- Невозможность импорта текстур.
- Отсутствие возможности создания растительности и ввиду этого некоторая безжизненность ландшафтов.
- Не предусмотрена возможность повтора/отмены действий (Undo/Redo).
- В бесплатной версии невозможно создать изображения с разрешением более 1280×960 и ландшафтов размером более 513×513. Кроме этого недоступно сглаживание (антиалиасинг) изображений, и невозможно установить режим fast/ultra rendering, ускоряющий отрисовку изображения.

Все представленные продукты ориентированы на художников, что не подразумевает работу с моделью ландшафта в режиме реального времени, а только пост- и предобработку модели. Также не имеется возможности перенести функционал в другое ПО.

1.3. Выбор среды разработки и технологии программирования

После анализа существующих программных продуктов, для задачи визуализации было принято решение использовать бесплатную версию игрового движка Unity 5.5. Это решение обусловлено тем, что в дальнейшем разрабатываемый программный модуль предполагается использовать в разработке других программных продуктов на Unity3d и распространять модуль в электронном магазине Unity Store как Unity Asset.

Unity Asset – цифровой объект, преимущественно состоящий из однотипных данных, неделимая сущность, которая представляет часть игрового контента и обладает некоторыми свойствами. С точки зрения программной модели ассет (ресурс) может выступать в виде объекта, созданного на некотором наборе данных. Ресурс может храниться в виде отдельного файла [9].

Данное решение имеет ряд преимуществ:

- Большое распространение использования движка и самое большое сообщество пользователей;
- Готовые инструменты для разработки двух- и трехмерных приложений;
- Кроссплатформенность, поддержка таких платформ как: Windows, Mac OS, Linux, консоли Xbox, PlayStation, устройства виртуальной и дополненной реальности;
- Удобный интерфейс редактора для работы с объектами на трехмерной сцене в режиме реального времени;
- Удобные инструменты отладки;
- Полный функционал платформы .Net 2.0;
- Возможности импорта изображений, 3D-моделей, звуков и видео во многих форматах.

Недостатки решения:

- Привязка к конкретной архитектуре движка, из чего следует невозможность переноса исходного кода;
- Невозможность встраивания библиотек в любой другой проект отличный от Unity;
- Платформа .Net 2.0 на сегодняшний день устарела [8].

2. ИСПОЛЬЗУЕМЫЕ АЛГОРИТМЫ

2.1. Алгоритм моделирования трехмерного ландшафта

Для начала, дадим определение основным понятиям. Выбранная среда разработки и визуализации реализует полигональный подход в трехмерной графике. Любой объект полигональной графики задается с помощью двумерных геометрических примитивов, представленных в виде набора полигонов. Полигон – это плоский многоугольник. Любая полигональная поверхность состоит из конечного множества небольших многоугольников. Соседние многоугольники связаны собой общими ребрами и вершинами. В связи с тем, что многоугольники (обычно треугольники и четырехугольники) или полигоны имеют малые размеры, то при большом количестве они создают иллюзию гладкой трехмерной поверхности.

Используемые полигоны рассматриваются как набор упорядоченных последовательностей вершин. Вершины полигонов определяют точечную сетку, которая задает реберную сетку полигональной модели. Такая модель называется – *каркасной моделью*.

Ребро – это сторона многоугольника, которая определяется двумя вершинами, следующими друг за другом. Кроме того, последняя вершина многоугольника соединена с первой, что замыкает многоугольник. Ребро рассматривается как отрезок, соединяющий две вершины. В полигональной модели ребра служат для соединения между собой двух граней. При этом, ребро, принадлежащее только одной грани, называется *граничным*. Граничные ребра могут использоваться для соединения разных полигональных моделей.

Область, ограниченную последовательность ребер, называется грань. Любой полигональный объект рассматривается как набор граней. Совокупность вершин, ребер и граней называется полигональной сеткой (polygon mesh, меш).

Изменение порядка вершин грани не влияет ни на форму граней, ни на последовательность ребер граней. Однако, при этом изменяется направление, в котором ориентированная грань. То есть каждая грань имеет «лицевую» и

«изнаночную» стороны. Положение лицевой стороны грани определяется вектором, который называют нормалью. Обычно нормаль – это вектор, исходящий из центра грани в направлении лицевой стороны перпендикулярно к поверхности. Как правило для упрощения рендеринга рисуется только лицевая сторона грани, определяемая по нормали.

Грань может содержать локальную координатную сетку (с переменными U и V), используемую при позиционировании накладываемых текстур. В общем случае, наличие такой сетки необязательно.

Полигональная сетка обычно состоит из многоугольников. Но лишь о треугольнике можно сказать, что он является плоским, поскольку известно, что через любые три точки всегда можно провести только одну плоскость. Таким образом, любой полигон в виде треугольника является плоской фигурой. Сказать то же самое о прочих многоугольниках нельзя. Даже четырехугольник, который первоначально был плоским, может изогнуться в процессе деформации объекта и превратиться в криволинейную поверхность. Использование же кривых поверхностей полигонов зачастую неприемлемо. Такой четырехугольник необходимо либо заранее разделить на две треугольные грани, либо принять специальные меры, исключающие появление неплоских полигонов.

В настоящее время треугольная полигональная сетка (*триангулярная сетка*) является наиболее предпочтительным представлением полигонального объекта. А процесс разбиение граней полигонального объекта на треугольники называется триангуляцией.

Достоинства полигонального подхода являются:

- Относительная легкость выполнения расчетов, так как каждый многоугольник либо представляет собой плоскую фигуру, либо его можно представить как набор плоских фигур;

- В области, требующие большей детализации, можно просто вставить дополнительные полигоны;

- Имеется возможность работать с отдельно взятыми областями, объединив их в дальнейшем в единое целое;
- Полигоны легче покрывать текстурой, присваивать им цвета или свойства материалов;
- Сплавление точек для соединения полигональных поверхностей дает незаметные соединения.

Однако, данный подход не лишен недостатков:

- Высокие погрешности интерполяции при использовании полигонов больших размеров;
- При сложной структуре описания объектов возникают проблемы установки приоритетов, определения и смены уровней детализации, описания и обработки многоугольников;
- Увеличение количества полигонов, придавая большую реалистичность визуализируемым трехмерным объектам, в то же время приводят к повышению требования по производительности используемых вычислительных средств. Что в свою очередь приводит к необходимости создания сложных параллельно-конвейерных структур с системой распараллеливания и синхронизации множества процессоров.

Для системы визуализации, отображающей многоугольники, повышение качества изображения эквивалентно повышению производительности системы, измеряемой в количестве обрабатываемых полигонов за время формирования одного кадра. Поэтому реалистичность изображения непосредственно зависит от сложности сцены и точности приближения [6, 7].

2.1.1. Входные параметры алгоритма

2.1.1.1. Карта высот

Самым важным входным параметром являются высоты ландшафта. Двухмерная карта, используемая для хранения высот, называется картой высот (heightmap) [7].

Обычно карта хранится в виде 8-разрядного изображения с градацией серого, где каждая точка изображения хранит высоту ландшафта в соответствующей позиции.

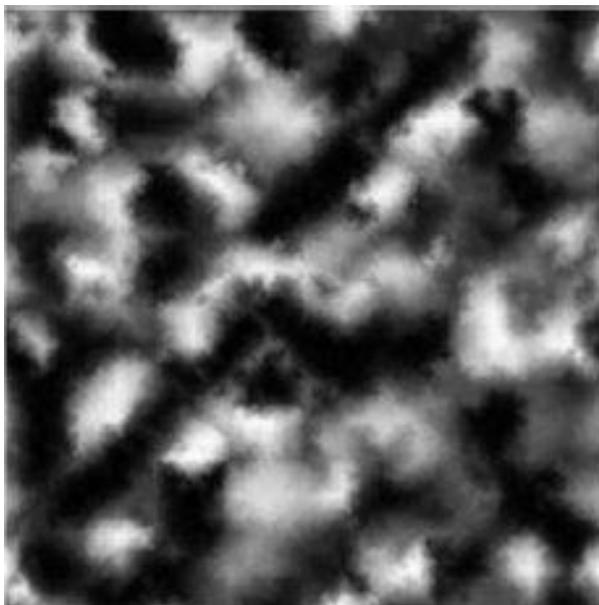


Рисунок 2 – Пример карты высот

Как правило для хранения карты высот используют необработанный формат файлов RAW, который содержит только информацию о цвете пикселей и никаких информационных заголовков. Однако, Unity позволяет представить изображение почти любого формата в виде карты высот, например, bmp или jpeg.

Изображение переводится в двумерный массив высот, где значение высоты – значение любого из цветового канала соответствующего пикселя. В дальнейшем полученный массив высот передается алгоритму аппроксимации.

2.1.1.2. Уровень детализации

Уровень детализации (level-of-detail, LOD) – это количественный коэффициент, введенный для заданий качества аппроксимации карты высот в виде полигональной сетки.

Для данного алгоритма величина уровня детализации является обратно пропорциональна количеству позиций, которые будут драться за одну вершину. Если LOD меньше единицы, тогда на одну вершину берется

несколько позиций карты высот, при этом высота вершины интерполируется по высотам занимаемых позиций. Аналогично для LOD большего единицы.

2.1.2. Выходные данные алгоритма

Выходными данными алгоритма является объект типа Mesh, который представляет из себя треугольную полигональную сетку (меш). Данный объект является входным данным для компонента визуализатора MeshFilter.

Компоненты (Component) – это функциональные части каждого объекта на сцене. Объекты сцены (игровые объекты, GameObject) являются контейнерами для различных компонентов.

Компонент меш фильтр (Mesh Filter) принимает объект типа Mesh и передаёт его компоненту Mesh Renderer для последующей визуализации меша на экране [9].

За более подробной информацией о Mesh можно обратиться в полное руководство по Unity3d. Для данной работы необходимы только три его свойства: vertices, triangles, uvs.

Свойство vertices является одномерным массивом типа данных Vector3, который описывает точку в трехмерном пространстве. В массиве хранятся вершины полигональной сетки. Средой разработки предусмотрено ограничение массива в 65535 элементов. При превышении ограничения вызывается исключение, которое прекращает дальнейшую работу объектом.

Свойство triangles является одномерным массивом целочисленных данных. В массиве хранятся индексы вершин из массива vertices. При этом каждая последовательная тройка индексов интерпретируется визуализатором как треугольник меша. При этом от последовательность извлекаемых вершин зависит вектор нормали рисуемой грани. Если вектора, образуемые 1-й и 2-й вершинами, 1-й и 3-й вершинами и их векторным произведением, являются левой тройкой векторов, то вектор нормали положительный, т.е. имеет положительную y-координату. Иначе нормаль отрицательная.

Свойство uvs является одномерным массивом типа данных Vector2, который описывает точку в двухмерном пространстве. Массив хранит

координаты для текстурированные грани. Координаты добавляются для каждой вершины из поля vertices. Данное поле не является обязательным и имеет значение по умолчанию для каждой вершины.

2.1.3. Алгоритм Quadtree

Наиболее распространёнными и эффективными структурами для аппроксимации указанного в п. 2.1.1 набора входных данных являются иерархические структуры данных, такие как квадродерева (дерево квадратов, quadtree). Соответственно, для иерархических структур данных используются рекурсивные алгоритмы. Рассмотрим алгоритм подробнее.

2.1.3.1. Ограниченный квадрант

Ограниченный квадрант является узлом дерева квадратов. Квадрант представляет собой четырехугольник, вершины которого находятся на высотах аппроксимируемой области карты высот.

В пространстве параметров дерево квадратов строится рекурсивно, пока для каждой области дискретизации не выполняется условие Липшица для параметрической кривой, которая ограничивает точность получающегося полигонального приближения. Кроме того, разбиение дерева ограничено, так что соседние области должны находиться на одном уровне иерархии (Рис. 3).

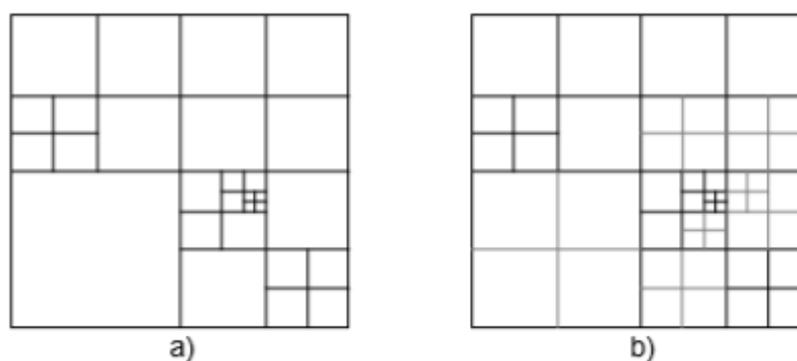


Рисунок 3 – Пример неограниченного разбиения на квадранты a) и ограниченного разбиения b)

Для триангуляции и визуализации выполняют следующие шаги:

- 1) Первоначальная отбор на однородной сетке.
- 2) Оценка каждого региона по некоторым критериям приемлемости (оценка ошибки аппроксимации).

3) Разбиение региона на 4 части. Повторение шагов 2 и 3 до тех пор, пока адаптивная выборка не удовлетворяет критериям приемлемости по всей поверхности.

4) Триангуляция и визуализация всей области квадродерева.

Чтобы предотвратить возможные «трещины» в полигональном представлении (Рис. 3), каждая область четырехугольника триангулируется относительно размера соседних областей. Из-за ограничения иерархии, что уровень детализации соседних областей отличается не более чем на 1, «трещины» не могут появиться, как на (Рис. 4). Такая триангуляция называется согласованная [5].

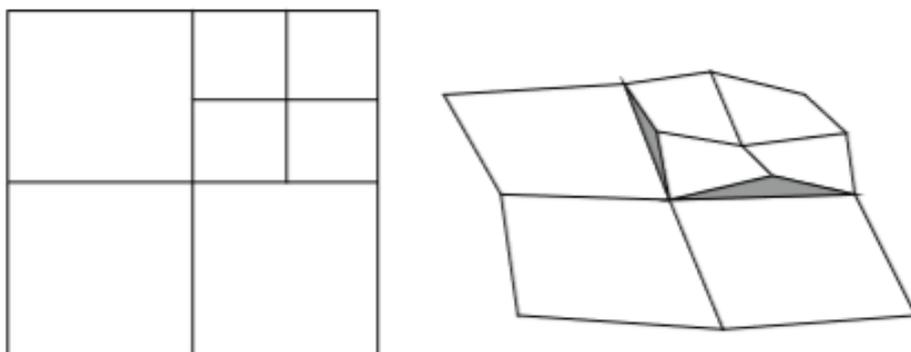


Рисунок 4 – Пример "трещин"

Правило триангуляции заключается в следующем: каждый квадрант разбивается на 8 треугольников, по 2 треугольника на ребро, кроме случая, когда соседний квадрант большего размера, тогда 1 треугольник (Рис. 5).

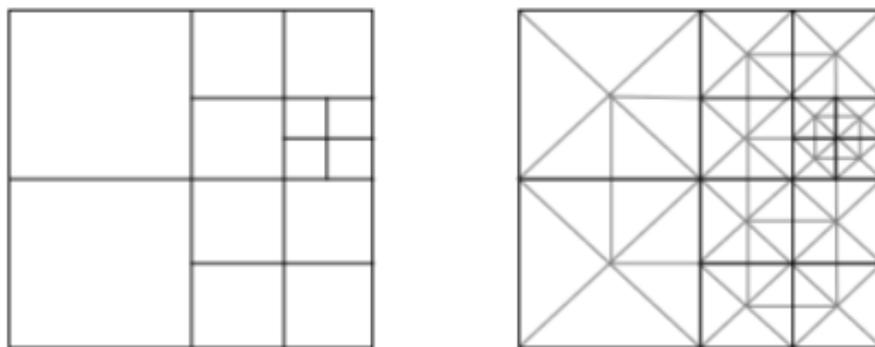


Рисунок 5 – Пример триангуляции

Набор треугольников формируется путем рекурсивного обхода квадродерева. Обход осуществляется вокруг центральной вершины квадранта против часовой стрелки за один проход (Рис. 6).

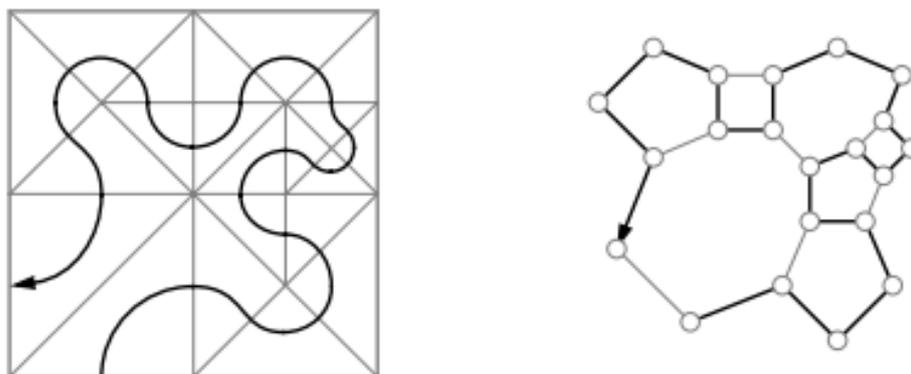


Рисунок 6 – Пример обхода квадродерева.

2.1.3.2. Поверхность на основе дерева квадрантов

Существует два алгоритма построения поверхности: восходящий (снизу-вверх) и нисходящий (сверху-вниз).

В методе восходящего построения входная сетка разбивается на атомарные узлы точек (3x3). Эти узлы образуют листовые узлы полного и сбалансированного квадранта по всей входной сетке.

Основная фаза этого метода состоит в том, чтобы скомбинировать все сжимаемые узлы снизу-вверх, чтобы создать иерархию ограниченных квадрантов. Узлы должны пройти два основных критерия:

1). *Критерий ошибки* – ошибка аппроксимации, возникающая при удалении не угловых вершин соединяемых узлов, должна быть не больше допустимого порога ошибки.

2). *Ограничение квадродерева* – размер узла равен размеру трех его братьев в иерархии, и ни узел, ни его братья не имеют соседей меньшего размера.

Ошибка аппроксимации представляет собой вертикальное расстояние удаленных вершин относительно их линейной интерполяции, обеспечиваемой родительским узлом (Рис. 7). Ошибка вершины В – это её вертикальное расстояние до среднего значения вершин А и С. Алгоритм завершается, если не может быть выполнено больше слияний, и он имеет линейную оценку $O(n)$, где n – размер входного набора данных.

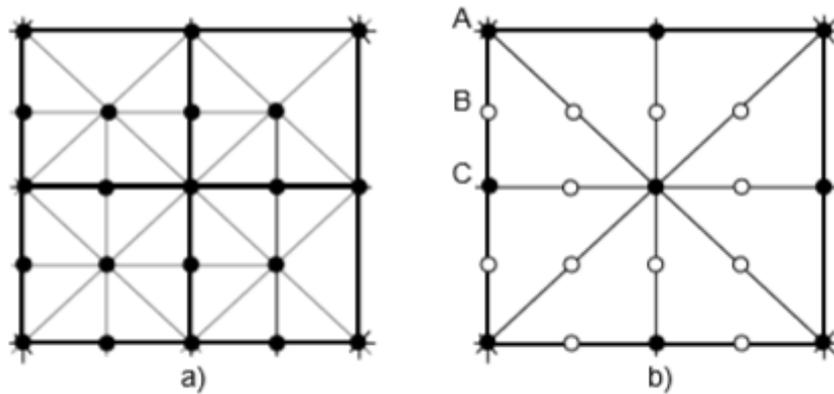


Рисунок 7 – Пример слияния узлов

Основная проблема восходящего метода является вычисление ошибки аппроксимации. Хотя вертикально расстояние от удаляемой вершины до линейной интерполяции соседних вершин находится в пределах порога ошибки ϵ , однако, не гарантируется что удаленная вершина находится в пределах ϵ расстояния до конечного результата итеративного восходящего слияния (Рис. 8). Фактически результирующая упрощенная поверхность может не интерполировать удаленные вершины, следовательно, упрощенная поверхность может не обладать требуемой точности аппроксимации.



Рисунок 8 – Ошибка аппроксимации при слиянии узлов

Второй метод, нисходящий, начинается с представления всего входного набора данных одним корневым узлом, который рекурсивно разбивается на узлы, никогда не сливая, что позволяет аппроксимировать весь набор данных. Метод все время сохраняет свойство дерева квадрантов, что соседние узлы отличаются не более чем на 1 уровень иерархии. Однако, чтобы сохранять это свойство возможна ситуация, что добавление новых вершин при

разбиении приведет к переразбиению соседних узлов. При этом переразбиение может распространяться как в вертикально по иерархии дерева, так и горизонтально по соседним квадрантам [1, 2, 3].

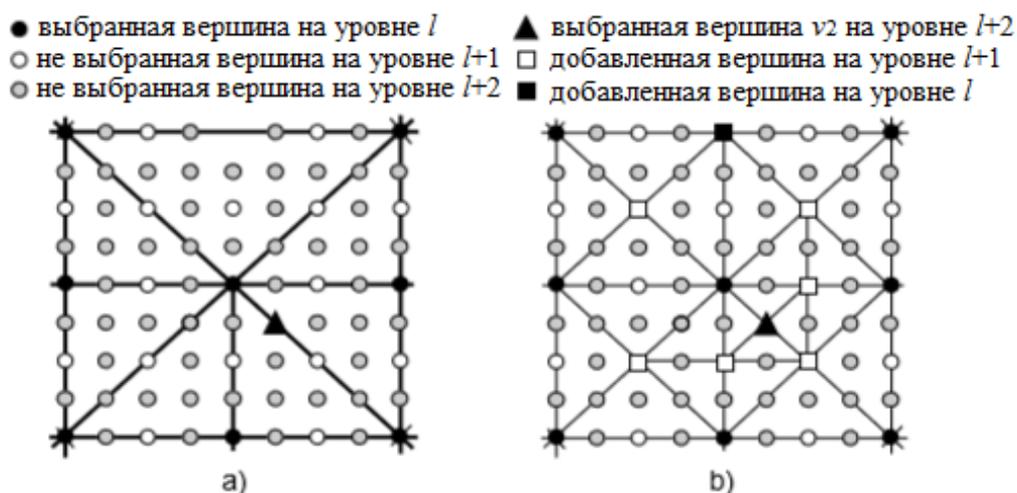


Рисунок 9 – Распространение переразбиения

Например, как показано на Рис. 9, при добавлении вершины V_2 , текущее разбиение на уровне $l + 1$ модифицируется, и модификация распространяется, а затем происходит разбиение на уровне $l + 2$.

Хотя такой метод чувствителен к изменениям входных данных, в тоже время, метод гарантирует, что ошибка аппроксимации будет находиться в пределах заданного порога ϵ . Метод также имеет линейную оценку $O(n)$.

В данной работе используется второй метод (нисходящий), потому что несмотря на риск возрастания вычислительных затрат он гарантирует точность аппроксимации.

2.1.3.3. Оптимизация

Использование иерархических структур сопряжено с возможностью чрезмерного разрастания иерархического дерева, что потребляет лишние ресурсы. Оптимизация выполняется для уменьшения затрат таких важных ресурсов как память (на хранение дерева) и вычислительные ресурсы процессора (на обход дерева).

Во-первых, вводится понятие *области визуализации*, т.е. прямоугольной области на карте высот или текстуре, которую следует

визуализировать. Это делается потому что нет необходимости визуализировать все входные данные сразу, т.к. это может занять слишком много времени и ресурсов. При движении камеры по сцене область визуализации двигается за ней. Следовательно, незачем обновлять все квадродерево и меш. Необходимо только добавить новые данные из области визуализации и удалить участки, которые уже не входят в область.

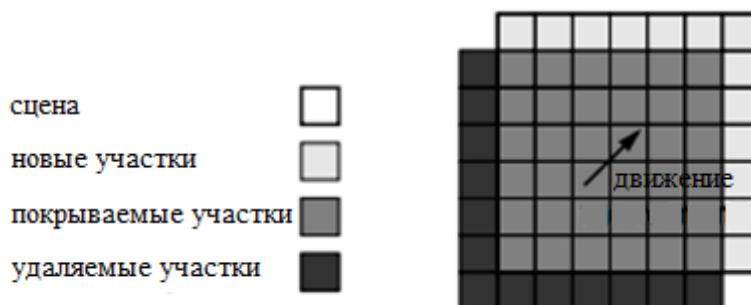


Рисунок 10 – Движение области визуализации

Во-вторых, в области визуализации нет необходимости поддерживать постоянно один LOD, ведь самое важное только то что находится в непосредственной близости от камеры. Следовательно, объекты более от камеры можно сделать менее детальными. Для это необходимо только перейти по иерархии квадродерева на уровень выше [3, 4].

2.2. Алгоритмы интерполяции

Интерполяция (уточнение) – это способ нахождения приближенного значения неизвестной величины, находящейся в окружении точек с известной величиной.

В работе алгоритма интерполяция может применяться на этапе нахождения высоты вершин граней полигональной сетки или сглаживания. В работе используются два алгоритма интерполяции: билинейная и бикубическая.

2.2.1. Билинейная интерполяция

Является расширением линейной интерполяции для функций двух переменных. Основная идея алгоритма заключается в том, чтобы провести обычную линейную интерполяцию сначала в одном направлении, затем в

перпендикулярном. Билинейной интерполяцией интерполирует значения функции в четырехугольнике по четырем её значениям в вершинах прямоугольника. Если для интерполяции выбирается больше четырех значений, то множество вершин разбивается на четыре более простых подмножества и интерполяция выполняется рекурсивно в них.

2.2.2. Бикубическая интерполяция

Является расширением кубической интерполяции на случай функции двух переменных. Поверхность, полученная в результате бикубической интерполяции, является гладкой функцией (непрерывной), в отличие от поверхностей, полученных в результате билинейной интерполяции. В случае бикубической интерполяции значение функции в искомой точке вычисляется через её значения в 16 соседних точках.

Может применяться как основной алгоритм интерполяции или для больших значений LOD, т.к. получаем гладкую и более реалистичную поверхность. Однако, алгоритм обладает большой вычислительной сложностью, что может существенно повлиять на производительность всего программного модуля.

3. АРХИТЕКТУРНОЕ ПРОЕКТИРОВАНИЕ

3.1. Функциональные требования

К разрабатываемому модулю предъявляются следующие требования:

- Должна быть возможность импорта карты высот;
- Модуль должен аппроксимировать ландшафт по карте высот;
- Должна быть возможность хранить структуру аппроксимированного ландшафта;
- Должна быть возможность создать трехмерную сцену, где визуализируется аппроксимированный ландшафт;
- Должна быть возможность регулировать степень детализации визуализированного ландшафта;
- Должна быть возможность визуализировать аппроксимированный ландшафт.

3.2. Архитектурные решения

Основой алгоритма является построение дерева квадрантов, его обход и модификация.

Процесс построения может быть весьма ресурсоемким как по ресурсам вычислительного устройства (размер памяти, вычислительные ресурсы), так и по затраченному времени. Не посредственно повлиять можно на затрачиваемое время. Оно зависит от размеров карты высот, значения LOD и размеров визуализируемой области. Кроме того, основной функционал не сможет работать без построенного дерева квадрантов, а будет простаивать в ожидании. Гораздо практичнее передавать в основную программу уже созданное квадродерево, потому что весь функционал может сразу начинать работу. Отсюда следует решение, что дерево квадрантов с заранее заданным LOD следует строить до начала работы основного приложения. Лучше всего будет аппроксимировать карту высот на этапе импорта в редактор Unity, а построенное дерево сохранить в файл, из которого его можно загружать в память в программе.

Ограничение на количество вершин при создании объекта типа данных Mesh не позволяет создавать большие участки ландшафта с высокой степенью детализации. Поэтому, разделим общее пространство на области фиксированного размера, которые в дальнейшем будет называться *террейн* (*terrain, местность*). Каждый террейн аппроксимируется поддеревом основного дерева квадрантов, при этом LOD поддерева не может быть больше LOD основного дерева. Обращение к основному дереву происходит лишь при крайней необходимости, например, при отсутствии данных о террейне. Такое решение позволяет:

- Нет необходимости держать все дерево квадрантов в памяти, достаточно загрузить лишь необходимые поддеревья;
- Позволяет вынести многие операции с поддеревьями в отдельные потоки, чтобы не занимать основной поток;
- Для уменьшения или увеличения детализации террейна достаточно перейти на другой уровень в иерархии поддерева.

Однако, решение имеет и ряд недостатков:

- Программная реализация усложняется из-за необходимости многопоточности и синхронизации потоков;
- Копии поддеревьев целесообразнее хранить в отдельных от основного дерева и загружать их из файлов, т.к. каждый поток будет обращаться к одному и тому же файлу с основным деревом, что будет задерживать остальные потоки.

3.3. Варианты использования

Диаграмма вариантов использования описывает взаимоотношения и зависимости между группами вариантов использования и действующими лицами, участвующими в процессе. Важно понимать, что диаграммы вариантов использования не предназначены для отображения проекта и не могут описывать внутреннее устройство системы.

Для разрабатываемого модуля диаграмма приведена на рис. 11.

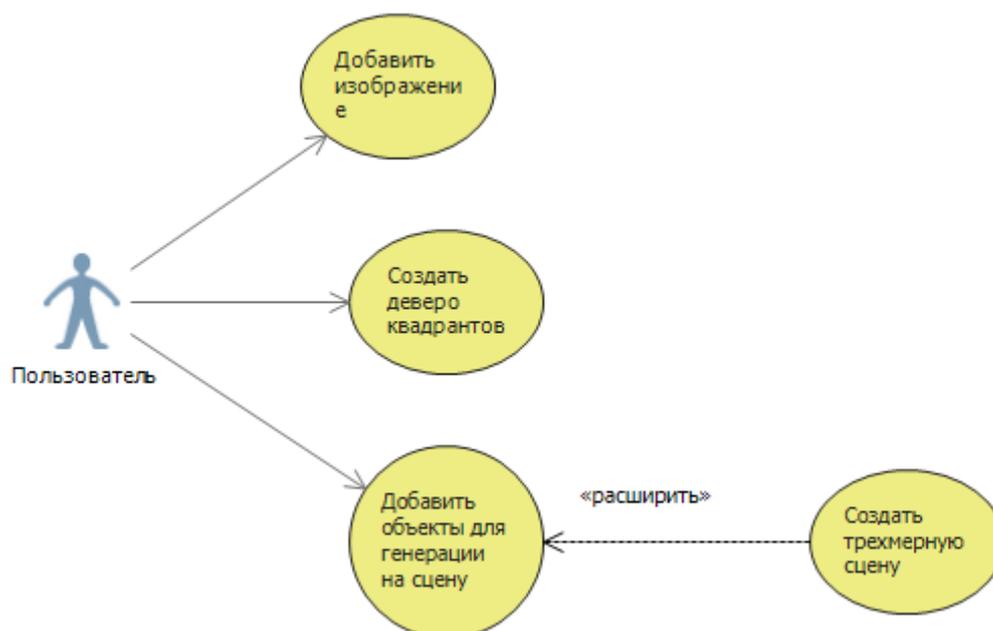


Рисунок 11 – Диаграмма вариантов использования

3.4. Диаграмма классов

Для описания внутреннего устройства проекта используются диаграммы классов. Диаграммы классов являются одной из форм статического описания системы с точки зрения ее проектирования, показывая ее структуру. Диаграмма классов не отображает динамическое поведение объектов, изображенных на ней классов. На диаграммах классов показываются классы, интерфейсы и отношения между ними.

Диаграмма классов разрабатываемого модуля приведена в Приложении А.

4. ПРОГРАММНАЯ РЕАЛИЗАЦИЯ

4.1. Основные определения среды визуализации

4.1.1. Программирование в Unity3d

В Unity классы программы, которая будет исполняться оформляются в виде скриптов (сценариев). Каждый скрипт обязательно наследуется от класса `MonoBehaviour`, в котором реализован функционал движка. Каждый скрипт является компонентом игрового объекта (объекта, который присутствует на сцене), который реализует функционал или часть функционала этого объекта. Все скрипты автоматически подписаны на основные события, которые генерирует движок (например, обновление кадра), программист может лишь реализовать свой обработчик события. Все обработчики событий движка имеют строго заданные имена и сигнатуры, например, `void Start()`. Все обработчики событий выполняются в основном потоке исполняемой программы [9].

4.1.2. Порядок выполнения функций событий

Как упоминалось в п. 4.1.1, в скриптинге Unity есть некоторое количество функций событий, которые исполняются в заранее заданном порядке по мере исполнения скрипта. Порядок исполнения описан ниже:

- **Awake**: Эта функция всегда вызывается до любых функций `Start` и также после того, как объект был создан на сцене (если `GameObject` неактивен на момент старта, `Awake` не будет вызван, пока `GameObject` не будет активирован, или функция в каком-нибудь прикрепленном скрипте не вызовет `Awake`).
- **Start**: Функция `Start` вызывается до обновления первого кадра (`first frame`) только если скрипт включен.
- **FixedUpdate**: Зачастую случается, что `FixedUpdate` вызывается чаще чем `Update`. `FU` может быть вызван несколько раз за кадр, если частота кадров (FPS) низок и функция может быть и вовсе не вызвана между кадрами, если FPS высок. Все физические вычисления и обновления происходят сразу после

FixedUpdate. FixedUpdate вызывается в соответствии с фиксированным таймером, независимым от частоты кадров.

– **Update:** Update вызывается раз за кадр. Это главная функция для обновлений кадров [9].

4.1.3. Компонент физики твердого тела

Rigidbody (твердое тело) позволяют GameObject действовать под контролем физического движка. Наибольшее отличие между управлением трансформациями и твёрдыми телами заключается в использовании силы (force). Твёрдые тела могут управляться силами и вращением, трансформации же не могут. Трансформации можно перемещать и вращать, но это не то же самое, что и использование физики. Добавление силы / вращения к твёрдому телу позволяет изменить позицию и вращение компонента трансформаций (Transform) объекта. Вот почему нужно использовать только один из них. Изменение трансформаций при использовании физики может создать событие столкновения.

Компонент Rigidbody должен быть явно добавлен в GameObject, прежде чем объект будет затронут движком физики. Он использует под гравитацию и может получать вектор приложения силы через скрипты, но может потребоваться добавить коллайдер, чтобы заставить его вести себя именно так, как необходимо [9].

4.1.4. Компоненты для физического взаимодействия

Компонент collider (коллайдер) определяет форму объекта для физических столкновений. Коллайдер, который невидим, не обязательно должен быть такой же формы, как и сетка объекта, и на самом деле грубое приближение часто более эффективно и неразлично в игровом процессе.

Простейшими коллайдерами являются так называемые примитивные типы коллайдеров. В 3D это коллайдер Box, Sphere Collider и Capsule Collider. В 2D это Box Collider 2D и Circle Collider 2D. Любое количество коллайдеров может быть добавлено к одному объекту для создания составных коллайдеров.

При тщательном позиционировании и калибровке составные коллайдеры часто могут довольно хорошо аппроксимировать форму объекта, сохраняя при этом низкую нагрузку на процессор. Однако, есть некоторые случаи, когда даже составные коллайдеры недостаточно точны. В 3D можно использовать Mesh Colliders, который точно соответствует форме полигональной сетки объекта. Однако, эти коллайдеры гораздо более интенсивно нагружают процессор, чем примитивные типы. Кроме того, сетчатый коллайдер обычно не может столкнуться с другим сетчатым коллайдером (т. е. ничего не произойдет, когда они пересекутся). Это можно обойти, отметив в инспекторе редактора коллайдер сетки как. Как правило, хорошим тоном является использование коллайдеров сетки для геометрии сцены и приближение формы движущихся объектов с использованием составных примитивных коллайдеров.

Коллайдеры могут быть добавлены к объекту без компонента Rigidbody. Они называются статическими коллайдерами. Коллайдеры на объекте, у которого есть Rigidbody, называются динамическими коллайдерами. Статические коллайдеры могут взаимодействовать с динамическими коллайдерами, но поскольку у них нет Rigidbody, они не будут двигаться в ответ на столкновения.

Система скриптов может обнаруживать, когда происходят столкновения, и обработчик события OnCollisionEnter. Однако физический движок можно использовать просто для обнаружения, когда один коллайдер входит в пространство другого, не создавая столкновения. Коллайдер, настроенный как триггер (с использованием свойства Trigger), не ведет себя как сплошной объект и просто позволяет другим коллайдерам проходить сквозь себя. Когда коллайдер входит в свое пространство, триггер будет вызывать обработчик OnTriggerEnter на скриптах объекта триггера [9].

4.1.5. Игрок

Игроком считается GameObject, который помечен тэгом «Player» и к которому добавлен компонент камера. Камера – это компонент, который

отображает сцену на экран. Для физического взаимодействия в объект игрока добавлен Capsule Collider.

4.2. Построения дерева квадрантов

4.2.1. Импорт карты высот

Карта высот представлена в виде изображения типа bmp.

Импорт карты высот производится из редактора Unity. Для этого достаточно просто перенести изображение в папку Assets редактора. Или щелкнуть правой кнопкой мыши по папке Assets в редакторе, в появившемся меню выбрать Import Asset и в появившемся диалоговом окне выбрать файл с изображением.

4.2.2. Генерация дерева квадрантов

Генерация дерева происходит в редакторе. API редактора Unity позволяет добавлять новые элементы интерфейса и выполнять программы прямо в редакторе.

В редакторе были добавлены поле типа Texture2D, которое принимает на вход изображение, LOD, максимальное количество террейнов (MaxTerrainCount) и кнопку для запуска программы генерации (Рис. 12).

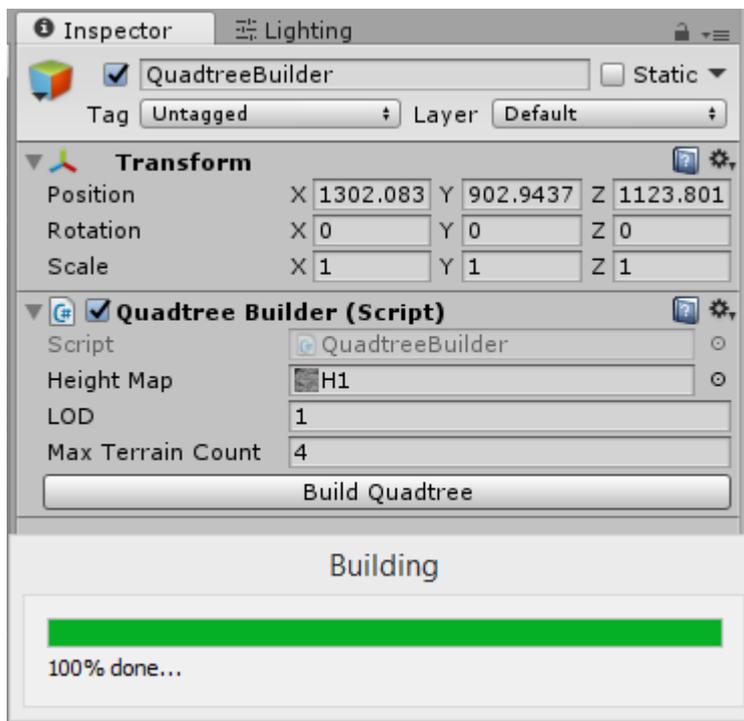


Рисунок 12 – Интерфейс скрипта генерации дерева квадрантов

Процесс генерации квадродерева происходит в потоке параллельном потоку редактора, информация о прогрессе выводится в элемент ProgressBar. Результат работы программы сохраняется в файлы типа .json и располагаются в папке Resources. Папка Resources предназначена для контента, который предполагается использовать (динамически подгружать или выгружать из памяти) по ходу выполнения основной программы.

Как уже упоминалось раньше, LOD это величина обратно пропорциональная количеству пикселей карты высот, принимаемых за одну вершину. Высота вершины интерполируется алгоритмами, описанными в п. 2.2.

Решения о разделении всей аппроксимируемой области на части (террейны), количество и размеры частей принимается на основании размеров карты высот и LOD – чем выше значение LOD и больше карта, тем на большее количество частей делится вся область. Однако, если количество частей превысит MaxTerrainCount, то работа программы завершится и в консоль редактора выведется сообщение об ошибке.

4.3. Реализация основного функционала

Основной функционал разделен между двумя скриптами: скрипт террейна и скрипт объекта-менеджера, который реализует создание/удаление объектов террейнов.

4.3.1. Функционал скрипта QuadTerrain

Скрипт террейна реализует создание меша и коллайдера, и их обновление.

Входным доступными параметрами скрипта являются: LOD террейна, коэффициент расстояния между точками в горизонтальной плоскости (xPerUnit и zPerUnit), коэффициент для высоты (HeightPerUnit) и коэффициент перепада высот (HeightBetweenUnits). Также террейну можно передать материал, который наложится на меш.

После создания объекта террейна выполняется обработчик события Start(), в котором выполняется создание меша. Для заполнения созданного

меша выполняется триангуляция квадрантов согласно LOD, после чего вершины и набор индексов вершин передаются в меш. Для операции обновления также выполняется триангуляция, но результаты передаются в уже существующий меш. В обоих случаях триангуляции выполняется асинхронно основному потоку программы. Пример террейна представлен на Рис. 13.

Коэффициенты `xPerUnit` и `zPerUnit` – это масштабирующие коэффициенты, которые являются расстоянием между соседними точками по осям X и Z соответственно.

Во время триангуляции может возникнуть следующая ситуация: граница между двумя террейнами с разными LOD проходит через местность с резкими переходами высот, например, через гористую местность – тогда возможно образование «трещин». Для предотвращения подобной ситуации при триангуляции террейн с меньшим LOD проводит на границе проверки на частые и большие перепады высот. Порогом перепада высот двух соседних точек является коэффициент `HeightBetweenUnits`. В местах, где перепад высот выше `HeightBetweenUnits`, проводится более детальная триангуляция, с переходом на уровень ниже в иерархии квадродерева террейна.

Объект террейна содержит триггер, который вызывает событие пересечения своего коллайдера игроком. Событие передается объекту-менеджеру.

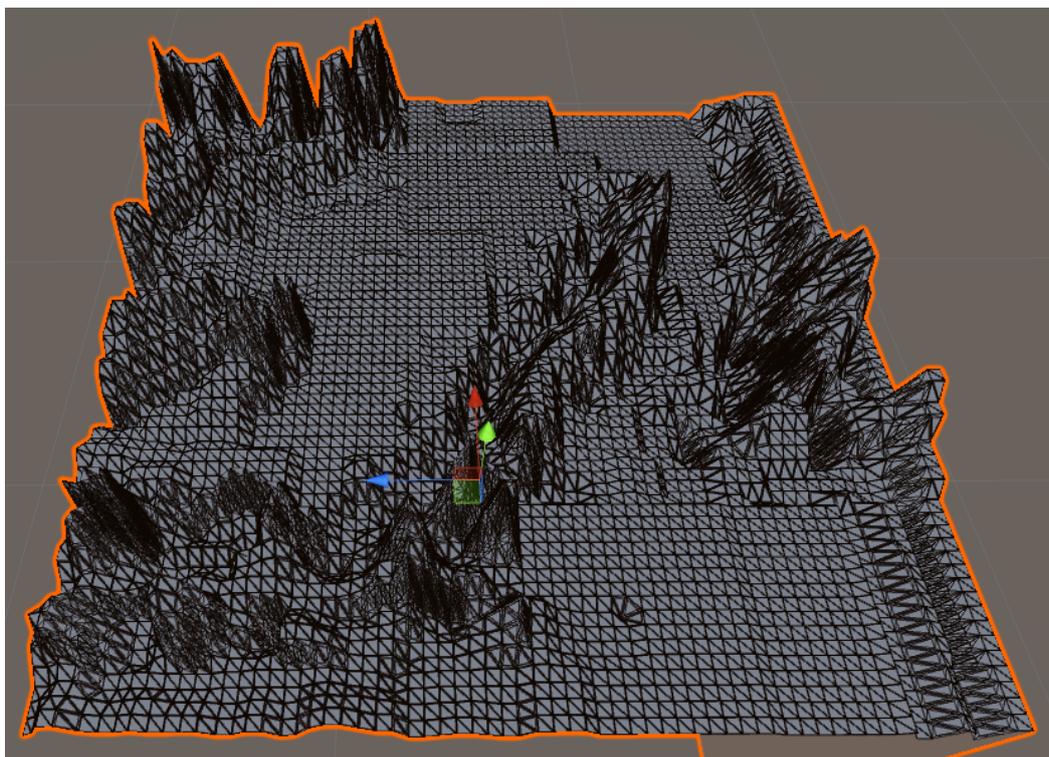


Рисунок 13 – Пример террейна на сцене редактора

4.3.2. Функционал скрипта QuadManager

Основные функции менеджера – мониторинг позиции игрока, создание и удаление террейнов, обновление террейнов.

Менеджер отвечает за обновление информации об область визуализации (VisualizationArea), создание и удаление террейнов и обновление террейнов в области визуализации.

В области визуализации содержится информация о террейне, на котором в данный момент находится игрок, состав соседних террейнов, количество террейнов в области, массив LOD для террейнов в области (при этом, чем дальше террейн от игрока тем больше индекс в массиве LOD).

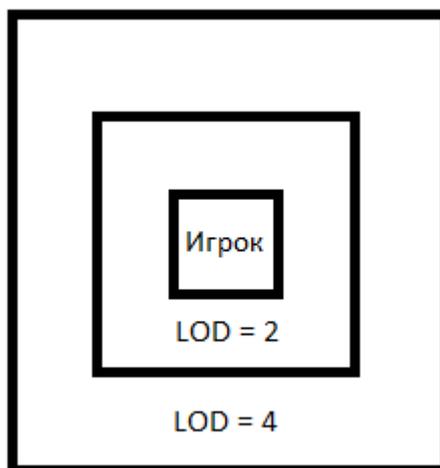


Рисунок 14 – Пример области визуализации

При пересечении триггера какого-либо террейна игроком, у всех террейнов в области визуализации вызывается обновление. Если террейн находится за пределами области визуализации, то объект террейна удаляется со сцены и из области визуализации. В случае, когда в области визуализации не хватает террейнов, например, после запуска программы или после удаления террейна, менеджер асинхронной загружает необходимое дерево квадрантов из файла в папке ресурсов, создает новый объект на сцене и добавляет в область визуализации.

4.3.3. Движение по поверхности ландшафта

Для передвижения игрока по сцене объекту-игроку добавляется скрипт, который обрабатывает события нажатия клавиш на клавиатуре и движения мыши, и задает скорость по направлению движения. Клавиши клавиатуры задают направление движения. После задания направления движения, объект передвигается в заданном направлении.

Направление движения задается суммой векторов направления камеры и вектором осей. Вектор направления камеры изменяется движением мыши. Вектор осей – это сумма направлений горизонтальной и вертикальной осей. Направление осей устанавливаю по нажатию заданных в настройках клавиш. По умолчанию для горизонтальной оси следующие значения клавиши и значения: W – 1.0f, S – -1.0f – для вертикальной оси: D – 1.0f, A – -1.0f.

Передвижение происходит при каждом обновлении кадра. Вектор направления умножается на скалярное значение скорости передвижения, и передается компоненту RigidBody как вектор скорости. При таком способе перемещения, посредством физики твердого тела, движок физики рассчитывает все коллизии коллайдера игрока с коллайдерами всех встреченных по направлению вектора скорости объектов. Что предотвращает «просачивание» объекта один в другой, как, возможно, было бы если для перемещения использовалась простая сумма координат игрока с полученным вектором направления движения. Кроме того, при перемещении через присвоение координат происходил бы пересчет физического взаимодействия со всеми окружающими объектами, что добавило бы вычислительной нагрузки при обновлении кадра (при частоте смены кадров 30 раз в секунду) и остается возможность «просачивания» игрока внутрь или за объект.

5. РЕЗУЛЬТАТЫ РАБОТЫ

Результатом работы является программная библиотека для работы в редакторе Unity. С помощью разработанной библиотеки, создадим проект для демонстрации.

Для начала, импортируем карту высот (Рис. 15) размером 1024x1024 пикселей:

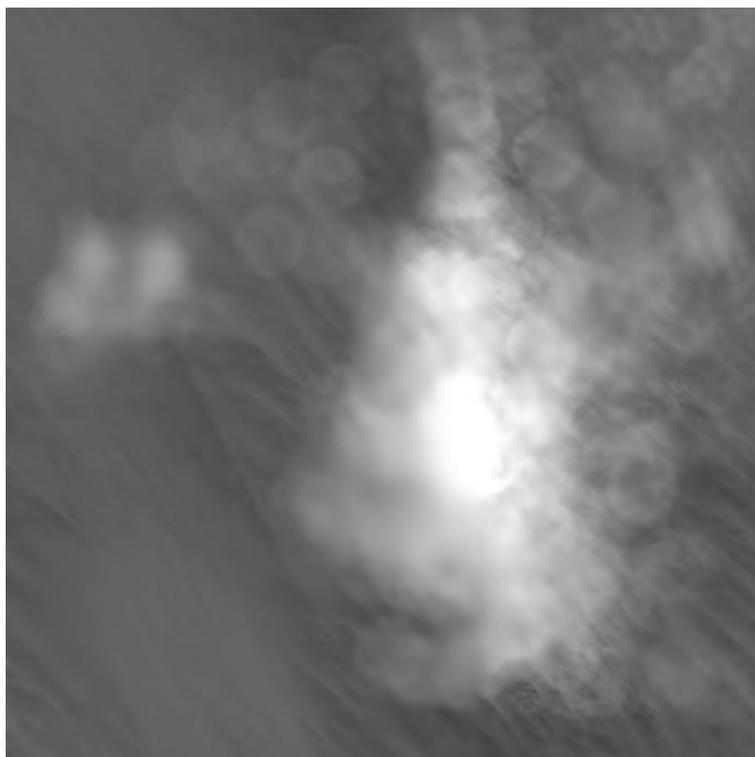


Рисунок 15 – Пример карты высот

Затем запустим построение дерева с $LOD = 0.5$ с размером карты 2048x2048, тем самым получив один файл с деревом квадрантов и четыре файла с поддеревьями (Рис. 16):

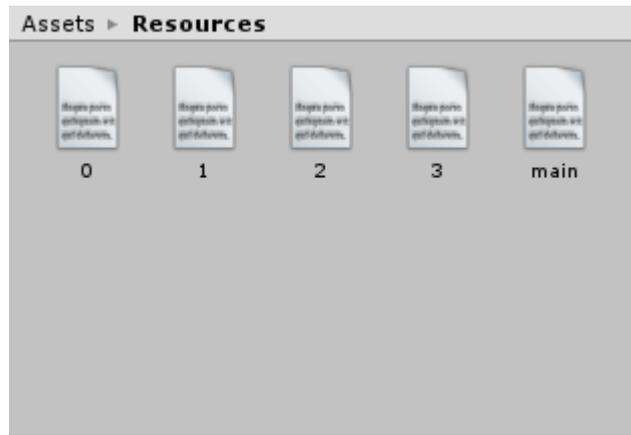


Рисунок 16 – Пример файлов в папке Resources

Теперь, когда дерево построено, создадим новую сцену, поместив на неё источник постоянного света (Direction Light), объект игрока с камерой и объект со скриптом QuadManager (Рис. 17). Запустим программу в редакторе нажав кнопку .

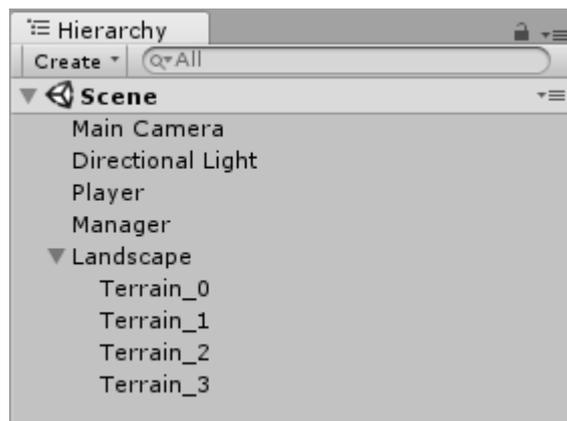


Рисунок 17 – Окно иерархии объектов на сцене

В окне редактора имеется возможность переключения режима отображения в режим отображения полигональной сетки и режим наложения материала. Для наглядности на Рис. 18 показан результат визуализации в виде полигональной сетки, а на Рис. 19 результат с наложением материала на грани меша. При этом пусть максимальный LOD террейна будет 1.0.

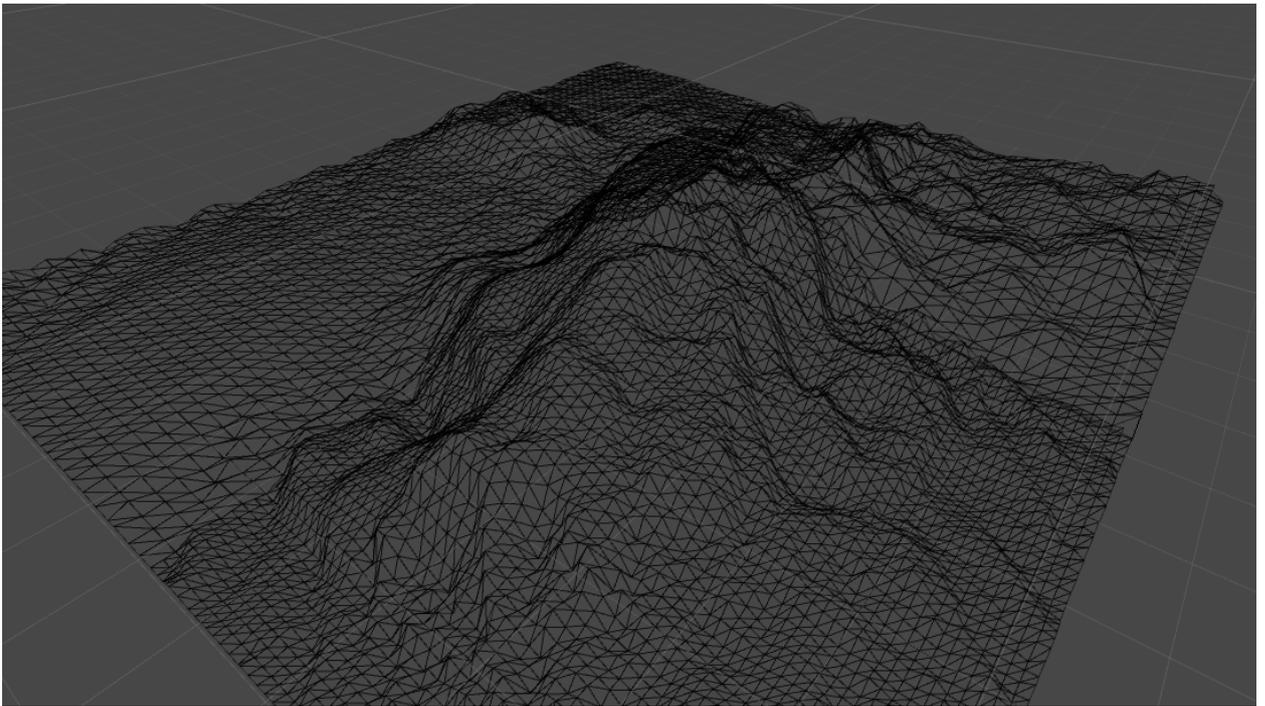


Рисунок 18 – Пример работы программы в режиме отображения
полигональной сетки

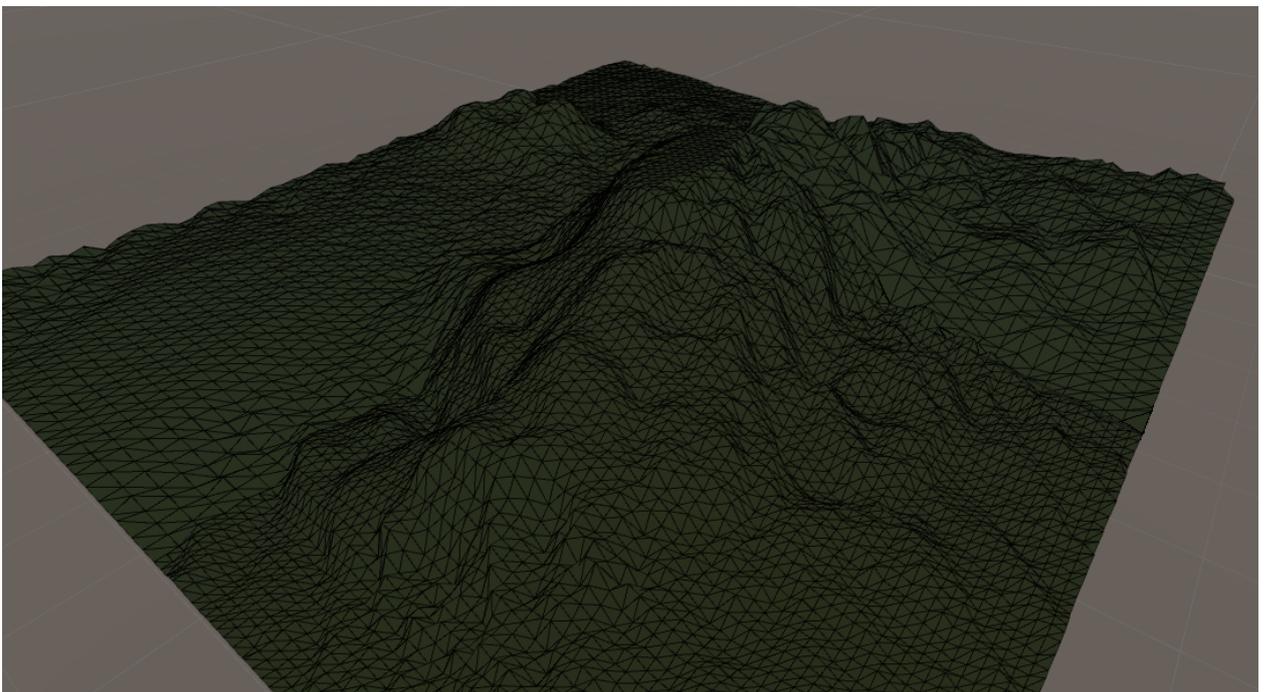


Рисунок 19 – Пример работы программы в режиме покрытия
материалом

Уменьшим LOD до значения 0.5 и посмотрим через камеру игрока.

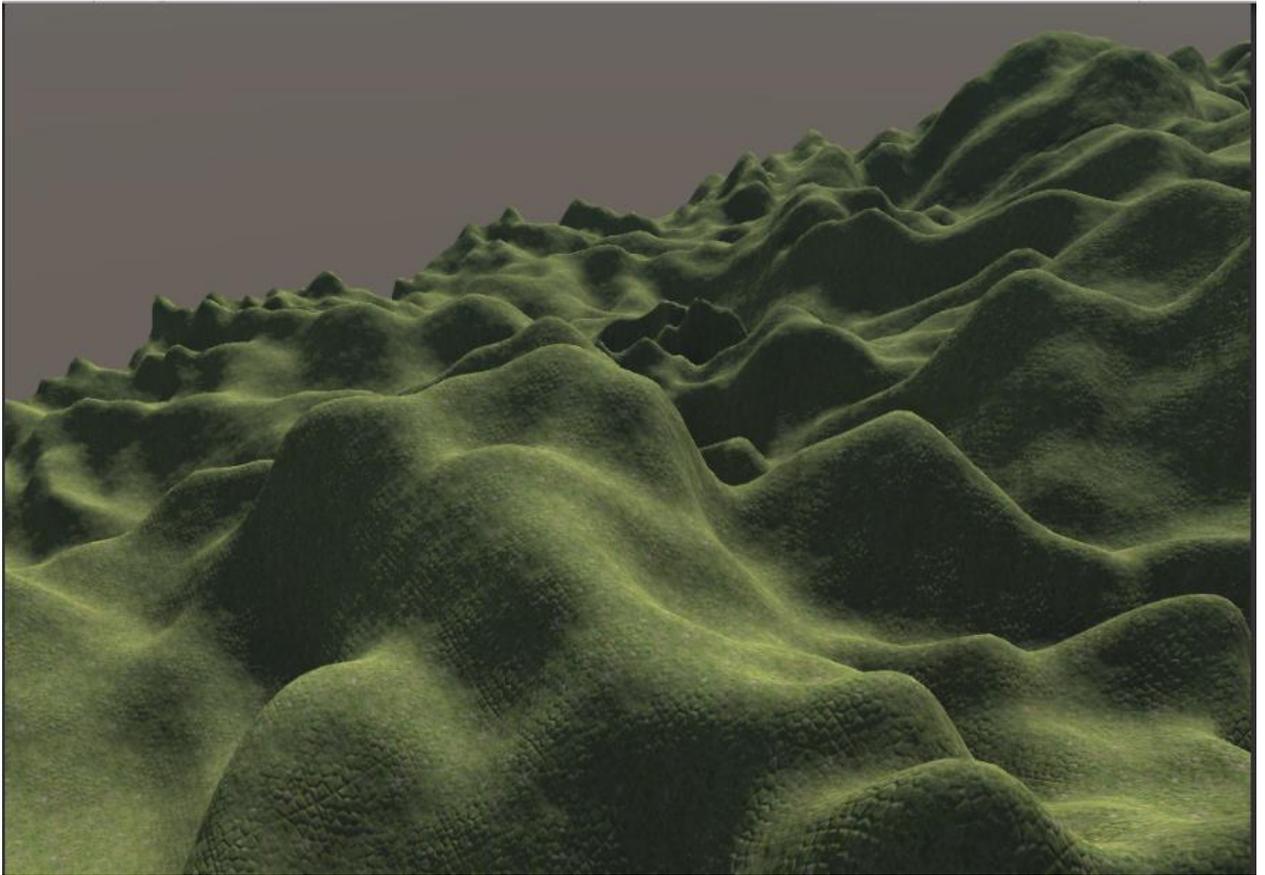


Рисунок 20 – Пример гористой местности

Как видно на Рис. 20, после изменения LOD поверхность ландшафта стала более сглаженной. Пользователь может перемещаться по всей по все поверхности. Для предотвращения «падения» объекта игрока при выходе за область ландшафта, на по краям были размещены объекты с компонентами BoxCollider, которые не дают игроку пройти дальше.

6. ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ

6.1. Оценка коммерческого потенциала и перспективности проведения научных исследований с позиции ресурсоэффективности и ресурсосбережения

6.1.1. Потенциальные потребители результатов исследования

Целевой рынок обусловлен спецификой разрабатываемого специализированного программного обеспечения, например, строительного или игры.

Разработка будет использоваться разработчиками Unity. Данная разработка позволит повысить эффективность создания виртуального ландшафта, снизить затраты на постобработку и увеличить качество изображаемой местности.

6.1.2. Технология QuaD

Технология QuaD (QQuality ADvisor) представляет собой гибкий инструмент измерения характеристик, описывающих качество новой разработки и ее перспективность на рынке и позволяющие принимать решение целесообразности вложения денежных средств в научно-исследовательский проект.

Оценка по технологии QuaD приведена в Таб. 1:

Таблица 1 – Оценочная карта для сравнения конкурентных технических решений (разработок)

Критерии оценки	Вес критерия	Баллы	Максимальный балл	Относительное значение (3/4)	Средневзвешенное значение (5x2)
1	2	3	4	5	
Показатели оценки качества разработки					
1. Энергоэффективность	0.0625	90	100	0.9	0.05625
2. Надежность	0.0625	100	100	1	0.0625

3. Унифицированность	0.0625	100	100	1	0.0625
4. Потребность в ресурсах памяти	0.0625	95	100	0.95	0.059375
5. Функциональная мощность (предоставляемые возможности)	0.0625	100	100	1	0.0625
6. Простота эксплуатации	0.0625	95	100	0.95	0.059375
7. Качество интеллектуального интерфейса	0.0625	90	100	0.9	0.05625
8. Ремонтопригодность	0.0625	100	100	1	0.0625
Показатели оценки коммерческого потенциала разработки					
9. Конкурентоспособность продукта	0.0625	100	100	1	0.0625
10. Уровень проникновения на рынок	0.0625	80	100	0.8	0.05
11. Перспективность рынка	0.0625	100	100	1	0.0625
12. Цена	0.0625	90	100	0.9	0.05625
13. Послепродажное обслуживание	0.0625	100	100	1	0.0625
14. Финансовая эффективность научной разработки	0.0625	100	100	1	0.0625
15. Срок выхода на рынок	0.0625	100	100	1	0.0625
16. Наличие сертификации разработки	0.0625	100	100	1	0.0625
Итого	1				0,9625

Оценка качества и перспективности по технологии QuaD определяется по формуле:

$$P_{cp} = \sum V_i \cdot B_i, \quad (1)$$

где P_{cp} – средневзвешенное значение показателя качества и перспективности научной разработки;

V_i – вес показателя (в долях единицы);

B_i – средневзвешенное значение i -го показателя.

Значение P_{cp} позволяет говорить о перспективах разработки и качестве проведенного исследования. Если значение показателя P_{cp} получилось от 100 до 80, то такая разработка считается перспективной. Если от 79 до 60 – то перспективность выше среднего. Если от 59 до 40 – то перспективность средняя. Если от 39 до 20 – то перспективность ниже среднего. Если 19 и ниже – то перспективность крайне низкая.

$$P_{cp} = 96.25$$

Данное значение лежит в интервале от 80 до 100, следовательно, такая разработка является перспективной.

6.1.3. SWOT – анализ

SWOT – Strengths (сильные стороны), Weaknesses (слабые стороны), Opportunities (возможности) и Threats (угрозы) – представляет собой комплексный анализ научно-исследовательского проекта. SWOT-анализ применяют для исследования внешней и внутренней среды проекта.

Матрица SWOT приведена в Таб. 2.

Таблица 2 – Матрица SWOT

	<p>Сильные стороны:</p> <p>С1. Удобство в эксплуатации. С2. Скорость работы С3. Новизна используемых технологий. С4. Функциональные возможности подсистемы.</p>	<p>Слабые стороны:</p> <p>Сл1. Привязка к определенной среде визуализации. Сл2. Слабое наполнение 3D сцен различным контентом.</p>
<p>Возможности:</p> <p>В1. Спрос на научную разработку в индустрии компьютерных игр.</p>	<p>1) Ориентирование на разработку игровых приложений.</p>	<p>1) Наем новых сотрудников для производства контента.</p>

В2. Повышение стоимости конкурентных разработок.	2) Демонстрация возможностей продукта на IT выставках и конференциях.	2) Параллельная разработка продукта в других средах. 3) Предложения о сотрудничестве разработчикам игровых движков.
Угрозы: У1. Большая конкуренция. У2. Расширение функциональной мощности в разработках конкурентов.	1) Продвижение продукции с акцентированием на достоинствах системы. 2) Расширение функциональных возможностей подсистемы.	1) Перепроектирование подсистемы 2) Улучшение работы с клиентами.

6.2. Определение возможных альтернатив проведения научных исследований

Проект может быть выполнен в различных исполнениях. В данном случае будут рассмотрены 3 исполнения:

Таблица 3 – Морфологическая матрица для НИР

Характеристики разработки	Варианты исполнения		
	1	2	3
А. Среда визуализации	Unity 5	Unreal Engine 4	Cry Engine 2.3
Б. Ограничение времени разработки	Да	Нет	Нет
В. Функционал	Стандартный	Ограниченный	Расширенный
Г. Основное оборудование	Ноутбук	Персональный компьютер общего назначения	Рабочая станция
Д. Исполнитель	Дипломник	Практикант	Разработчик уровня senior

6.3. Планирование научно-исследовательских работ

6.3.1. Структура работ в рамках научного исследования

При организации работы, в том числе и научно-исследовательской необходимо планировать занятость каждого участника и определить сроки выполнения отдельных работ. Следует отметить, что в организации рабочего процесса участвует два исполнителя: дипломник и руководитель темы.

В Таб. 4 определен полный перечень работ и их исполнители:

Таблица 4 – Перечень этапов, работ и распределение исполнителей

Основные этапы	№ раб	Содержание работ	Исполнители
Выбор задания, утверждение темы	1	Выбор и утверждение задания и темы работы	Дипломник, Научный руководитель
Выбор направления исследований	2	Подбор и изучение материалов по теме	Дипломник
	3	Календарное планирование работ по теме	Дипломник, Научный руководитель
	4	Обсуждение литературы	Дипломник, Научный руководитель
Теоретические и экспериментальные исследования	5	Реализация подсистем	Дипломник
	6	Создание сценариев тестирования и тестирование написанного ПО	Дипломник
Обобщение и оценка результатов	7	Анализ результатов	Дипломник, Научный руководитель
	8	Оценка эффективности полученных результатов	Дипломник, Научный руководитель
Оформление отчета по НИР (комплекта документации по ВКР)	9	Оформление отчета по ВКР	Дипломник, Научный руководитель

6.3.2. Определение трудоёмкости выполнения работ

Для построения линейного графика необходимо рассчитать длительность этапов в рабочих днях, а затем перевести в календарные дни. Расчет продолжительности выполнения каждого этапа в рабочих днях выполняется по формуле:

$$T_{РД} = \frac{t_{ож}}{K_{ВН}} \cdot K_{Д}, \quad (2)$$

где $t_{ож}$ – трудоемкость работы, чел/дн.;

$K_{ВН}$ – коэффициент выполнения работ ($K_{ВН} = 1$);

K_d – коэффициент, учитывающий дополнительное время на компенсации и согласование работ ($K_d = 1,2$).

Расчет продолжительности этапа в календарных днях ведется по формуле:

$$T_{Kd} = T_{Рд} * T_K, \quad (3)$$

где $T_{Рд}$ – продолжительность выполнения этапа в рабочих днях,

T_{Kd} – продолжительность выполнения этапа в календарных днях,

T_K – коэффициент календарности.

Коэффициент календарности рассчитывается по формуле:

$$T_K = \frac{T_{кал}}{T_{кал} - T_{вд} - T_{пд}}, \quad (4)$$

где $T_{КАЛ}$ – календарные дни ($T_{КАЛ} = 365$),

$T_{ВД}$ – выходные дни ($T_{ВД} = 104$),

$T_{ПД}$ – праздничные дни ($T_{ПД} = 14$).

$$T_K = \frac{365}{365 - 118} = 1.48$$

Для расчета ожидаемого значения продолжительности работ $t_{ож}$ применяется две оценки: t_{min} и t_{max} (метод двух оценок).

$$t_{ож} = \frac{3t_{min} + 2t_{max}}{5}, \quad (5)$$

где t_{min} – минимальная трудоемкость работ, чел/дн.,

t_{max} – максимальная трудоемкость работ, чел/дн.

Расчет трудозатрат на выполнение проекта приведен в Таб. 5.

Календарный график проведенной ВКР представлен в Таб. 6.

Таблица 5 – Трудозатраты на выполнение проекта

Этап	Исполнители	Продолжительность работ, дни									Длительность работ, чел/дн.											
		t_{min}			t_{max}			$t_{ож}$			$T_{рД}$						$T_{кД}$					
											НР			Д			НР			Д		
		Исп. 1	Исп. 2	Исп. 3	Исп. 1	Исп. 2	Исп. 3	Исп. 1	Исп. 2	Исп. 3	Исп. 1	Исп. 2	Исп. 3	Исп. 1	Исп. 2	Исп. 3	Исп. 1	Исп. 2	Исп. 3	Исп. 1	Исп. 2	Исп. 3
1. Выбор и утверждение задания и темы работы	НР	1	1	1	2	1	2	1,4	1	1,4	1,68	1,2	1,68	0	0	0	2,49	1,78	2,49	0	0	0
2. Подбор и изучение материалов по теме	НР, Д	4	4	4	6	5	6	4,8	4,4	4,8	2,88	2,64	2,88	2,88	2,64	2,88	4,26	3,91	4,26	4,26	3,91	4,26
3. Календарное планирование работ по теме	НР, Д	1	1	1	2	1	2	1,4	1	1,4	0,84	0,6	0,84	0,84	0,6	0,84	1,24	0,89	1,24	1,24	0,89	1,24
4. Обсуждение литературы	НР, Д	4	4	4	6	5	6	4,8	4,4	4,8	2,88	2,64	2,88	2,88	2,64	2,88	4,26	3,91	4,26	4,26	3,91	4,26
5. Программная реализация	Д	45	45	45	60	56	65	51	49,4	53	0	0	0	61,2	59,3	63,6	0	0	0	90,5 8	87,7 6	94,1 3
6. Создание сценариев тестирования и тестирование написанного ПО	Д	2	2	2	3	2	5	2,4	2	3,2	0	0	0	2,88	2,4	3,84	0	0	0	4,26	3,55	5,68
7. Анализ результатов	Д	2	2	2	3	2	3	2,4	2	2,4	0	0	0	2,88	2,4	2,88	0	0	0	4,26	3,55	4,26
8. Оценка эффективности полученных результатов	Д	2	2	2	3	2	3	2,4	2	2,4	0	0	0	2,88	2,4	2,88	0	0	0	4,26	3,55	4,26
9. Оформление отчета по ВКР	НР, Д	7	7	7	10	9	11	8,2	7,8	8,6	4,92	4,68	5,16	4,92	4,68	5,16	7,26	6,93	7,64	7,26	6,93	7,64
Итого:								78,8	74	82	13,2	11,7 6	13,4 4	81,3 6	77,0 6	84,9 6	19,5 1	16,4 2	19,8 9	120, 38	114, 05	126, 03

Таблица 6 – Календарный план-график проведения НИР по теме

Этап	Т _р , НР	Т _р , С	Дата начала	Дата окончания	Февраль				Март				Апрель				Май				Июнь				
					1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4					
1. Выбор и утверждение задания и темы работы	2	0	07.02.2017	10.02.2017		■																			
2. Подбор и изучение материалов по теме	3	3	11.02.2017	17.02.2017		■	▨																		
3. Календарное планирование работ по теме	1	1	17.02.2017	19.02.2017			■	▨																	
4. Обсуждение литературы	3	3	19.02.2017	25.02.2017				■	▨																
5. Реализация подсистем	0	62	26.02.2017	01.05.2017						▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	
6. Создание сценариев тестирования и тестирование написанного ПО	0	3	02.05.2017	05.05.2017																			▨	▨	
7. Анализ результатов	0	3	06.05.2017	09.05.2017																				▨	▨
8. Оценка эффективности полученных результатов	0	3	10.05.2017	13.05.2017																				▨	▨
9. Оформление отчета по ВКР	5	5	13.05.2017	24.05.2017																				▨	▨

6.3.3. Бюджет научно-технического исследования (НТИ)

6.3.3.1. Расчет материальных затрат НТИ

Расчет материальных затрат осуществляется по следующей формуле:

$$Z_M = (1 + k_T) * \sum_{i=1}^m C_i + N_{рас\ xi}, \quad (6)$$

где m – количество видов материальных ресурсов, потребляемых при выполнении научного исследования;

$N_{рас\ xi}$ – количество материальных ресурсов i -го вида, планируемых к использованию при выполнении научного исследования (шт., кг, м, м² и т.д.),

C_i – цена приобретения единицы i -го вида потребляемых материальных ресурсов (руб./шт., руб./кг, руб./м, руб./м² и т.д.),

k_T – коэффициент, учитывающий транспортно-заготовительные расходы.

Транспортные расходы примем в размере 20% от стоимости материалов.

Материальные затраты рассчитаны в Таб. 7:

Таблица 7 – Материальные затраты

Наименование	Единица измерения	Количество			Цена за ед., руб.			Затраты на материалы, (Z_M), руб.		
		Исп.1	Исп.2	Исп.3	Исп.1	Исп.2	Исп.3	Исп.1	Исп.2	Исп.3
Лицензия среды визуализации	шт.	1	1	1	4000 0	5000 0	6000 0	4800 0	6000 0	7200 0
Внешний жесткий диск	шт.	1	1	1	2000	3000	4000	2400	3600	4800
Итого								5040 0	6360 0	7680 0

6.3.3.2. Расчет затрат на специальное оборудование для научных (экспериментальных) работ

Амортизационные отчисления рассчитываются по формуле:

$$Z_{ам} = \frac{C_i * H_a}{100\%}, \quad (7)$$

где $Z_{ам}$ – ежемесячная сумма амортизационных отчислений,

C_i – цена (балансовая стоимость) i -го оборудования,

H_a - норма амортизационных отчислений (%), которая в соответствии с Налоговыми кодексом РФ определяется по следующей формуле:

$$H_a = \frac{1}{T_{п.и.}} * 100\%, \quad (8)$$

где $T_{п.и.}$ – срок полезного использования объекта (в днях)

определяется в соответствии с Классификацией основных средств, включаемых в амортизационные группы.

Амортизационные отчисления рассчитаны в Таб. 8:

Таблица 8 – Величина амортизационных отчислений

Наименование	Количество			Ст. первонач., руб.			Т п.и., дни			На, % в мес			А в мес., руб.			А за период, руб.		
	Исп. 1	Исп. 2	Исп. 3	Исп. 1	Исп. 2	Исп. 3	Исп. 1	Исп. 2	Исп. 3	Исп. 1	Исп. 2	Исп. 3	Исп. 1	Исп. 2	Исп. 3	Исп. 1	Исп. 2	Исп. 3
Настольный ПК	1	1	1	40000	50000	60000	730			0,137			54,8	68,5	82,2	219,2	274	328,8
Всего																219,2	274	328,8

6.3.3.3. Основная заработная плата исполнителей темы

В этой статье расходов планируется и учитывается основная заработная плата исполнителей, непосредственно участвующих в проектировании разработки:

$$C_{осн зп} = \sum t_i * C_{зп_i}, \quad (9)$$

где t_i - затраты труда, необходимые для выполнения i -го вида работ, в рабочих днях,

$C_{зп_i}$ - среднедневная заработная плата работника, выполняющего i -ый вид работ, (руб./день).

Среднедневная заработная плата определяется по формуле:

$$C_{зп_i} = \frac{D + D * K}{F}, \quad (10)$$

где D – месячный оклад работника (в соответствии с квалификационным уровнем профессиональной квалификационной группы),

K - районный коэффициент (для Томска – 30%),

F – количество рабочих дней в месяце (в среднем 22 дня).

Расходы на основную заработную плату определяются как произведение трудоемкости работ каждого исполнителя на среднедневную заработную плату. Оклад руководителя (доцент, к.ф.-м.н.) составляет 23264,86 руб/мес. Оклад бакалавра (1 кв. уровень УВП) составляет 6976,22 руб/мес.

Расчет затрат на основную заработную плату приведен в Таб. 9:

Таблица 9 – Расчет основной заработной платы

Исполнитель	Оклад, руб.	Средняя заработная плата, руб./дн.	Трудоемкость, раб. дн.			Основная заработная плата, руб.		
			Исп. 1	Исп. 2	Исп. 3	Исп. 1	Исп. 2	Исп. 3
Руководитель	23264,86	1374,74						
			13,2	11,76	13,4	18146,57	16166,94	18421,52
Дипломник	6976,22	412,21	81,36	77,06	84,96	33537,41	31764,9	35021,36
ИТОГО						51683,98	47931,84	53442,88

6.3.3.4. Дополнительная заработная плата исполнителей темы

Результаты расчётов дополнительной заработной платы представлены в Таб. 10:

Таблица 10 – Расчет дополнительной заработной платы

Исполнитель	Основная заработная плата, руб.			Коэффициент дополнительной заработной платы	Дополнительная заработная плата, руб.		
Руководитель	18146,57	16166,94	18421,52	0,15	2721,99	2425,04	2763,23
Дипломник	33537,41	31764,9	35021,36		5030,61	4764,74	5253,21
ИТОГО					7752,6	7189,78	8016,44

6.3.3.5. Отчисления во внебюджетные фонды (страховые отчисления)

В данной статье расходов отражаются обязательные отчисления по установленным законодательством Российской Федерации нормам органам государственного социального страхования (ФСС), пенсионного фонда (ПФ) и медицинского страхования (ФФОМС) от затрат на оплату труда работников.

Величина отчислений во внебюджетные фонды определяется исходя из следующей формулы:

$$C_{\text{внеб}} = k_{\text{внеб}} * (C_{\text{осн зп}} + C_{\text{доп зп}}), \quad (11)$$

где $k_{\text{внеб}}$ – коэффициент отчислений на уплату во внебюджетные фонды (пенсионный фонд, фонд обязательного медицинского страхования и пр.).

Расчет отчислений во внебюджетные фонды представлен в Таб. 11:

Таблица 11 – Отчисления во внебюджетные фонды

Исполнитель	Основная заработная плата, руб.			Дополнительная заработная плата, руб.			Отчисления во внебюджетные фонды		
	Исп. 1	Исп. 2	Исп. 3	Исп. 1	Исп. 2	Исп. 3	Исп. 1	Исп. 2	Исп. 3
Руководитель	18146,57	16166,94	18421,52	2721,99	2425,04	2763,23	6260,57	5577,59	6355,43
Дипломник	33537,41	31764,9	35021,36	5030,61	4764,74	5253,21	11570,41	10958,89	12082,37
Коэффициент отчислений во внебюджетные фонды	30%								
Итого							17830,98	16536,48	18437,8

6.3.3.6. Накладные расходы

Накладные расходы учитывают прочие затраты организации, не попавшие в предыдущие статьи расходов: печать и ксерокопирование материалов исследования, оплата услуг связи, электроэнергия, почтовые и телеграфные расходы, размножение материалов и т.д. Их величина определяется по следующей формуле:

$$Z_{\text{накл}} = (\text{сумма статей } 1 \div 5) \cdot k_{\text{нр}}, \quad (12)$$

где $k_{\text{нр}}$ – коэффициент, учитывающий накладные расходы.

Величину коэффициента накладных расходов можно взять в размере 16%.

Расчет накладных расходов представлен в Таб. 12:

Таблица 12 – Расчет накладных расходов

	$k_{\text{нр}}$	Статья 1	Статья 2	Статья 3	Статья 4	Статья 5	$Z_{\text{накл}}$
Исп. 1	0,16	50400	219,2	51683,98	7752,6	17830,98	20461,88
Исп. 2		63600	274	47931,84	7189,78	16536,48	21685,14
Исп. 3		76800	328,8	53442,88	8016,44	18437,8	25124,15

6.3.3.7. Формирование бюджета затрат научно-исследовательского проекта

Рассчитанная величина затрат научно-исследовательской работы (темы) является основой для формирования бюджета затрат проекта, который при формировании договора с заказчиком защищается научной организацией в качестве нижнего предела затрат на разработку научно-технической продукции.

Определение бюджета затрат на научно-исследовательский проект по каждому варианту исполнения приведен в Таб. 13:

Таблица 13 – Расчет бюджета затрат НТИ

Наименование статьи	Сумма, руб.		
	Исп.1	Исп.2	Исп.3
1. Материальные затраты НТИ	50400	63600	76800
2. Затраты на специальное оборудование для научных (экспериментальных) работ	219,2	274	328,8
3. Затраты по основной заработной плате исполнителей темы	51683,98	47931,84	53442,88
4. Затраты по дополнительной заработной плате исполнителей темы	7752,6	7189,78	8016,44
5. Отчисления во внебюджетные фонды	17830,98	16536,48	18437,8
6. Накладные расходы	20461,88	21685,14	25124,15
7. Бюджет затрат НТИ	148348,64	157217,24	182150,07

6.4. Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования

Для выбора варианта исполнения проекта необходимо рассчитать наиболее эффективный путь исследования.

$$I_{\text{финр}}^{\text{исп.}i} = \frac{\Phi_{pi}}{\Phi_{\text{max}}}, \quad (13)$$

где $I_{\text{финр}}^{\text{исп.}i}$ – интегральный финансовый показатель разработки;

Φ_{pi} – стоимость i -го варианта исполнения;

Φ_{max} – максимальная стоимость исполнения научно-исследовательского проекта (в т.ч. аналоги).

Интегральный финансовый показатель разработки:

$$I_{\text{финр}}^{\text{исп.1}} = 148348,64 / 182150,07 = 0,81,$$

$$I_{\text{финр}}^{\text{исп.2}} = 157217,24 / 182150,07 = 0,86,$$

$$I_{\text{финр}}^{\text{исп.3}} = 182150,07 / 182150,07 = 1.$$

Интегральный показатель ресурсоэффективности разработки:

Сравнительная оценка характеристик вариантов исполнения проекта представлена в Таб. 1414:

Таблица 14 – Сравнительная оценка характеристик вариантов исполнения проекта

Объект исследования Критерии	Весовой коэффициент параметра	Исп.1	Исп.2	Исп.3
1. Способствует росту производительности труда пользователя	0,1	5	3	4
2. Удобство в эксплуатации (соответствует требованиям потребителей)	0,15	5	2	3
3. Помехоустойчивость	0,15	5	3	3
4. Энергосбережение	0,20	4	3	3
5. Надежность	0,25	5	4	4
6. Материалоемкость	0,15	4	4	4
ИТОГО	1			

$$I_{pi} = \sum a_i \cdot b_i, \quad (14)$$

где I_{pi} – интегральный показатель ресурсоэффективности для i -го варианта исполнения разработки,

a_i – весовой коэффициент i -го варианта исполнения разработки,

b_i^a, b_i^p – бальная оценка i -го варианта исполнения разработки, устанавливается экспертным путем по выбранной шкале оценивания,

n – число параметров сравнения.

$$I_{p-исп1} = 5*0,1+5*0,15+5*0,15+4*0,2+5*0,25+5*0,15 = 4,8,$$

$$I_{p-исп2} = 3*0,1+2*0,15+3*0,15+3*0,2+4*0,25+4*0,15 = 3,25,$$

$$I_{p-исп3} = 4*0,1+3*0,15+3*0,15+3*0,2+4*0,25+4*0,15 = 3,5.$$

Интегральный показатель эффективности:

$$I_{исп.1} = \frac{I_{p-исп1}}{I_{финр}} \quad (15)$$

$$I_{исп1} = 4,8 / 0,81 = 5,93,$$

$$I_{исп2} = 3,25 / 0,86 = 3,78,$$

$$I_{исп3} = 3,5 / 1 = 3,5.$$

Сравнительная эффективность вариантов исполнения:

$$\mathcal{E}_{ср} = \frac{I_{исп i}}{I_{исп min}} \quad (16)$$

$$\mathcal{E}_{ср1} = 5,93 / 3,5 = 1,$$

$$\mathcal{E}_{ср2} = 3,78 / 3,5 = 0,64,$$

$$\mathcal{E}_{ср3} = 3,5 / 3,5 = 1.$$

Сравнительная эффективность разработки представлена в Таб. 15.

Таблица 15 – Сравнительная эффективность разработки

№ п/п	Показатели	Исп.1	Исп.2	Исп.3
1	Интегральный финансовый показатель разработки	0,81	0,86	1
2	Интегральный показатель ресурсоэффективности разработки	4,8	3,25	3,5
3	Интегральный показатель эффективности	5,93	3,78	3,5
4	Сравнительная эффективность вариантов исполнения	1,69	1,08	1

Выбранный вариант исполнения 1, для решения поставленной задачи, наиболее эффективен. Показатели его эффективности больше 1, следовательно, эффективность внедрения данного варианта исполнения более высока. Другие исполнения имеют показатели эффективности, также больше 1. Однако, они ниже, чем у выбранного варианта исполнения. Данная особенность обусловлена увеличением трудозатрат в 3-м варианте исполнения, и увеличении материальных затрат во 2-ом варианте исполнения.

7. СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ

7.1. Производственная безопасность

7.1.1. Анализ вредных и опасных факторов, которые может создать объект исследования

Объектом исследования ВКР является программная библиотека генерации 3D ландшафта, который осуществляет работу с графическими данными. Поэтому объект исследования не представляет угрозы ни для прочих приложений, установленных и работающих в той же исполнительной среде, ни для возможных сетевых функций среды исполнения, не содержит и не является вредоносным программным обеспечением, которое может распространяться через сеть.

7.1.2. Анализ вредных и опасных факторов, которые могут возникнуть на рабочем месте при проведении исследований

7.1.2.1. Описание рабочего места

Научно-исследовательская работа выполнялась в рабочем кабинете, оснащенном персональными электронно-вычислительными машинами (ПЭВМ).

В процессе работы на бакалавра-программиста действуют множество опасных и вредных факторов. Задача охраны труда свести действие этих факторов к минимуму и создать оптимальные условия труда.

Работа с ЭВМ регламентируется санитарными правилами и нормами [1].

Рабочее место находится на пятом этаже здания и представляет собой комнату длиной – 5 м, шириной – 4 м и высотой – 3 м. Естественное освещение кабинета осуществляется посредством двух окон размерами 1,7 м х 1,5 м. Дверь – деревянная, одностворчатая, белого цвета. Высота двери – 2 м, ширина - 1 м. Стены комнаты окрашены водоэмульсионной краской синего и зеленого оттенков. Потолок побелен. Пол покрыт плиткой. Площадь кабинета составляет 20 м², объем – 60 м³.

Помещение оборудовано на одно рабочее место, где установлен один персональный компьютер с двумя ЖК мониторами. Требования, которые

определены к минимальной площади и объему на одно рабочее место – при периметральном расположении площадь одного рабочего места должна быть не менее 4,0 м² – для данного помещения выполняются [10].

7.1.2.2. Техногенная безопасность

Для данной рабочей зоны характерны факторы, приведенные в Таб. 16:

Таблица 16 – Опасные и вредные факторы при проведении работ

Источник фактора, наименование видов работ	Факторы (по ГОСТ 12.0.003-74)		Нормативные документы
	Вредные	Опасные	
Разработка и тестирование программы генерации 3D ландшафта	1) Микроклимат; 2) Шум; 3) Электромагнитные поля; 4) Освещение.	1) Возникновение пожара; 2) Поражение электрическим током.	1) СанПиН 2.2.2/2.4.1340 – 03; 2) СанПиН 2.2.1/2.1.1.1278-03; 3) СНиП 21 – 01 – 97.

7.1.2.3. Вредные факторы

7.1.2.3.1. Микроклимат

Микроклимат является важной характеристикой производственных помещений. В организме человека происходит непрерывное выделение тепла. Одновременно с процессами выделения тепла происходит непрерывная теплоотдача в окружающую среду. Равновесие между выделением тепла и теплоотдачей регулируется процессами терморегуляции, т.е. способностью организма поддерживать постоянство теплообмена с сохранением постоянной температуры тела. Отдача тепла происходит различными видами: излучением, конвекцией, испарение влаги.

Нарушение теплового баланса в условиях высокой температуры может привести к перегреву тела, и как следствие к тепловым ударам с потерей сознания. В условиях низкой температуры воздуха возможно переохлаждение организма, могут возникнуть простудные болезни, радикулит, бронхит и другие заболевания.

К параметрам микроклимата относятся:

- температура воздуха;
- температура поверхностей;
- относительная влажность воздуха;
- скорость движения воздуха.

Оптимальные значения этих характеристик зависят от сезона (холодный, тёплый), а также от категории физической тяжести работы. Для инженера-программиста она является лёгкой (1а), так как работа проводится сидя, без систематических физических нагрузок.

Согласно требованиям, оптимальные параметры микроклимата в офисах приведены в Таб. 17 [10]:

Таблица 17 – Оптимальные значения характеристик микроклимата

Период года	Температура воздуха, °С	Температура поверхностей, °С	Относительная влажность, %	Скорость движения воздуха, м/с
Холодный	22-24	21-25	40-60	0,1
Тёплый	23-25	22-26	40-60	0,1

7.1.2.3.2. Шум

Одной из важных характеристик производственных помещений является уровень шума.

Основными источниками шума в помещении являются:

- система охлаждения центральных процессоров;
- жесткие диски;

– шум с улицы.

Повышенный уровень шума неблагоприятно воздействует на организм человека в целом, так и на нервную систему и органы слуха в частности, что ведет к падению производительности труда и может привести к развитию заболеваний нервной системы и снижению слуха.

При выполнении основной работы на ПЭВМ уровень шума на рабочем месте не должен превышать 50 дБ. Допустимые уровни звукового давления в помещениях для персонала, осуществляющего эксплуатацию ЭВМ при разных значениях частот, приведены в Таб. 18 [10]:

Таблица 18 – Допустимые уровни звука на рабочем месте

Вид трудовой деятельности, рабочее место	Уровни звукового давления, дБ, в октавных полосах со среднегеометрическими частотами, Гц									Уровни звука и эквивалентного звука (в дБА)
	31,5	63	125	250	500	1000	2000	4000	8000	
Конструкторские бюро, программисты, лаборатории	86	71	61	54	49	45	42	40	38	50

7.1.2.3.3. Электромагнитные поля

Воздействие электромагнитных полей на человека зависит от напряженностей электрического и магнитного полей, потока энергии, частоты колебаний, размера облучаемого тела.

Нарушения в организме человека при воздействии электромагнитных полей незначительных напряженностей носят необратимый характер. При воздействии полей, имеющих напряженность выше предельно допустимого уровня, развиваются нарушения со стороны нервной, сердечно-сосудистой систем, органов пищеварения и некоторых биологических показателей крови.

Работа проводилась на современном компьютере, где значения электромагнитного излучения малы и отвечают требованиям, которые приведены в Таб. 19 [10]:

Таблица 19 – Допустимые уровни электромагнитных полей

Наименование параметров	Допустимые значения
Напряженность электромагнитного поля на расстоянии 50 см. вокруг ВДТ по электрической составляющей должна быть не более: в диапазоне частот 5 Гц – 2 кГц в диапазоне частот 2 – 400 кГц	25 В/м 2.5 В/м
Плотность магнитного потока должна быть не более: в диапазоне частот 5 Гц – 2 кГц в диапазоне частот 2 – 400 кГц	250 нТл 25 нТл
Напряженность электростатического поля:	15 кВ/м

7.1.2.3.4. Освещение

Около 80% общего объема информации человек получает через зрительный канал. Качество поступающей информации во многом зависит от освещения, неудовлетворительное качество которого вызывает утомление организма в целом. При неудовлетворительном освещении снижается производительность труда и увеличивается количество допускаемых программистом ошибок.

Так как работа инженера-программиста подразумевает зрительный тип работы, то организация правильного освещения имеет значительное место. Пренебрежение данным фактором может привести к профессиональным болезням зрения.

В рабочем помещении сочетаются естественное освещение (через окна) и искусственное освещение (использование ламп при недостатке естественного освещения) [11].

Светильники в помещении располагаются равномерно по площади потолка, тем самым обеспечивая равномерное освещение рабочих мест.

Разряд зрительных работ программиста относится к разряду III подразряду г (высокой точности), параметры искусственного освещения указаны в Таб. 20 [10]:

Таблица 20 – Нормативные значения освещённости

Характеристика зрительной работы	Наименьший или эквивалентный размер объекта различения, мм	Разряд зрительной работы	Подразряд зрительной работы	Контраст объекта с фоном	Характеристика фона	Искусственное освещение		
						Освещённость, лк		
						При системе комбинированного освещения		При системе общего освещения
всего	В том числе от общего							
Высокой точности	От 0,3 до 0,5	III	Г	Средний и большой <<	Светлый << << средний	400	200	200

7.1.2.4. Опасные факторы

7.1.2.4.1. Опасность возникновения пожара

В помещениях с ПЭВМ повышен риск возникновения пожара. Неисправность электрооборудования, освещения, неправильная их эксплуатация, наличие статического электричества неудовлетворительный надзор за пожарными устройствами и производственным оборудованием может послужить причиной пожара. Пожар на предприятии наносит большой материальный ущерб и часто сопровождается несчастными случаями с людьми.

Пожарная безопасность включает в себя комплекс организационных и технических мероприятий, направленных на обеспечение безопасности людей, предотвращения пожара, ограничение его распространения, а также создание условия для успешного тушения пожара.

Для профилактики пожара должны обеспечиваться регулярные проверки пожарной сигнализации, первичных средств пожаротушения; проводиться инструктаж и тренировки по действиям в случае пожара; не загромождаться и не блокироваться эвакуационные выходы; выполняться

требования правил технической эксплуатации и правил техники безопасности при эксплуатации электроустановок; во всех служебных помещениях должен быть установлен «План эвакуации людей при пожаре», регламентирующий действия персонала в случае возникновения очага возгорания и указывающий места расположения пожарной техники [12, 14].

7.1.2.4.1. Опасность поражения током

В связи с наличием электрооборудования для данного производственного объекта характерным является возможность поражения электрическим током. Для снижения данного риска необходимо соблюдать нормы электробезопасности.

Электробезопасность — это система организационных и технических мероприятий и средств, обеспечивающих защиту людей от вредного и опасного для жизни воздействия электрического тока, электрической дуги, электромагнитного поля и статического электричества.

Опасность поражения электрическим током, в отличие от прочих опасностей, усугубляется тем, что человек не в состоянии без специальных приборов обнаружить напряжение дистанционно, как, например, движущейся части, раскалённый объект, открытые люки, не ограждённые площадки на больших высотах. Опасность обнаруживается слишком поздно — когда человек уже поражён.

Помещение, где расположены персональные вычислительные машины, относится к помещениям без повышенной опасности, так как отсутствуют следующие факторы [10]:

- сырость;
- токопроводящая пыль;
- токопроводящие полы;
- высокая температура;

– возможность одновременного прикосновения человека к имеющим соединение с землёй металлоконструкциям зданий, технологическим аппаратам и механизмам и металлическим корпусам электрооборудования.

7.1.3. Обоснование мероприятий по защите исследователя от действия опасных и вредных факторов

7.1.3.1. Защита от действия вредных факторов

7.1.3.1.1. Микроклимат

Для обеспечения комфортных условий используются как организационные методы (рациональная организация проведения работ в зависимости от времени года и суток, чередование труда и отдыха), так и технические средства (вентиляция, кондиционирование воздуха, отопительная система).

7.1.3.1.2. Шум

Для снижения уровня шума, производимого персональными компьютерами, регулярно проводится их техническое обслуживание: чистка от пыли, замена смазывающих веществ; также применяются звукопоглощающие материалы. Для снижения уровня шума с улицы в окнах установлены герметичные стеклопакеты, также рекомендуется посадка зеленых насаждений на прилегающей территории.

7.1.3.1.3. Электромагнитные поля

Основной способ снижения вредного воздействия – это увеличение расстояния от источника (не менее 50 см от пользователя). Также рекомендуется применять мониторы с пониженным уровнем излучения, устанавливать защитные экраны, а также соблюдать регламентированные режимы труда и отдыха.

7.1.3.1.4. Освещение

Согласно СНиП II-4-79, в помещениях вычислительных центров необходимо применять систему комбинированного освещения.

При выполнении работ категории высокой зрительной точности (наименьший размер объекта различения 0,3...0,5мм) величина коэффициента

естественного освещения (КЕО) должна быть не ниже 1,5%, а при зрительной работе средней точности (наименьший размер объекта различения 0,5...1,0 мм) КЕО должен быть не ниже 1,0%. В качестве источников искусственного освещения обычно используются люминесцентные лампы типа ЛБ или ДРЛ, которые попарно объединяются в светильники, которые должны располагаться над рабочими поверхностями равномерно [11].

Требования к освещенности в помещениях, где установлены компьютеры, следующие: при выполнении зрительных работ высокой точности общая освещенность должна составлять 300лк, а комбинированная - 750лк; аналогичные требования при выполнении работ средней точности - 200 и 300лк соответственно.

Кроме того, все поле зрения должно быть освещено достаточно равномерно – это основное гигиеническое требование. Иными словами, степень освещения помещения и яркость экрана компьютера должны быть примерно одинаковыми, т.к. яркий свет в районе периферийного зрения значительно увеличивает напряженность глаз и, как следствие, приводит к их быстрой утомляемости.

7.1.3.2. Защита от опасных факторов

7.1.3.2.1. Опасность возникновения пожара

Для предотвращения пожара помещение с ПЭВМ должно быть оборудовано первичными средствами пожаротушения: углекислотным огнетушителем типа ОУ-2 или ОУ-5. Также помещение должно быть оснащено пожарной сигнализацией. Рекомендуемый тип — система на основе оптических пожарных извещателей ДИП-3СУ и пульта Сигнал-20П SMD.C-2000. Рекомендуется также оборудовать помещение автоматической установкой объемного газового пожаротушения, например, системой азотного пожаротушения «Гарсис».

7.1.3.2.2. Опасность поражения током

Персональный компьютер питается от сети 220В переменного тока с частотой 50Гц. Это напряжение опасно для жизни, поэтому обязательно мер техники безопасности (ТБ):

- перед началом работы нужно убедиться, что выключатели и розетка закреплены и не имеют оголённых токоведущих частей

- при обнаружении неисправности оборудования и приборов необходимо, не делая никаких самостоятельных исправлений, сообщить человеку, ответственному за оборудование

- К мероприятиям по предотвращению возможности поражения электрическим током следует отнести:

- при производстве монтажных работ необходимо использовать только исправный инструмент, аттестованный службой КИПиА.

- с целью защиты от поражения электрическим током, возникающим между корпусом приборов и инструментом при пробое сетевого напряжения на корпус, корпуса приборов и инструментов должны быть заземлены.

- при включенном сетевом напряжении работы на задней панели должны быть запрещены.

- все работы по устранению неисправностей должен производить квалифицированный персонал.

- необходимо постоянно следить за исправностью электропроводки.

7.2. Экологическая безопасность

Под охраной окружающей среды характеризуется различного рода мероприятиями, влияющие на следующие природные зоны:

- атмосфера;
- гидросфера;
- литосфера.

7.2.1. Анализ влияния объекта исследования на окружающую среду

Помещение с персональным компьютером относится к пятому классу, размер санитарно-защитной зоны которого равен 50 метров [16], так как работа на персональном компьютере не является экологически опасной.

7.2.2. Анализ влияния процесса исследования на окружающую среду

7.2.2.1. Загрязнение литосферы

Основные виды загрязнения литосферы – твердые бытовые и промышленные отходы [13].

7.2.3. Обоснование мероприятий по защите окружающей среды

7.2.3.1. Загрязнение литосферы

В ходе выполнения ВКР, образовывались различные твердые отходы. К ним можно отнести: бумагу, батарейки, лампочки, использованные картриджи, отходы от продуктов питания и личной гигиены, отходы от канцелярских принадлежностей и т.д.

Защита почвенного покрова и недр от твердых отходов реализуется за счет сбора, сортирования и утилизации отходов и их организованного захоронения.

7.3. Безопасность в чрезвычайных ситуациях

7.3.1. Анализ вероятных ЧС, которые может инициировать объект исследований

Как утверждалось ранее, объект исследования (программная библиотека) не создает вредных или опасных факторов. Однако, высокая вычислительная нагрузка на вычислительное устройство может привести к перегреву устройства, что может являться одной из причин возгорания аппаратных компонентов. Для предотвращения возгорания современные ЭВМ снабжают механизмами, которые отключают аппаратуру при перегреве.

7.3.2. Анализ вероятных ЧС, которые могут возникнуть на рабочем месте при проведении исследований

Чрезвычайные ситуации характерные для данного объекта – пожар. Подробнее об опасности возникновения пожара см. п. 1.2.4.1 «Опасность возникновения пожара».

7.3.3. Обоснование мероприятий по предотвращению ЧС и разработка порядка действия в случае возникновения ЧС

7.3.3.1. Пожарная безопасность

Согласно определению категорий помещений, зданий и наружных установок по взрывопожарной и пожарной безопасности (НПБ 105-03) производства подразделяются по пожарной и взрывной опасности на категории А, Б, В, Г, Д. Категории взрывопожарной и пожарной опасности помещений и зданий следует определять для наиболее неблагоприятного в отношении пожара или взрыва периода исходя из вида находящихся в аппаратах и помещениях горючих веществ и материалов, их количества и пожароопасных свойств, особенностей технологических процессов.

Помещение по пожарной и взрывной опасности относят к категории Д (пониженная пожароопасность), характеризующейся отсутствием легковоспламеняющихся веществ и материалов в горячем состоянии.

При строительстве зданий и сооружений с учётом категории производства применяют строительные материалы и конструкции, которые подразделяются на три группы:

- сгораемые;
- трудносгораемые;
- несгораемые.

Здание, в котором находится помещение относится к несгораемым.

Для предотвращения пожара помещение с ПЭВМ должно быть оборудовано первичными средствами пожаротушения: углекислотным огнетушителем типа ОУ 2 или ОУ 5.

В случае возникновения ЧС как пожар, необходимо предпринять меры по эвакуации из помещения в соответствии с планом эвакуации, которые

расположены на каждом этаже здания. План эвакуации для 5 этажа, где находится помещение, показан на Рис. 21.

При невозможности самостоятельно потушить пожар необходимо вызвать пожарную команду, после чего поставить в известность о случившемся инженера по технике безопасности [12, 14].

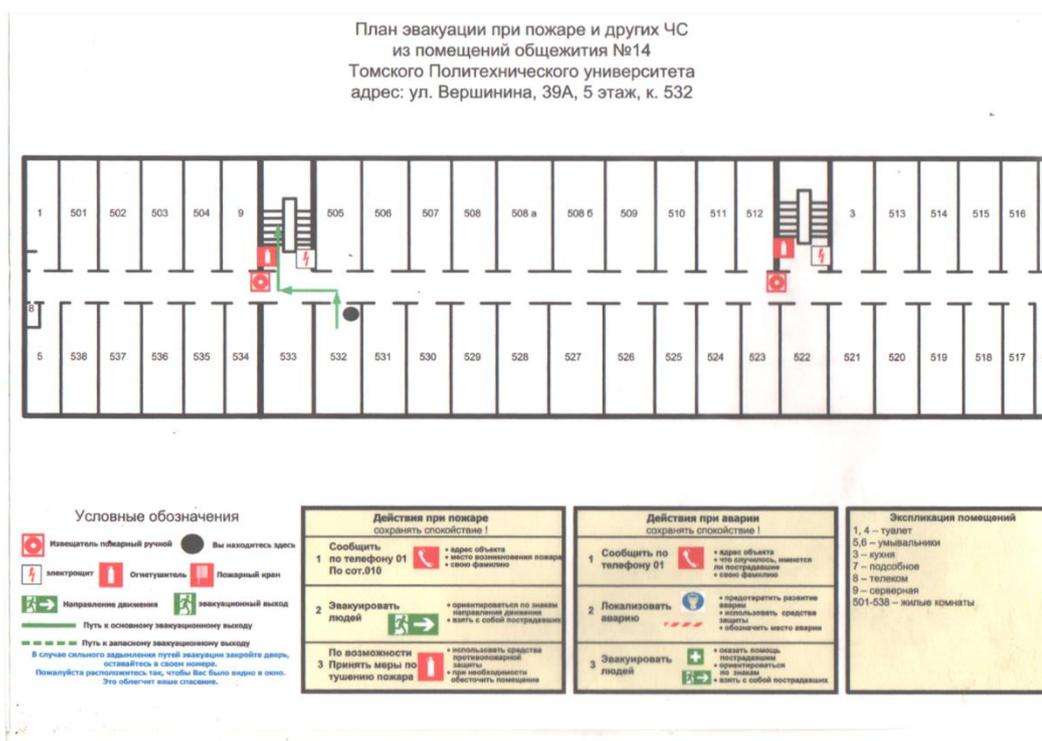


Рисунок 21 – План эвакуации при пожаре

7.4. Правовые и организационные вопросы обеспечения безопасности

7.4.1. Специальные (характерные для проектируемой рабочей зоны) правовые нормы трудового законодательства

Законодательством РФ регулируются отношения между организацией и работниками, касающиеся оплаты труда, трудового распорядка, социальных отношений, особенности регулирования труда женщин, детей, людей с ограниченными способностями и др.

Продолжительность рабочего дня не должна превышать 40 часов в неделю. Для работников до 16 лет – не более 24 часов в неделю, от 16 до 18 лет – не более 35 часов, как и для инвалидов I и II группы. Для работников,

работающих на местах, отнесенных к вредным условиям труда 3 и 4 степени – не более 36 часов.

При работе в ночное время продолжительность рабочей смены на один час меньше. К работе в ночные смены не допускаются беременные женщины; работники, не достигшие возраста 18 лет; женщины, имеющие детей в возрасте до трех лет, инвалиды, работники, имеющие детей-инвалидов, а также работники, осуществляющие уход за больными членами их семей в соответствии с медицинским заключением, матери и отцы – одиночки детей до пяти лет.

Организация обязана предоставлять ежегодные отпуска продолжительностью 28 календарных дней. Для работников, занятых на работах с опасными или вредными условиями, предусматривается дополнительный отпуск.

Работнику в течение рабочего дня должен предоставляться перерыв не более двух часов и не менее 30 минут, который в рабочее время не включается. Всем работникам предоставляются выходные дни, работа в выходные дни производится только с посменного согласия работника.

Организация выплачивает заработную плату работникам. Возможно удержание заработной платы, в случаях, предусмотренных ТК РФ ст. 137. В случае задержки заработной платы более чем на 15 дней работник имеет право приостановить работу, письменно уведомив работодателя.

Законодательством РФ запрещены дискриминация по любым признакам, а также принудительный труд [15].

7.4.2. Организационные мероприятия при компоновке рабочей зоны

При работе за компьютером необходимо соблюдать правильную позу и положение рук – это исключает нарушения в опорно-двигательном аппарате.

При размещении рабочих мест с ПЭВМ расстояние между рабочими столами с видеомониторами, должно быть не менее 2,0 м, а расстояние между боковыми поверхностями видеомониторов – не менее 1,2 м. Экран

видеомонитора должен находиться от глаз пользователя на расстоянии 600 - 700 мм, но не ближе 500 мм с учетом размеров алфавитно-цифровых знаков и символов.

Конструкция рабочего стола должна обеспечивать допустимое размещение на рабочей поверхности используемого оборудования с учетом его количества и конструктивных особенностей, характера выполняемой работы. Высота рабочей поверхности стола должна регулироваться в пределах 680 - 800 мм; при отсутствии такой возможности высота рабочей поверхности стола должна составлять 725 мм.

Конструкция рабочего стула (кресла) должна обеспечивать поддержание рациональной рабочей позы при работе на ПЭВМ, позволять изменять позу с целью снижения статического напряжения мышц шейно-плечевой области и спины для предупреждения развития утомления. Тип рабочего стула (кресла) следует выбирать с учетом роста пользователя, характера и продолжительности работы с ПЭВМ.

Рабочий стул (кресло) должен быть подъемно-поворотным, регулируемым по высоте и углам наклона сиденья и спинки, а также расстоянию спинки от переднего края сиденья, при этом регулировка каждого параметра должна быть независимой, легко осуществляемой и иметь надежную фиксацию.

Время непрерывной работы профессиональных пользователей ПЭВМ ограничивается регламентированными перерывами, общее время работы не более 6 часов. Согласно Р 2.2.2006 – 05. Тогда продолжительность работы с ПЭВМ при вводе и редактировании данных, чтении с экрана не должна превышать 4 часа при 8 часовом рабочем дне. Через каждый час работы должен делаться перерыв на 5-10 мин, а через 2 часа – перерыв на 15 мин [10].

ЗАКЛЮЧЕНИЕ

В результате выполнения бакалаврской работы разработан модуль генерации и визуализации 3D ландшафта с использованием функционала игрового графического движка Unity3d.

В отличие от аналогов, внедрение в проект разработанного модуля позволяет создавать большие динамически подгружаемые пространства. Аналоги позволяют лишь создать статичный объект. Также модуль позволяет строить поверхность ландшафта любой сложности. Перспективной задачей может быть модификация модуля для масштабирования генерации до целых планет.

Разработанный модуль является бесплатным. Модуль имеет доступный для других разработчиков API, распространение через Unity Store позволит другим разработчикам скачивать и добавлять его в свои проекты.

В перспективе есть задачи внесения ряда дополнений в модуль, таких как: импорт трехмерных моделей, настройка параметров типа местности, добавление воды.

СПИСОК ЛИТЕРАТУРЫ

1. Hierarchical surface representations using constrained delaunay triangulations [Электронный ресурс] / Andreas Voigtmann, Ludger Becker, and Klaus Hinrichs. URL: https://www.researchgate.net/publication/307381448_Hierarchical_surface_representations_using_constrained_Delauney_triangulations, требуется регистрация. – Загл. с экрана. – Яз. англ. Дата обращения: 05.06.2017 г.

2. Large Scale Terrain Visualization Using The Restricted Quadtree Triangulation [Электронный ресурс] / R. Pajarola. – URL: <ftp://ftp.inf.ethz.ch/doc/papers/ti/grpw/Vis98.pdf>, свободный. – Загл. с экрана. – Яз. англ. Дата обращения: 05.06.2017 г.

3. Overview of Quadtree-based Terrain Triangulation and Visualization [Электронный ресурс] / R. Pajarola. – URL: <https://pdfs.semanticscholar.org/ae3d/922cb5ac1eef9bab178e0a976b5eced069dd.pdf>, свободный. – Загл. с экрана. – Яз. англ. Дата обращения: 05.06.2017 г.

4. Generating Oriented Pointsets From Redundant Depth Maps Using Restricted Quadtrees [Электронный ресурс] / M. Rothermel, N. Haala, D. Fritsch. – URL: <http://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XL-3/281/2014/isprsarchives-XL-3-281-2014.pdf>, свободный. – Загл. с экрана. – Яз. англ. Дата обращения: 05.06.2017 г.

5. Динамическая оптимизация ландшафта на базе преобразования Хаара и квадродерева вершин [Электронный ресурс] / Юсов Е., Турлапов В. – URL: http://graphicon.ru/html/2006/proceedings/papers/fr14_36_YusovTurlapov.pdf, свободный (05.06.2017). – Загл. с экрана. – Яз. рус. Дата обращения: 05.06.2017 г.

6. Виды трехмерной графики. Полигональная и фрактальная трехмерные графики [Электронный ресурс] / Гасов В.М., Михеев В.А. – URL:

- http://iu5.bmstu.ru/pluginfile.php/644/mod_page/content/7/fractal.pdf, свободный.
– Загл. с экрана. – Яз. рус. Дата обращения: 05.06.2017 г.
7. Трехмерное моделирование [Электронный ресурс]. – URL: <http://www.russika.ru/ef.php?s=5489>, свободный. – Загл. с экрана. – Яз. рус. Дата обращения: 05.06.2017 г.
8. Полный обзор Unity 5 [Электронный ресурс]. – URL: <http://devgam.com/polnyj-obzor-unity-5>, свободный. – Загл. с экрана. – Яз. рус. Дата обращения: 05.06.2017 г.
9. Руководство Unity [Электронный ресурс]. – URL: <https://docs.unity3d.com/ru/530/Manual/UnityManual.html>, свободный. – Загл. с экрана. – Яз. рус. Дата обращения: 05.06.2017 г.
10. СанПиН 2.2.2/2.4.1340 – 03. Санитарно – эпидемиологические правила и нормативы «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы». – М.: Госкомсанэпиднадзор, 2003.
11. Гигиенические требования к естественному, искусственному и совмещенному освещению жилых и общественных зданий. СанПиН 2.2.1/2.1.1.1278-03
12. СНИП 21 – 01 – 97. Пожарная безопасность зданий и сооружений. М.: Гострой России, 1997. – с.12.
13. ГОСТ 17.4.3.04-85. Охрана природы. Почвы. Общие требования к контролю и охране от загрязнения.
14. ППБ 01–03. Правила пожарной безопасности в Российской Федерации. – М.: Министерство Российской Федерации по делам гражданской обороны, чрезвычайным ситуациям и ликвидации последствий стихийных бедствий, 2003.
15. Трудовой кодекс РФ на 2012 год – перераб. и доп. – М.; Рид Групп, 2012.
16. СанПиН 2.2.1/2.1.1.1200-03 Санитарно-защитные зоны и санитарная классификация предприятий, сооружений и иных объектов. – М.: Госкомсанэпиднадзор России, 2003

