Федеральное государственное автономное

образовательное учреждение высшего образования

«Волгоградский государственный университет»

институт Математики и информационных технологий

кафедра Информационных систем и компьютерного моделирования

УТВЕРЖДАЮ

Зав. каф. ИСКМ

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.В. Хоперсков

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_\_\_ г.

**ОТЧЕТ**

**о прохождении производственной практики, преддипломной практики**

|  |  |  |  |
| --- | --- | --- | --- |
| Студент | Королев Артемий Сергеевич | |  |
| Направление подготовки | 09.03.04 Программная инженерия | | |
| Группа | ПРИ-171 |  | |
| Руководитель практики | Г. С. Иванченко | к.ф.-м.н., доцент каф. ИСКМ | |
| Ответственный за организацию практики | С.О. Зубович | к.ф.-м.н., ведущий специалист отдела аналитики и тестирования Волгоградского филиала ООО «Миго-групп» | |
| Место прохождения практики | ФГАОУ ВО «ВолГУ», каф. ИСКМ | | |
| Сроки прохождения практики | «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_ 20\_\_\_ г. | | «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_ 20\_\_\_ г. |

Волгоград 2021

**Содержание**

Введение 3

1 Работа с данными и представление их в визуальной форме 5

1.1 Методы визуализации многомерных информационных данных 6

1.2 Применение диаграмм, как метод для визуализации данных 8

1.3 Другие различные методы работы и визуализации данных 13

2 Географическая информационная система – возникновение, работа с ней и сопутствующие виды 14

2.1 История развития и возникновения геоинформационной системы 14

2.2 Визуализации данных в геоинформационной системе, актуальность и классификация 14

2.3 Технологии дистанционного зондирования поверхности Земли из космоса 16

3 Практическая часть – разработка программного обеспечения для работы с данными геоинформационной системы 19

3.1 Информационная модель программы, её реализованные функции 20

3.2 Формат исходных подаваемых данных GRD и её визуализация в рабочей области 27

3.3 Функции приложения и их реализация в программном коде 30

3.4 Тестирование реализованного приложения, применение на практике 33

Заключение 40

Литература 41

Приложение А Полученные при выполнении работы компетенции 45

# Введение

В настоящее время информационный ресурс – один из главных атрибутов общества. Каждый день человек получает огромное количество информации, но обрабатывать и анализировать весь объём – невыполнимая задача. В этом может помочь визуализация – средство наглядного представления абстрактной информации. Не смотря на множество способов визуального представления данных и их обработки, нет единого решения для каждой работы, а потому большой процент точности представления информации зависим от выбранного способа визуализации необходимых для обработки данных.

Визуализация пространственных данных используется в первую очередь для задач научной визуализации [1]. Под научной визуализацией понимается создание графических изображений и моделей как для наглядного изучения принципов работы исследуемого процесса с разных аспектов, так и для иллюстрации результатов исследований. При этом большой объем результатов моделирования представляется в компактной и легко воспринимаемой форме. Репрезентация изучаемых систем и процессов в графической иллюстрации позволяет наблюдателям осмотреть проект изнутри, что было бы невозможно без визуализации данных. И, часто, только благодаря правильной визуализации, люди могут полностью понять особенности исследуемого явления [2].

Целью данной работы является изучение основ визуализации, данных цифровой модели рельефа, а также результатов моделирования. Создать информационную модель программы, разработка начатой в предыдущем семестре. Выявить самые нужные и полезные функции и доработки программы. Понять принцип и устройство этих функций.

В качестве входных данных также использованы файлы формата GRD, сохранённый в формате сетки, используемой для создания карт. Исходный формат файла содержит XYZ-данные, которые используются в ГИС-приложениях и часто используются для задач математического моделирования.

# 1 Работа с данными и представление их в визуальной форме

Визуализация – общее название способов представления числовой информации или физического явления в виде, удобном для зрительного наблюдения и анализа [14].

Визуализация информации – это процесс отражения отвлечённых научных или же деловых данных в форме изображений, которые содействуют в усвоении значения данных или же зрительное представление объединения дискретных данных [3]. На сегодняшний день известно три вида визуализации:

* статическая;
* динамическая;
* интерактивная.

Наиболее актуальной сферой визуализации является научная. 3Д-представление позволяет спроектировать не только единичный объект, но и группу различных предметов и строений. Самым действенным способом презентации учебного материала является наглядность, благодаря которой объясняются некоторые явления в разных сферах жизни, к примеру, процесс расщепления атома. Также, визуализация помогает смоделировать объекты и события, которые мы уже не увидим в существующей реальности – развитие жизни на планете до нашей эры [4].

На данный момент популярна 3D-визуализация. С помощью особого программного обеспечения можно визуализировать такие картографические материалы, как: спутниковые снимки, результаты аэро- и фотосъемки, цифровые модели рельефа, и другие геоданные [7] [20].

## Методы визуализации многомерных информационных данных

В настоящий момент объём и сложность информации продолжают расти и накапливаться. Достаточно часто структурные единицы данных представляют набор из множества независимых признаков:

1. Количественные признаки:
   1. непрерывно;
   2. дискретно.
2. Качественные признаки:
   1. дискретно и неупорядоченно;
   2. дискретно и содержит немного элементов.

Как результат обработки данных различными способами визуализации получается трёхмерное изображение выводимой информации. Визуализация многомерных данных, как правило, строится на основе модели для двумерных данных, а нижеперечисленные визуальные измерения отражают признаки объекта:

1. Положения в пространстве;
2. Цвет. Сильно влияет на визуализацию, делая её более понятной. Цветовая слепота или дальтонизм могут сказаться на предыдущем признаке, делая различие цветов невозможным (рисунок 1);

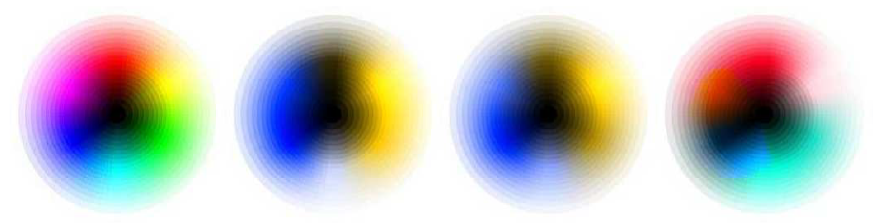


Рисунок 1 – Отличия восприятия цветов между оригинальным изображением (слева) и результатом цветов, получаемых у людей, страдающих различными формами дальтонизма [10]

1. Цветовое перенасыщение. Считается плохим тоном перенасыщать изображения яркими цветами, так как это сильно влияет на понимание происходящего.
2. Форма. Улучшают визуализацию также добавление различных фигур, диаграмм и так далее. В тоже время не допускается нагромождение такими формами.
3. Размер. Элементы и формы из предыдущего пункта стоит подразделять по размерам, учитывая важность, нужность и количество повторений.
4. Узор. Часто употребляется для отображения больших областей на диаграммах, картах и изображений как таковых.

Стоит понимать, как изображать диаграммы для создания наглядности. Неверные диаграммы будут вводить в заблуждения людей и усложнять работу с данными, как на рисунке 2. А так как существует достаточное количество методов визуализации, необходимо хорошо разбираться и в них. Подробнее будет рассказано о них далее [30].



Рисунок 2 – Пример некорректной диаграммы. Монотонный характер «зависимости» может создать иллюзию корреляции между числом пиратов и средней температурой

## Применение диаграмм, как метод для визуализации данных

Диаграммы, по большей части, строятся по двумерным данным и относятся к видам [15]:

1. массив. Чаще всего используется двумерный массив, хранящий координаты по оси и значение самой координаты. Или как в наших grd-файлах, трёхмерный массив – координаты по X и Y и высота над уровнем моря.
2. Граф. Он может состоять из:
   1. матрица смежности. Применяется во взвешенных графах.
   2. Матрица инцидентности. Направленные графы.
   3. Список рёбер.
3. Карта. В отличии от остальных, содержит географические координаты объекта.

Самый популярный вид диаграмм – график (рисунки 4-7). В свою же очередь, диаграммы подразделяются по количеству осей:

1. Диаграммы не имеющие оси. Treemap – одна из таких диаграмм (рисунок 3). Элемент диаграммы – клетка, имеющая в себе три измерения: надпись, размер, цвет. Клетки располагаются по убыванию размера сверху вниз, слева направо. Для улучшения визуализации этой диаграммы обычно добавляют штриховку и надписи в качестве символьных обозначений.

Данный класс диаграмм, для удобства использования, систематизированы по способу представления:

* точечная диаграмма. Точечные диаграммы используются для отображения степени зависимости переменных друг от друга через размещённые на плоскости координат точки данных. Строки в таблице данных представлены маркером и каждая из них заданы на осях X и Y. Позиция маркера зависит от значений в столбцах.
* График. Точечная диаграмма со связанными точками. Позволяет отслеживать динамику изменения данных.
* Диаграммы-области. Диаграммы с областями используют для отображения изменения количественных значений в заданном интервале или же за определённый период времени. Зачастую, этот тип диаграмм применяют при отражении тенденций, но не конкретных значений. Линейные диаграммы обладают схожими свойствами.

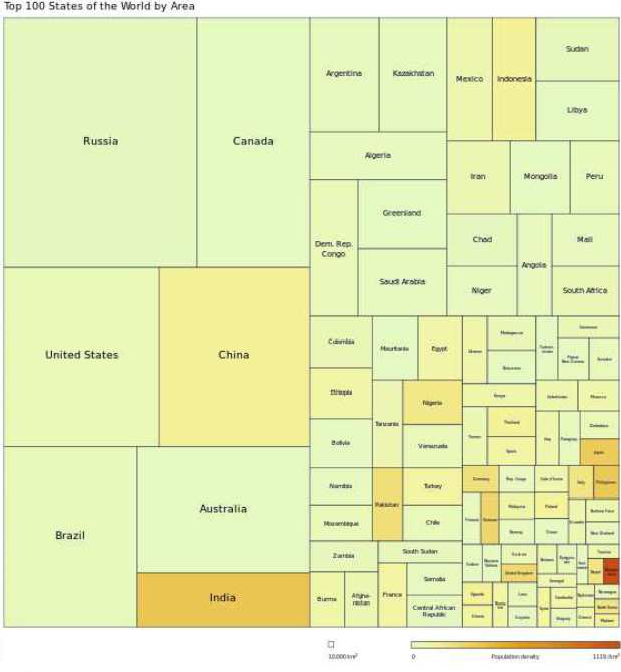


Рисунок 3 – Пример диаграммы treemap. Распределение плотности населения на сто стран, занимающих наибольшую площадь [25]

1. Диаграммы с одной осью. Круговая диаграмма – диаграмма с одной осью, отлично демонстрирующая процентное содержание компонентов. Диаграмма, где одна ось отражает количественный признак, а другая – качественный, называется столбчатой или bar charts. Тогда визуальными показателями являются высота столбца и его цвет.
2. Диаграммы с двумя осями. Категория включает в себя большое количество разного рода диаграмм, которые сгруппировали по типам признаков, связанных с осями. Этого рода диаграммы очень схожи с таблицами. У них высокий потенциал визуализации многомерных данных, при условии, что все ячейки таблицы будут содержать вложенные диаграммы.

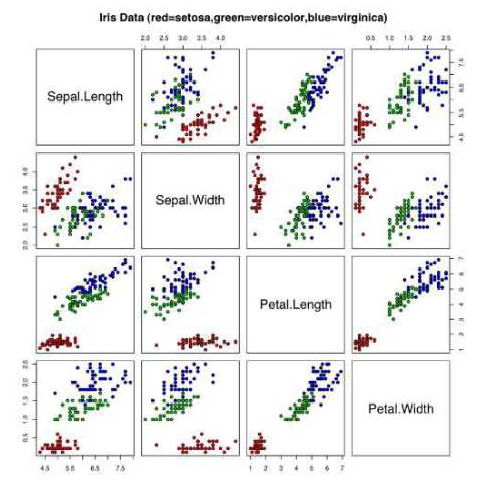


Рисунок 4 – Пример круговых и точечных диаграмм

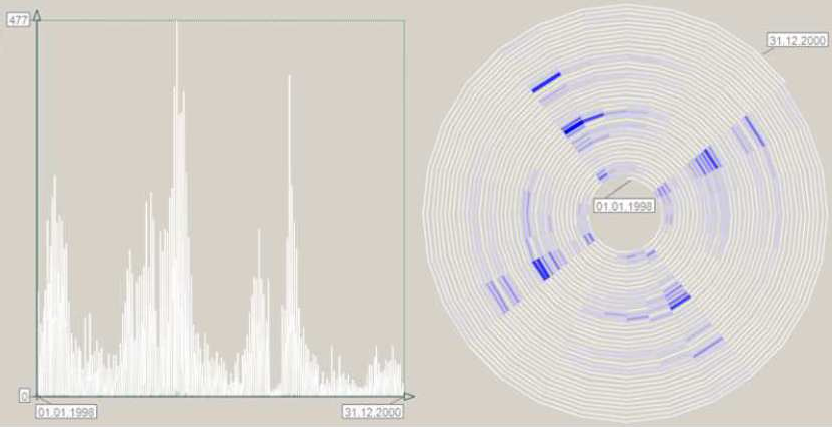


Рисунок 5 – Пример циклического паттерна [10]

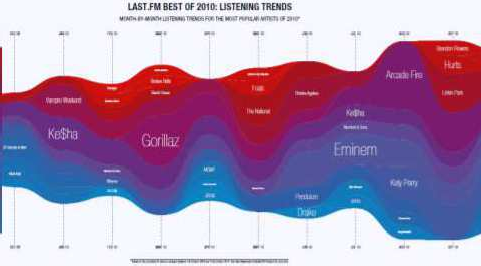


Рисунок 6 – Пример streamgraph. Исполнители на одном из музыкальных сервисов [18]

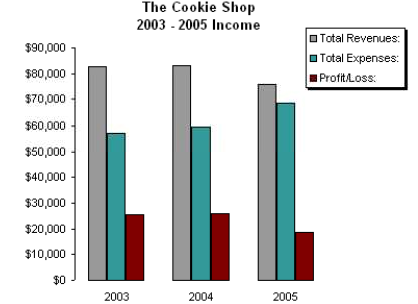


Рисунок 7 – Пример столбчатой диаграммы

Трёхмерные диаграммы, такие как точечные, столбчатые и трёхмерные графики представляются в изометрической проекции или в перспективе [8].

Радиальные диаграммы (рисунок 8) позволяют проводить сравнение между множеством переменных и позволяют иметь неограниченное количество осей.

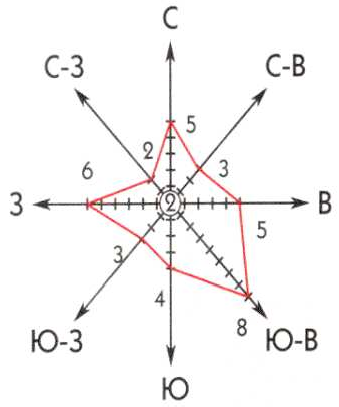


Рисунок 8 – Один из примеров радиальной диаграммы (роза ветров)

Диаграммы на основе графов:

* + 1. Сеть.На практике, сеть хорошо обозначает признаки при объединении её с другими диаграммами. Примером такого объединения может быть представление вершин графа в виде круговых диаграмм (рисунок 9).

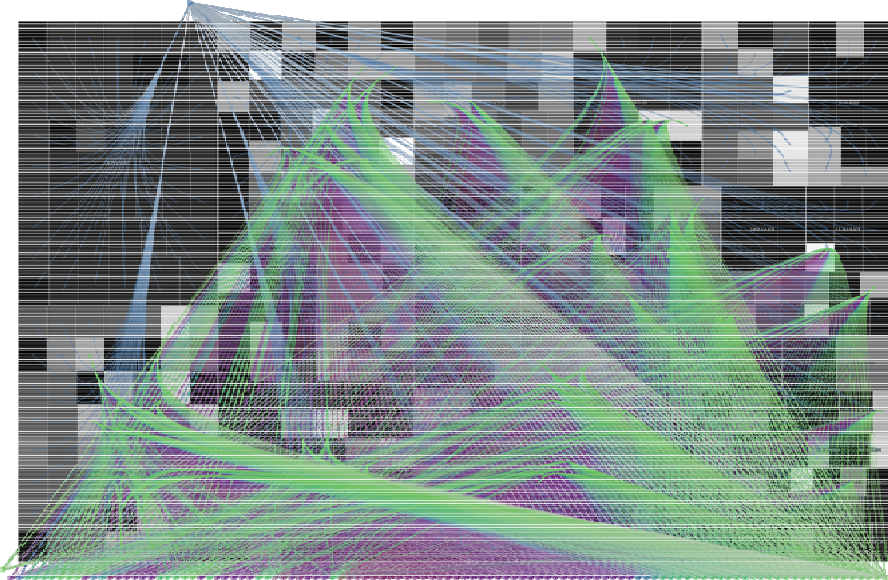


Рисунок 9 – Пример сети [10]

* + 1. Диаграмма Эйлера-Венна (рисунок 10). Такой вид диаграммы больше подходит для представления бинарных характеристик. В первоначальном варианте диаграммы предполагалось три множества, однако, сейчас появились виды с четырьмя и с пятью множествами [11].



Рисунок 10 – Диаграмма Эйлера-Венна с тремя, четырьмя и пятью множествами [10]

1. Картограмма. Такой вид диаграмм можно определить к узконаправленным в сфере способов визуализации. Картограммы схожи с точечными диаграммами, но, в отличие от них, объекты в них привязаны к конкретным географическим положениям (рисунок 11).

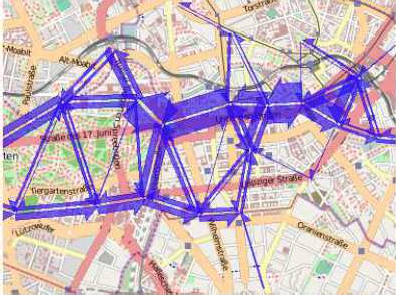


Рисунок 11 – Пример картограммы

Как раз картограммы мы и будем изучать в данной научно-исследовательской работе. Что такое визуализация и визуализация именно в ГИС?

## Другие различные методы работы и визуализации данных

# 2 Географическая информационная система – возникновение, работа с ней и сопутствующие виды

## История развития и возникновения геоинформационной системы

## Визуализации данных в геоинформационной системе, актуальность и классификация

Актуальность использования 3Д модели местности связана в первую очередь с современными навигационными устройствами, позволяющие уверенно ориентироваться даже на незнакомой территории. Так же построение электронной карты позволяет обеспечивать наглядное представление об окружающей обстановке в реальном времени и в любой точке нашей планеты.

Трехмерные ГИС позволяют создавать модели любой существующей инфраструктуры, а также создавать и сравнивать реалистичные варианты развития территорий.

С помощью визуализации данных в ГИС, пользователю представляется возможность создания и обработки картографической информации. Одной такой возможности будет демонстрация на экране монитора некоторых тематических карт для их общего визуального анализа, так же, плюс электронных карт служит возможность масштабирования с автоматической генерацией карт (рисунок 12).

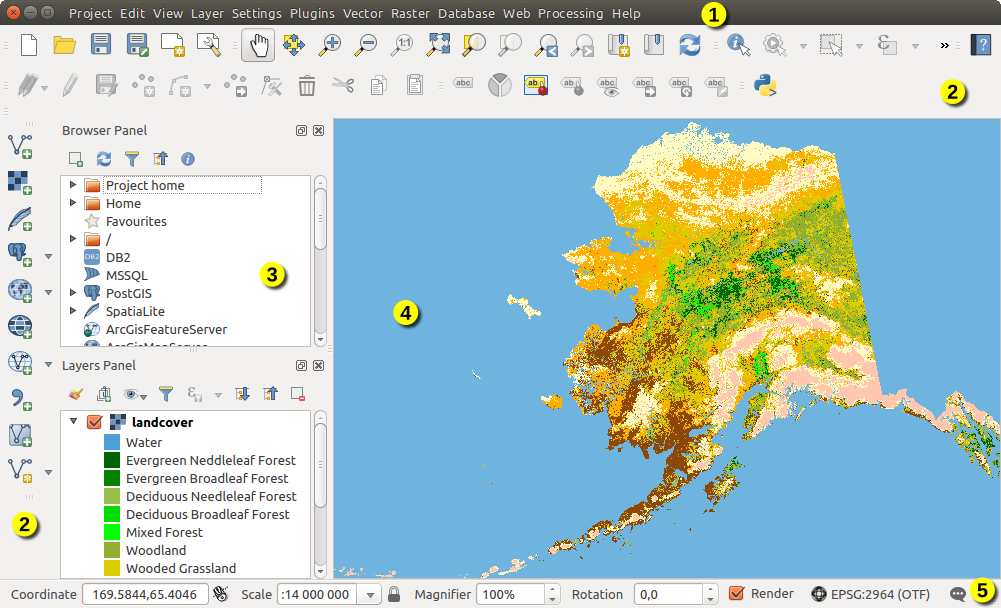


Рисунок 12 – Визуализация карт на примере программы QGIS [10]

Среди моделей визуализации карт существует классификация (рисунок 13):

* векторная модель;
* растровая модель;
* топологическая модель.

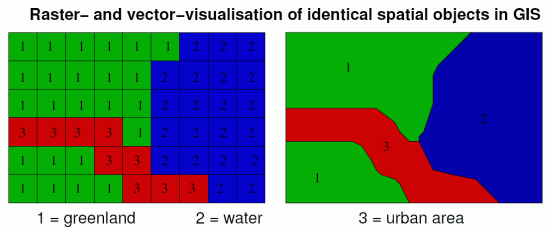


Рисунок 13 – Пример растровой и векторной модели визуализации объектов

Векторный план – это набор данных, представленных в цифровом формате, состоящий из объектов, которые определяются путем оцифровки растровых данных (рек, лесов, дорог и других географических объектов). То есть растровые карты являются источником геопространственных данных в производстве векторных карт [13].

Растровое изображение – это самый простой тип карты, который получается путем сканирования бумажного варианта изображения и дальнейшего проецирования на земную поверхность. Растровые карты являются всего лишь изображением карты [5].

Разновидностью векторных моделей считаются топологические модели. Топологические модели сводит «взаимосвязанность» объектов, которая бывает простой или же сложной. Топологические качества фигур не меняются при деформациях, выполняемых без разрывов или соединений. Термин топологический в ГИС понимают так, собственно, что в моделях объектов сберегаются взаимосвязи, которые расширяют способности применения данных ГИС для всевозможных видов пространственного анализа. К примеру, в закономерную структуру описания данных вводится информация о том, какие линии и в каких точках пересекаются, из чего произведён полигон и другие.

## Технологии дистанционного зондирования поверхности Земли из космоса

Что такое дистанционное зондирование Земли? ДЗЗ – это наблюдение за поверхностью Земли наземным, авиационным и космическими средствами, оснащёнными различными видами съемочной аппаратуры [6] [17].

Методы зондирования могут быть:

1. Пассивные – с использованием естественного отражённого или вторичного теплового излучения объектов на поверхности Земли, обусловленного солнечной активностью.
2. Активные – использующие вынужденное излучение объектов, инициированное искусственным источником направленного действия.

ДЗЗ, полученные с космического аппарата (КА), характеризуются большой степенью зависимости от прозрачности атмосферы. Поэтому на КА используется многоканальное оборудование пассивного и активного типов, регистрирующие электромагнитное излучение в различных диапазонах [9].

Первым из космических аппаратов, используемых для ДЗЗ, был Landsat. Программа Landsat – один из самых долгих проектов для получения снимков Земли. Первый спутник был запущен в 1972 году [24].

Landsat-8 произведён компанией Orbital Sciences Corporation по контракту от NASA на базе собственной платформы Orbital LEOSt. Для оптимизации точности определения ориентации спутника используются три высокоточных астродатчика (ar-3, два из которых работают в активном режиме), масштабируемая инерциальная система наведения SIRU (Scalable Inertial Reference Unit), приемники системы GPS и два трехосных магнетометра. Усовершенствование технических характеристик целевых приборов способствует уменьшению уровня радиометрических искажений по сравнению с приборами на предыдущих спутниках серии Landsat, а использование более совершенных ПЗС-устройств позволяет улучшить соотношение сигнал-шум и качество съемки [26]. Приборы измеряют 4096 различных уровней отраженного света, в то время как бортовая камера ETMP на спутник [Landsat-7](https://innoter.com/sputniki/landsat-7/) могла измерить 256 различных уровней [12]. Пример космоснимка со спутника Landsat-8 можно увидеть на рисунке 14.



Рисунок 14 – Космоснимок со спутника Landsat-8 прибрежных территории вблизи города Малибу.

# 3 Практическая часть – разработка программного обеспечения для работы с данными геоинформационной системы

Разработка программного обеспечения (ПО) для визуализации велась на объектно-ориентированном языке программирования С#. Он был выбран так как является достаточно простым и мощным языком для программирования, который позволяет создавать многофункциональные приложения, а из-за большого количества синтактических конструкций и возможности работать с платформой .NET, С# позволяет быстрее разрабатывать программные решения, чем любой язык программирования [19] [21].

Microsoft Visual Studio [34] [35] – это узкоспециализированная линейка программных продуктов, которая включает в себя интегрированную среду разработки (рисунок 15). Visual Studio ориентированно на разработку приложений. Приложения могут поддерживать графический интерфейс, так и быть консольным приложением. Visual Studio имеет редактор исходного кода с поддержкой технологии IntelliSence [31], который помогает редактировать исходный код, вносить правки не теряя при этом продуманность изменений [16].

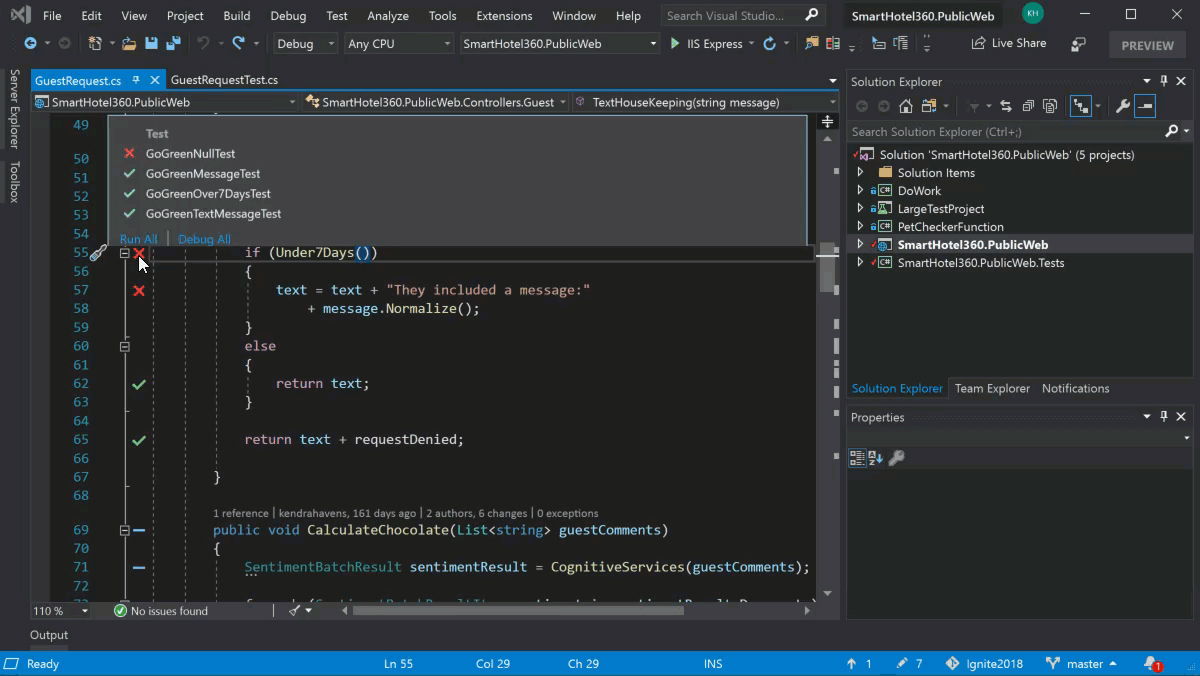


Рисунок 15 – Microsoft Visual Studio

## 3.1 Информационная модель программы, её реализованные функции

На рисунке 16 изображена информационная модель нашего приложения [27] [28].



Рисунок 16 – Информационная модель приложения

Прямыми линиями показана программа, реализованная на данный момент, пунктирными линиями – то, что находится в разработке и должно быть продемонстрировано в дипломной работе.

А именно, настройка нашего приложения, состоящая из нескольких пунктов:

1. Изменение цветовой гаммы

Изменение цветовой гаммы (рисунок 17) связано с тем, что каждому пользователю необходимо или хочется видеть изображение в определённом цвете, сделать элемент на экране более видимым или придать больше значения какой-то части изображения.

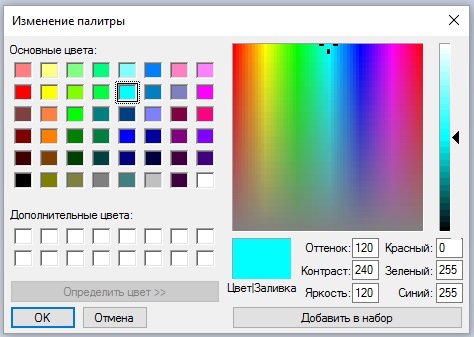


Рисунок 17 – Пример изменения цветовой палитры

1. Изменение шага цвета градиента

Под изменением шага цвета градиента подразумевается изменение количества цветов градиента изображения, для более подробной и качественной визуализации. Например, градиент белого и чёрного может представлен разными способами, может быть только 2 цвета (белый и чёрный соответственно), либо большим набором цветов, как это показано на рисунке 18, ниже.

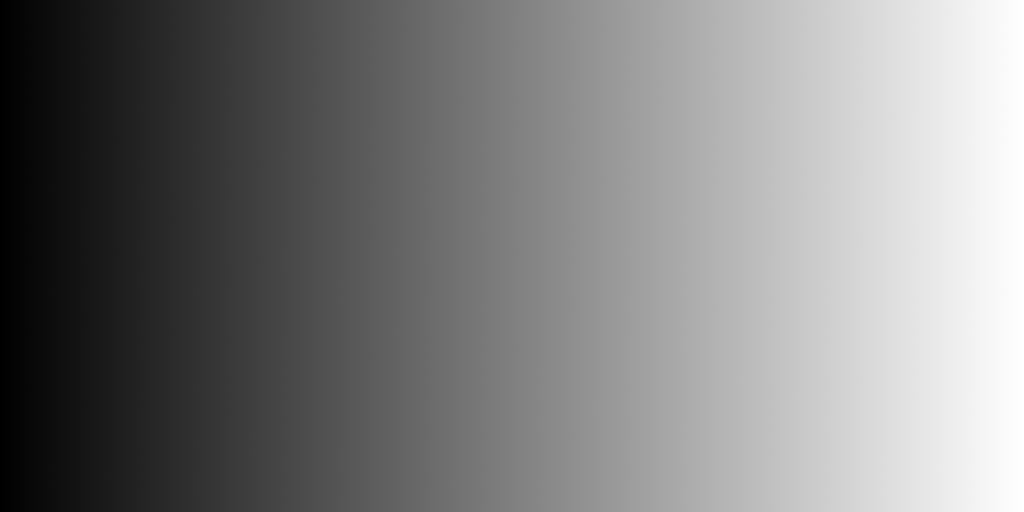


Рисунок 18 – Градиент от чёрного к белому

В дальнейшем будет добавлены новые функции, упрощающие работу с программой и делающей её более удобной для использования.

1. Добавление слоя

Добавление слоя – одна из важных функций в будущем приложении. С этим нововведением мы можем получать более полную картину визуализированной области. Подобный вид полученной функции можно наблюдать на 25 рисунке.

1. Отображение скрытых слоёв

Функция отображения слоёв, как на рисунке 19, означает, что при присутствии более одного слоя имеется возможность включать визуализацию того или иного слоя [32].

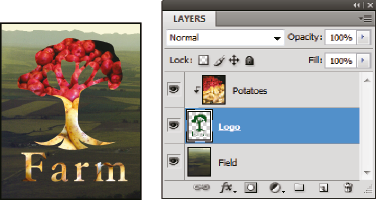


Рисунок 19 – Отображение слоёв в Photoshop

1. Таблица атрибутов у слоя

У слоя есть свои атрибуты, самое простое – значения X, Y и Z в этих координатах. Также объекты могут обладать различными характеристиками, такими как цвет, год, количество и другие. Пример таблицы атрибутов – рисунок 20.

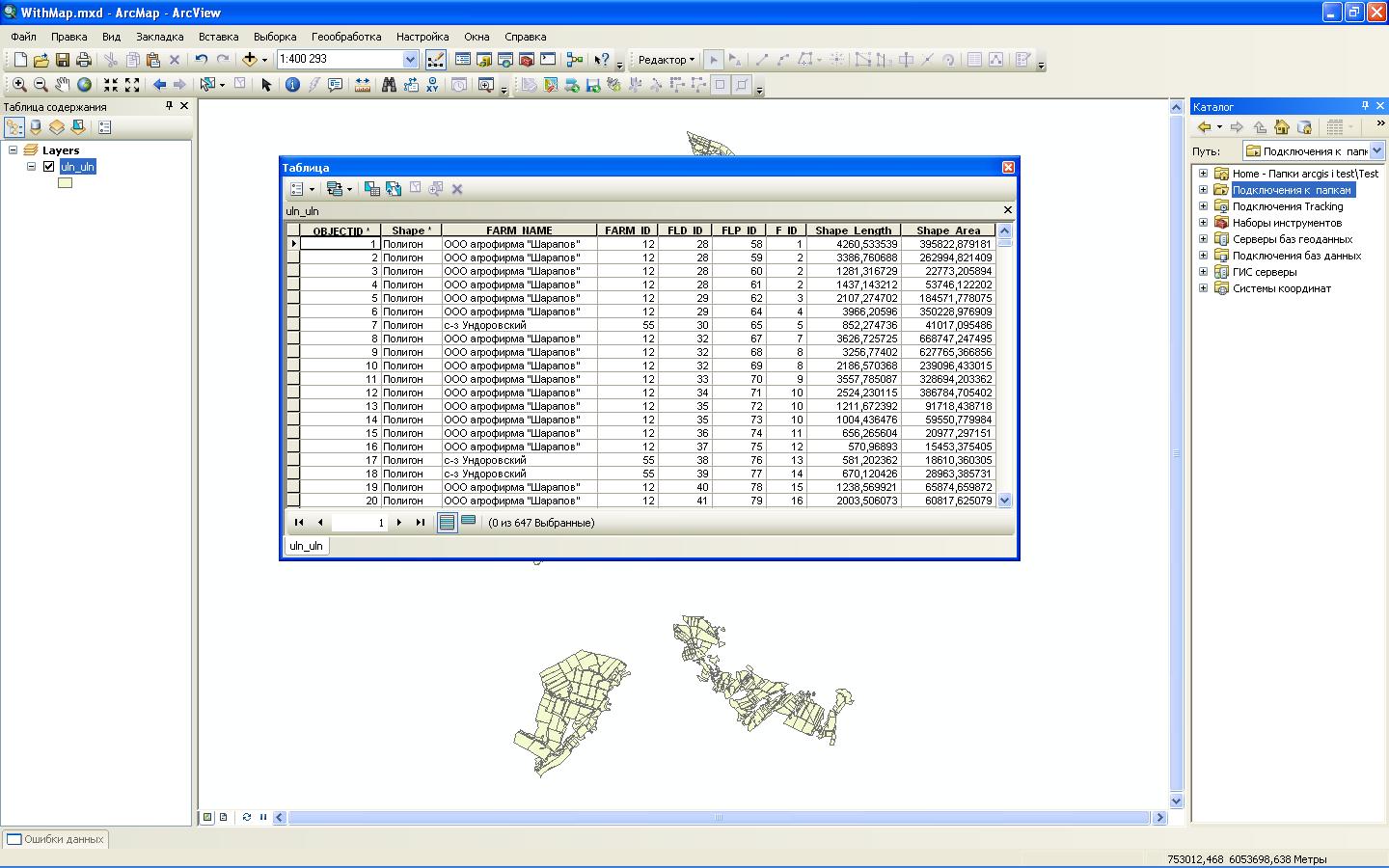


Рисунок 20 – Таблица атрибутов у слоя в ArcGIS

1. Комбинирование слоёв

Под комбинированием слоёв понимается функция взаимодействия слоёв между друг другом, один слой выше другого. Например, слой рельефа не должен быть выше слоя с строениями.

На рисунке 21 изображена карта Москвы, на ней можно отчётливо видеть, что слой дорог находится выше слоя рельефа. В ином случае, дорог не было бы видно.

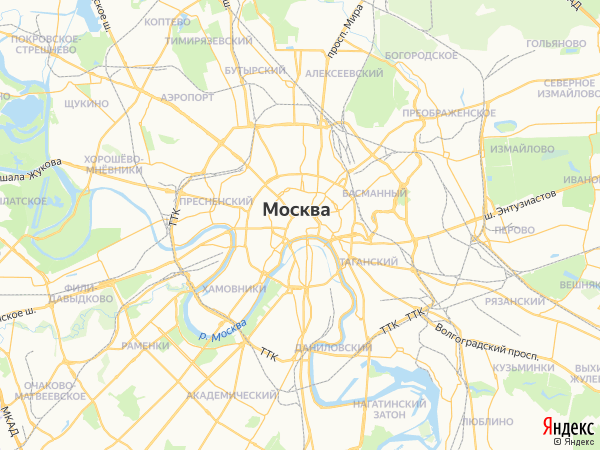


Рисунок 21 – Карта Москвы

1. Прозрачность слоёв

Прозрачность слоёв нужна в некоторых случаях для того, чтобы видеть, что находится под верхним слоем.

1. Инструменты рисования, добавления текста

Инструменты рисования и добавления текста (рисунок 22) позволят добавлять на визуализированную область свои надписи и фигуры. Они будут видны на последующем сохранённом изображении.



Рисунок 22 – Инструменты рисования на примере приложения Paint

1. Легенда

Легенда карты – это некоторый список или таблица условных обозначений на карте с разъяснением их обозначения. По большей части в ней будет обозначения высот и их цвета в зависимости от их значений. Легенда карты будет размещена в углу приложения, либо в отдельном окне, как показано на примере рисунка 23.

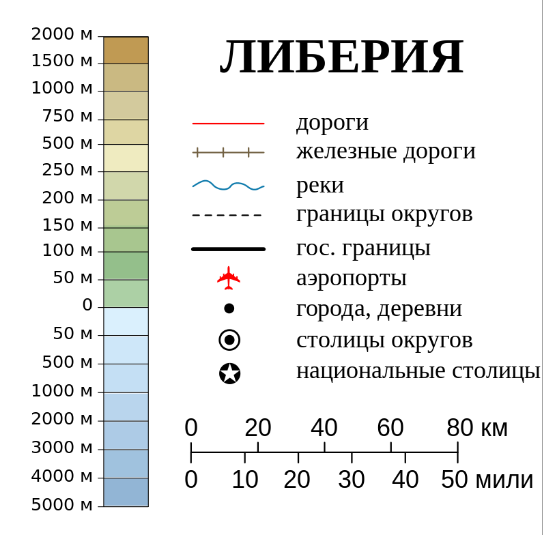


Рисунок 23 – Легенда карты Либерия

1. Создание 3D модели

3D модель – модель рельефа (рисунок 24), построенная в трёх плоскостях, служит для более чёткого представления визуализации [29].

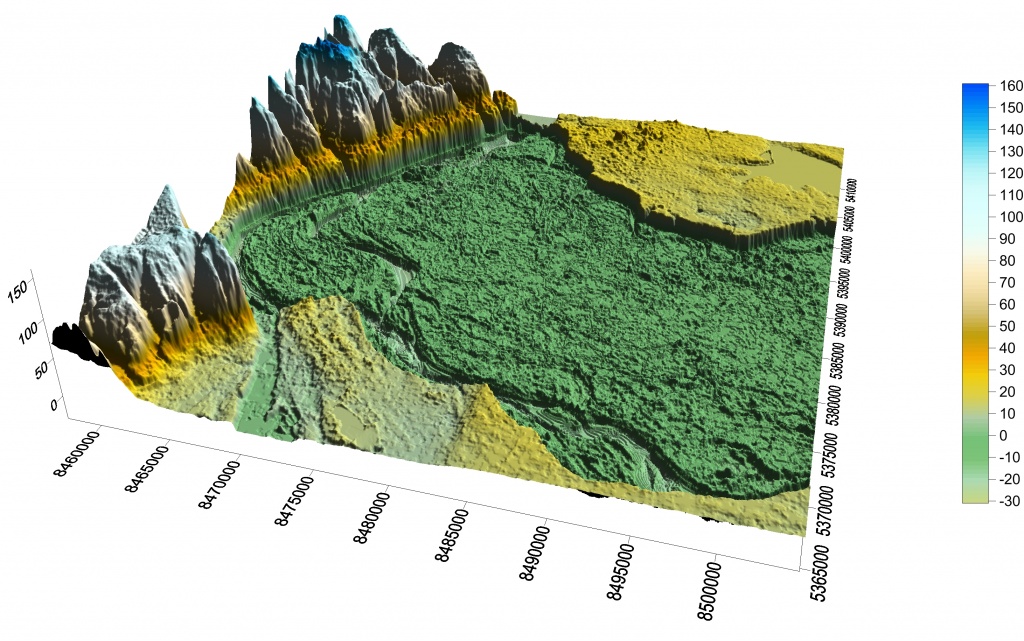


Рисунок 24 – Пример трехмерной цифровой модели рельефа и местности

Создание 3D модели – самая сложная в реализации функция программы. Наличие её в будущей версии до конца не определено.

Структура GRD файла позволяет создавать 3D модели, так как для неё есть все 3 типа нужных данных: значения X и Y, а также значения высот Z по всей прямоугольной области плоскости (x; y). Дополнительными функциями при построении 3D модели могут послужить функция поворота визуализированной модели, приближение и отдаление, а также определённая цветовая гамма, позволяющая видеть высоты.

## 3.2 Формат исходных подаваемых данных GRD и её визуализация в рабочей области

В данной научно-исследовательской работе в качестве исходных данных послужили файлы формата GRD [23] [33]. Рассмотрим данный формат.

Файлы в таком формате содержат значения в скалярных данных в регулярной прямоугольной сетке в пространстве (x, y) или долготе и широте (longitude, latitude).

Файлы такого формата чаще всего используются в приложении Surfer. Программное обеспечение **Golden Software Surfer** – средство для моделирования и анализа поверхностей, визуализации ландшафта, генерирования сетки, разработки трехмерных карт и многого другого [22].

Первая строка grd-файла имеет три числа: x\_min, d\_x, x\_max (или: lon\_min, d\_lon, lon\_max). Вторая строка также имеет три числа: y\_min, d\_y, y\_max (или: lat\_min, d\_lat, lat\_max).

Если сетка определена в долготе и ширине (longitude, latitude), то стоит обратить внимание на то, что единицы lon\_min, d\_lon, lon\_max, lat\_min, d\_lat, lat\_max являются градусами (например, 24º17'5'' должны быть преобразованы в десятичную дробь 24,2847 градусов). Восточная долгота +, Запад - -, Северная широта +, Юг - -.

Если же сетка определена в (x, y) пространстве, то шесть параметров в верхней части могут быть в различных единицах: в метрах, километрах, сантиметрах, милях или же футах.

Следующие строки дают данные в сетке в текстовом порядке начиная с верхнего левого угла (максимальная y или широта; минимальная x или долгота), идущего слева направо вдоль верхней строки (увеличение долготы или x на постоянной широте или y), затем слева направо вдоль второго ряда и так далее. Количество и положение разрывов строк не имеет значение в этой части файла. Таким образом можно иметь даже одно значение на всю строку. Пользователям легче читать эти файлы, если они имеют вид, демонстрирующий количество столбцов равное 1 + (x\_max - x\_min) / d\_x, как показано на рисунке 25 ниже.

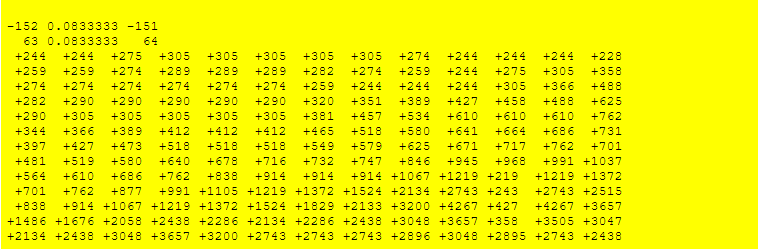


Рисунок 25 – Пример структуры grd-файла

Не существует ограничений на единицы значений данных в сетке, за исключением того, что все они должны быть одинаковыми.

В работе используются единицы СИ (метры для высоты).

В небольшом примере файла .grd, приведённом выше, приведена сетка высот вокруг горы Денали, Аляска, США. Она определена в пространстве (longitude, latitude) (152 ~ 151ºW, 63 ~ 64ºN), а высоты указаны в метрах.

Вершина (+6168 м) расположена в точке (151º00'13'' W, 63º04'10'' N), поэтому она находится между двумя последними записями в крайнем правом столбце. Знаки + на положительных значениях высоты могут быть опущены, но отрицательные знаки для западной долготы имеют решающее значение, так же, как и их порядок в первой строке файла (отрицательные числа слева, положительные – справа).

Таким образом мы рассмотрели построение структуры .gdr файла. Приступим к раскодировке на языке C#.

Создадим функцию ReadFile, принимающую на вход наш файл.

Листинг 1 Функция ReadFile

public void ReadFile(BinaryReader file)

{

file.ReadChars(4);

Nx = (int)file.ReadInt16();

Ny = (int)file.ReadInt16();

Z = new float[Ny, Nx];

XMin = file.ReadDouble();

XMax = file.ReadDouble();

YMin = file.ReadDouble();

YMax = file.ReadDouble();

ZMin = file.ReadDouble();

ZMax = file.ReadDouble();

for (int j = 0; j < Ny; ++j)

for (int i = 0; i < Nx; ++i)

Z[j, i] = file.ReadSingle();

Xstep = (XMax - XMin) / Nx;

Ystep = (YMax - YMin) / Ny;

}

В листинге 1 считываются данные с файла с расширением GRD. Пропускаются 4 буквы в самом начале, они не нужны. После этого считывается количество точек по координате X и Y. Высчитывается Z – количество всех точек на карте.

Принимаются угловые значения XMin – координата X слева, XMax – координата Y справа, YMin – координата Y сверху, YMax – координата Y снизу, ZMin и ZMax – минимальные и максимальные значения высот по всей карте.

Далее через конструкцию цикла for считывается в двумерный массив данные о высотах. Двумерный массив был выбран не просто так. Как было описано выше, для простого пользователя легче всего представление с помощью колонок, структурированно. Также и разработчику будет легче в будущем строить изображение, закрашивая его по строкам.

Следующий листинг 2 отражает закраску карты в элемент pictureBox.

Листинг 2 Функция pictureBox

public void Drawing()

{

int width = this.Width;

int height = this.Height;

int mapWidth = Map.Width;

int mapHeight = Map.Height;

double scale = Math.Min(width \* 1.0 / mapWidth, height \* 1.0 / mapHeight);

int scaleWidth = (int)(mapWidth \* scale);

int scaleHeight = (int)(mapHeight \* scale);

firstSize = pictureBox.Size = new Size(scaleWidth, scaleHeight);

firstLocation = pictureBox.Location = new Point((width - scaleWidth) / 2, (height - scaleHeight) / 2);

pictureBox.Image = Map;

}

В этом части кода высчитывается высоту карты height, а также ширину карты width. Интересная и нужная часть кода в этой функции принадлежит вычислению значения scale. Это значение определяет минимальное значение по ширине и высоте, иначе говоря, определяет, какая сторона меньше. Это нужно для построения карты, закрашивая её в pictureBox меньшей стороной. В ином случае возникла бы проблема, когда карта выходила за границы программы.

## 3.3 Функции приложения и их реализация в программном коде

Значение firstLocation отвечает за первичное положение курсора, увиденного изображения, при загрузке карты.

Листинг 3 Функция положения курсора при загрузке

private void pictureBox\_MouseWheel(object sender, MouseEventArgs e)

{

Graphics g = this.pictureBox.CreateGraphics();

int w = pictureBox.Width;

int h = pictureBox.Height;

Point mouseLocation = e.Location;

Point location = pictureBox.Location;

float scale = 1;

deltaScale += (e.Delta / 480.0F);

if (deltaScale > 2)

deltaScale = 2;

else if (deltaScale < 0)

deltaScale = 0;

else

scale += (e.Delta / 480.0F);

pictureBox.Location = new Point((int)(location.X + mouseLocation.X \* (1 - scale)),

(int)(location.Y + mouseLocation.Y \* (1 - scale)));

pictureBox.Width = (int)(w \* scale);

pictureBox.Height = (int)(h \* scale);

if (pictureBox.Width / firstSize.Width < 1)

{

pictureBox.Size = firstSize;

pictureBox.Location = firstLocation;

}

}

В листинге 3 реализуется управление колёсиком мышью, а именно приближение и отдаление. При увеличении картинки размер содержимого pictureBox увеличивается, а сам pictureBox остаётся неизменным.

Листинг 4 Функция положения курсора при движении

private void pictureBox\_MouseMove(object sender, MouseEventArgs e)

{

if (e.Button == MouseButtons.Left)

{

mouseDownPosition = e.Location;

Point location = pictureBox.Location;

int dx = e.X - mouseDownPosition.X;

int dy = e.Y - mouseDownPosition.Y;

pictureBox.Location = new Point(location.X + dx, location.Y + dy);

}

}

В листинге 4 содержится функция, позволяющая определять позицию мыши на экране для последующего использования этих данных в третьем листинге.

Следующий листинг 5 очень похож на предыдущий с тем лишь отличием, что по нажатию левой кнопки мыши будет выводиться значения координат, а также значение вершин, которые соответствуют нашим координатам.

Листинг 5 Функция вывода значений координат

private void pictureBox\_MouseDown(object sender, MouseEventArgs e)

{

if(e.Button == MouseButtons.Left)

{

int hoverX = (int)(XMin + e.X \* Xstep);

int hoverY = (int)(YMax - e.Y \* Ystep);

float hoverZ = Z[Ny - e.Y, e.X];

label1.Text = «( « + hoverX + « ; « + hoverY + « ; « + hoverZ.ToString() + « )»;

}

}

В элемент label1 будет поступать текст, имеющий следующий вид: («значение X»; «значение Y»; «высота, значение Z»).

Одна из функций программы является преобразование карты из графического интерфейса в изображение формата .png, как показано на 6 листинге.

Листинг 6 Функция создания изображения карты

public Bitmap CreateMap()

{

Color setColor(double z) => z == 0 ? Color.Transparent : HSL.HSL2RGB(new HSL(Color.Green, Lightness(z, 12)));

Bitmap image = new Bitmap(Nx, Ny);

for (int j = 0; j < Ny; j++)

for (int i = 0; i < Nx; i++)

image.SetPixel(i, Ny - 1 - j, setColor(Z[j, i]));

return image;

}

По сути, создаётся с помощью Bitmap изображение формата Nx на Ny построчно заполняя его методом SetPixel. В итоге можно сохранять любую карту для последующего её анализа или работы с ней.

В итоге, на момент написания данной работы имеется работающее приложение с функциями, описанными ниже.

## 3.4 Тестирование реализованного приложения, применение на практике

Запуская приложение, представляется главный экран (рисунок 26). На нём присутствует кнопки “Открыть карту”, “Сохранить карту”. При первом открытии рабочая область пуста.

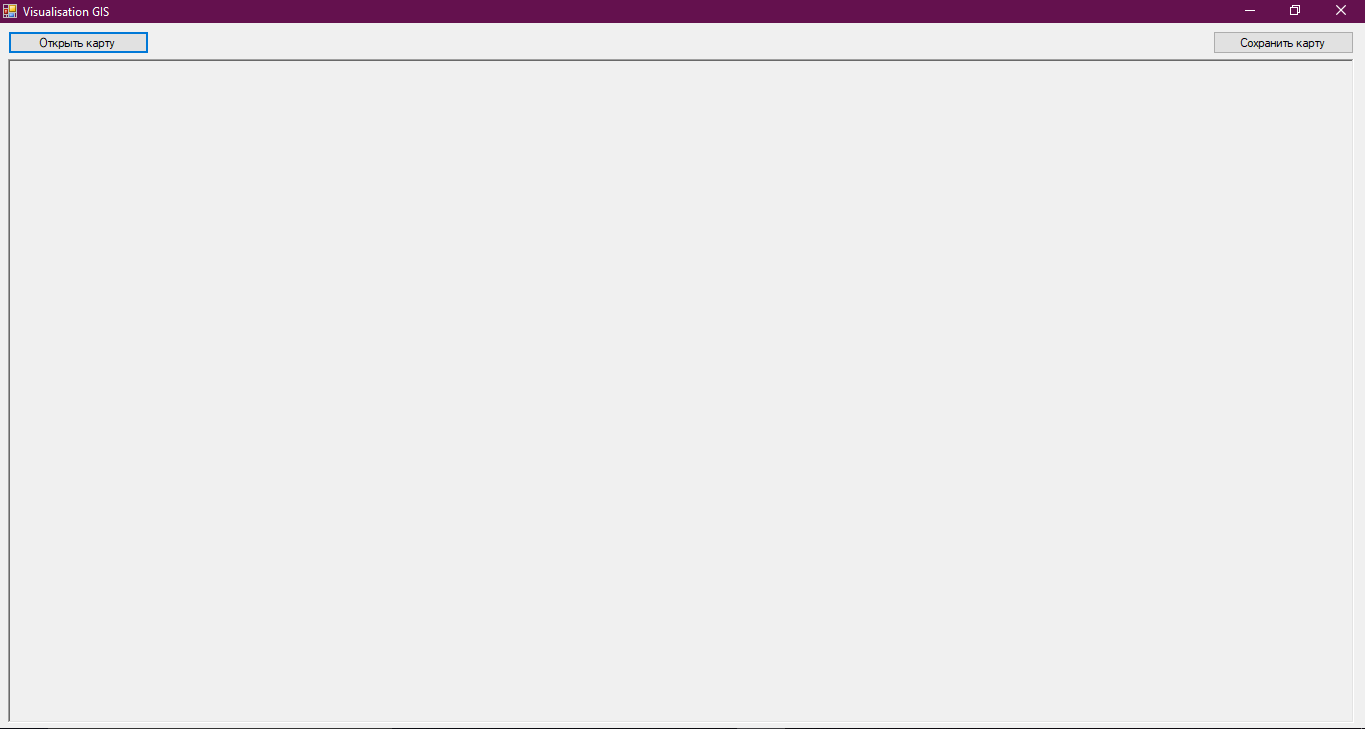


Рисунок 26 – Главный экран приложения

Для открытие .grd файла необходимо нажать на кнопку “Открыть карту” после чего произойдет открытие диалогового окна выбора файла (рисунок 27).

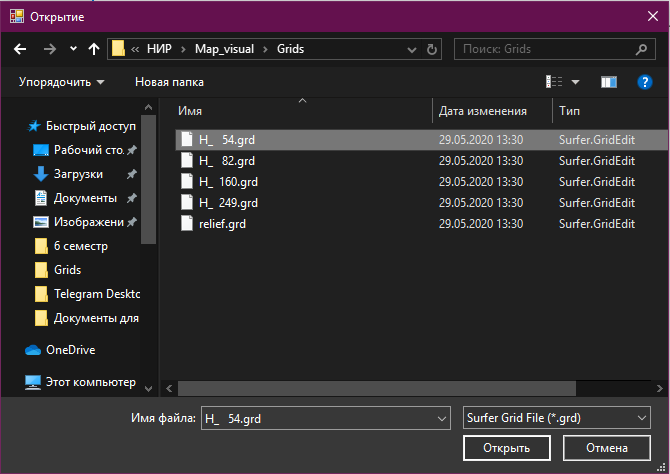


Рисунок 27 – Диалоговое окно выбора файла

Далее нажимая кнопку «Открыть», подтверждается выбор.

На главном экране появляется окно с некоторой информацией (рисунок 28), а именно:

1. Количество точек по оси X
2. Количество точек по оси Y
3. Начальная и конечная координаты по оси X
4. Начальная и конечная координаты по оси Y
5. Минимальное и максимальное значение высоты Z

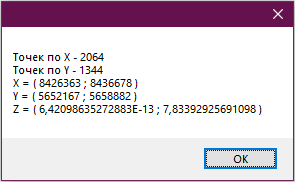


Рисунок 28 – Справочная информация о файле

Далее, нажав кнопку «ОК», происходит загрузка карты.

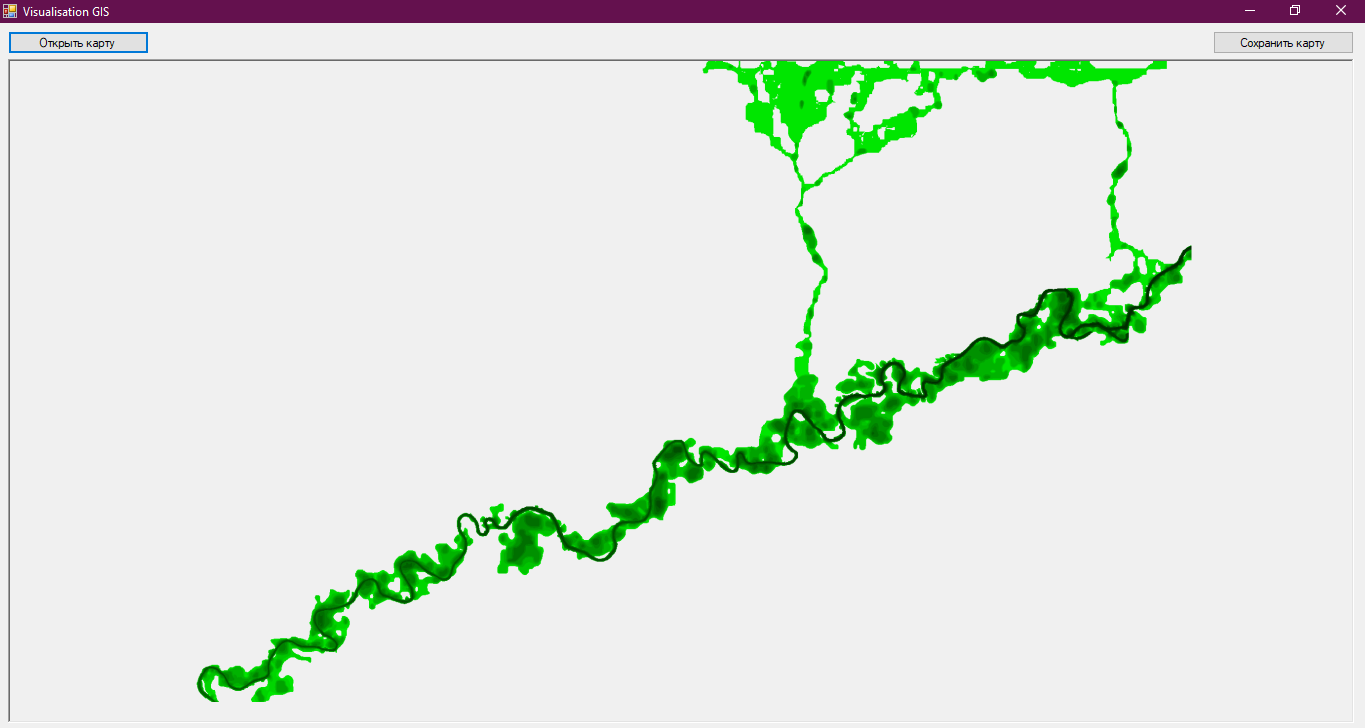


Рисунок 29 – Рабочая область с визуализированным файлом H\_ 54.grd

На рисунке 29 можно увидеть визуализацию нашего файла. На данный момент визуализируется файл зелёным градиентом без выбора цветовой гаммы. В последующей разработке программы цвет может быть выбран самим пользователем. Об этом будет сказано в следующей главе.

Далее будут продемонстрированы визуальные представления той же области, но только в другие периоды времени.

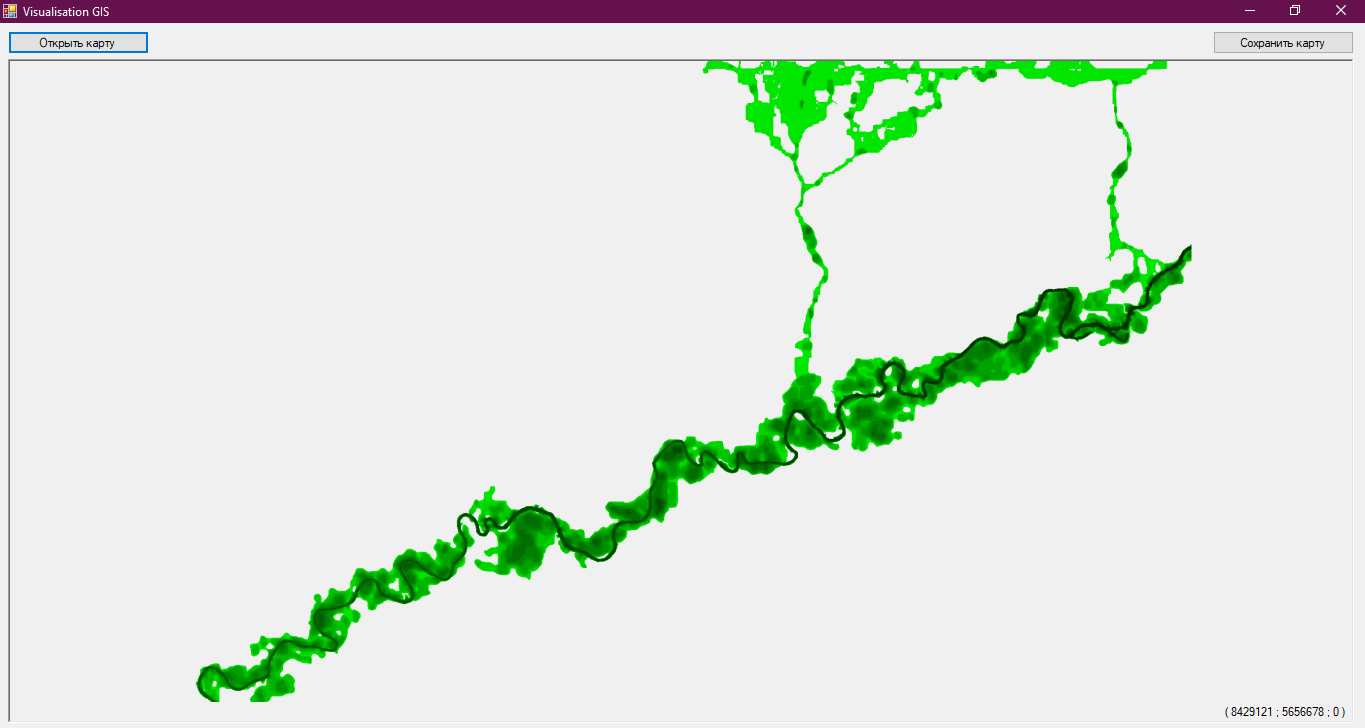


Рисунок 30 – Визуальное представление данных результата гидродинамического моделирования соответствует состоянию из выходного файла H\_ 82.grd

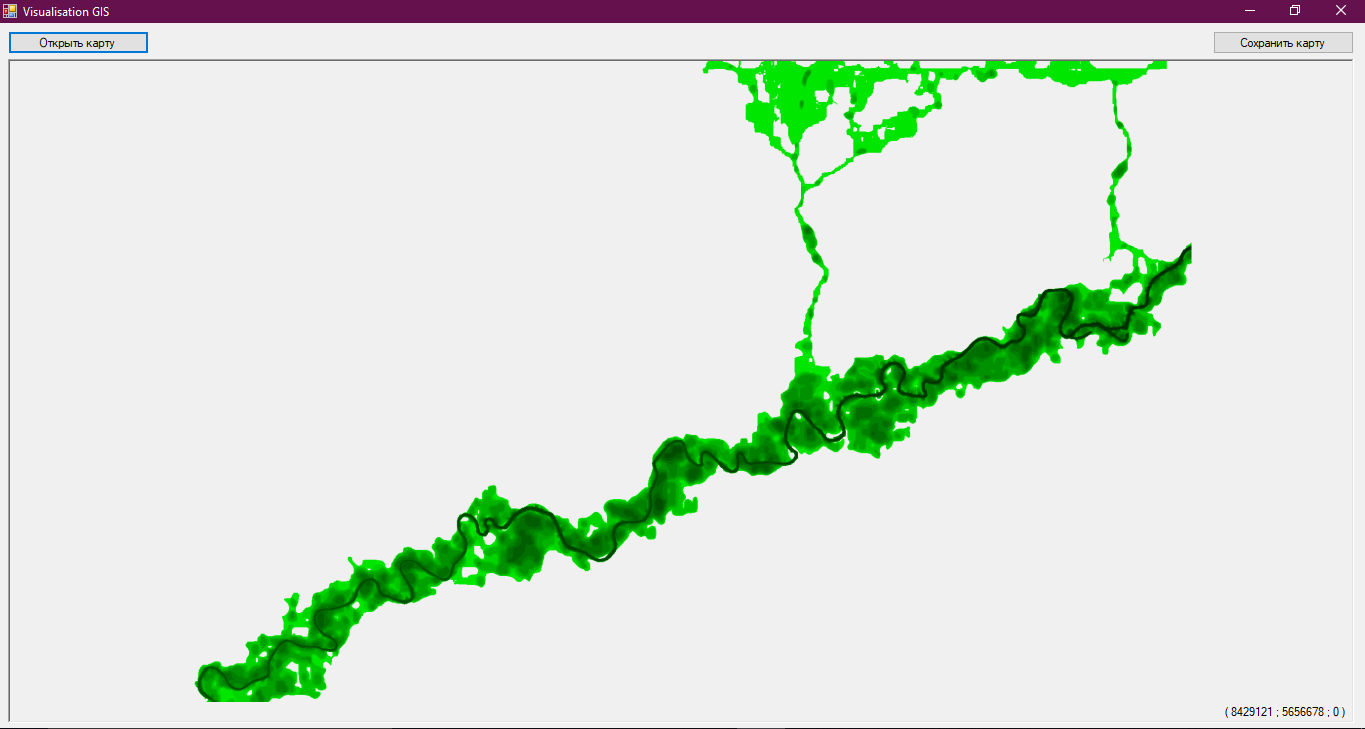


Рисунок 31 – Визуальное представление данных результата гидродинамического моделирования соответствует состоянию из выходного файла H\_ 160.grd

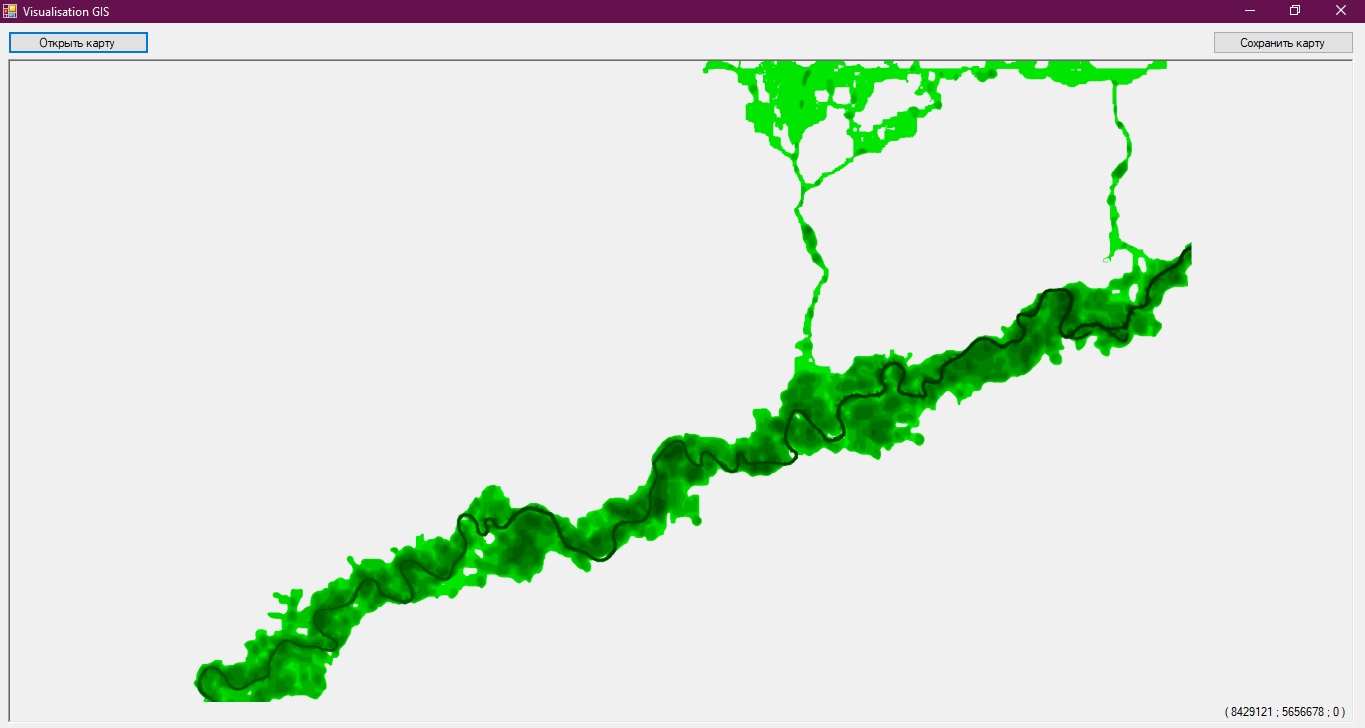


Рисунок 32 – Визуальное представление данных результата гидродинамического моделирования соответствует состоянию из выходного файла H\_ 249.grd

На рисунках 30-32 можно заметить, что оттенки зелёного различаются, это сделано для того, чтобы показать, как изменяется высота (значение Z) от расположения на карте (X; Y). Более тёмный цвет означает более высокое значение по модулю.

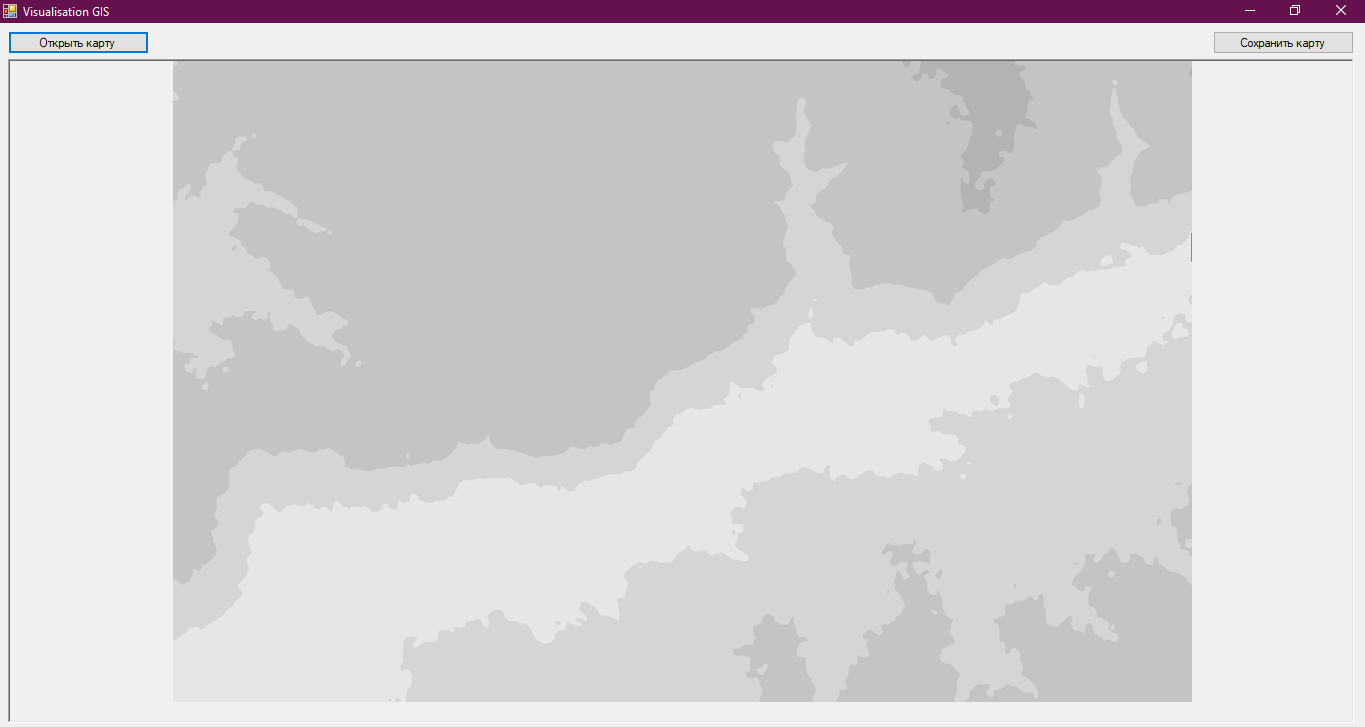


Рисунок 33 – Цифровая модель рельефа местности в окрестностях города Михайловка Волгоградской области

На рисунке 33 представлена визуализация рельефа той же местности в окрестностях города Михайловка, что и было показано на предыдущих рисунках. На нём, как и на предыдущих, более тёмный градиент означает более высокое значение Z по модулю.

Для сохранения визуализации нам необходимо нажать кнопку «Сохранить карту». Появится диалоговое окно, рисунок 34, где будет предложено выбрать название для изображения, а также расположение на диске.

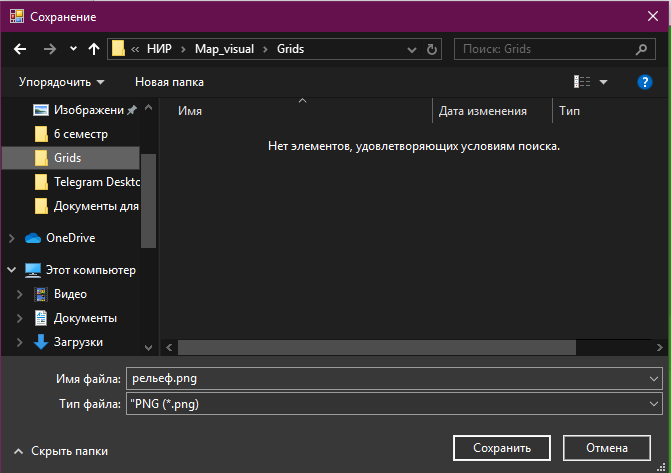


Рисунок 34 – Окно с сохранением изображения карты

В случае успешного сохранения, будет создан .png файл с выбранным нами названием. Сохранив и объединив картинки с картами на рисунках 29 и 33 получим визуализацию целой области, на которой будет изображён рельеф и результат одного из фалов расчета, проходящая по нему. Полученная визуализация изображена на рисунке 35.

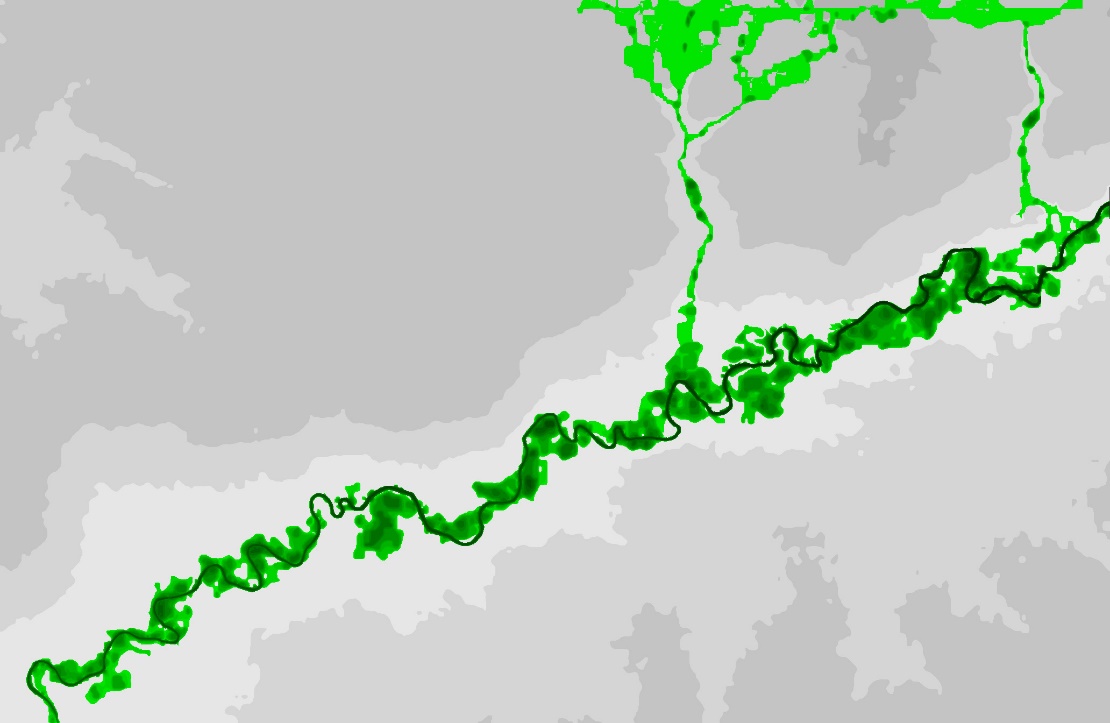


Рисунок 35 – Общая карта местности вблизи г. Михайловка, отображающая цифровую модель рельефа и результат моделирования

# Заключение

В рамках выполнения учебной практики, проектно-технологической практики были изучены методы визуализации данных, а также разработана информационная модель программы.

Основные результаты выполнения учебной практики: была изучена предметная область визуализации, ознакомились с методами визуализации многомерных данных, различными классами диаграмм, их применением в области научных исследований. Провел обзор в области визуализации связанных с представлением пространственных данных таких как картограмм, дистанционное зондирование Земли.

Во второй главе было описана разработка приложения для визуализации сеточных данных представленных в формате GRD. Изучена структура входного файла с расширением .grd, часто применяемых в основном для задач математического моделирования. Были описаны и разработаны функции для приложения, а в частности: функции перемещения по карте, масштабирования, функция для определения координат и высоты в указанной точке. Также выбрал и настроили цветовую палитру для отображения ЦМР и результатов гидродинамического моделирования. Разработана информационная модель программы, дополнены новые функции, описаны и расписаны дополнительные модули.

Благодаря рационально распределенному времени на протяжении всего периода производственной работа была выполнена в полном объеме.

# Литература

* 1. Авербух В. Л. и др. Опыт разработки специализированных систем научной визуализации // Научная визуализация. Квартал. – 2010. – Т. 4. – С. 27-39.
  2. Аноприенко А. Я., Ерёмченко Е. Н., Клименко С. В. Digital Earth как метод визуализации // GraphiCon 2017. – 2017. – С. 290-294.
  3. Бондарев А. Е., Галактионов В. А. Научная визуализация в задачах вычислительной физики: концепции, методы, перспективы // Новые информационные технологии в автоматизированных системах. – 2010. – №. 13 – 146 с.
  4. Васильев В. Р. и др. Контекстная визуализация пространственных данных // Информационные технологии и вычислительные системы. – 2004. – №. 4. – С. 25-34.
  5. Жексенаев А. Г. Основы работы в растровом редакторе GIMP (ПО для обработки и редактирования растровой графики) // Москва. – 2008. – Т. 1. – С. 2.
  6. Замятин А., Марков Н. Анализ динамики земной поверхности по данным дистанционного зондирования Земли // Litres. – 2018. – 175 с.
  7. Зенков А. И. Реализация модульного подхода при построении унифицированной системы научной визуализации // Сборник трудов научной международной конференции по компьютерной графике и анимации «Графикон-2002», г. Нижний Новгород. – 2002. – 3 с.
  8. Иванов А. В., Хилков С. А. Библиотека aiwlib—инструмент для создания приложений численного моделирования, визуализации и анализа результатов // Научная визуализация. – 2018. – Т. 10. – №. 1. – С. 110-127.
  9. Кашкин В. Б., Сухинин А. И. Дистанционное зондирование Земли из космоса. Цифровая обработка изображений // Логос. – 2001. – 246 с.
  10. Клышинский Э. С., Рысаков С. В., Шихов А. И. Обзор методов визуализации многомерных данных // Новые информационные технологии в автоматизированных системах. – 2014. – №. 17. – 530 с.
  11. Кузьмина Д. В. Использование диаграмм Эйлера-Венна для решения логических задач // Вестник современных исследований. – 2017. – №. 6-1. – С. 133-135.
  12. Лурье И. К. Геоинформационное картографирование. Методы геоинформатики и цифровой обработки космических снимков. // – 2008. – 423 с.
  13. Немчанинова Ю. П. Обработка и редактирование векторной графики в Inkscape (ПО для обработки и редактирования векторной графики) // Москва. – 2008. – 57 с.
  14. Пескова О. В. О визуализации информации // Инженерный журнал: наука и инновации. – 2012. – №. 1. – 173 с.
  15. Романова И. К. Современные методы визуализации многомерных данных: анализ, классификация, реализация, приложения в технических системах // Машиностроение и компьютерные технологии. – 2016. – №. 3. – 167 с.
  16. Рябинин К. В., Чуприна С. И., Бортников А. Ю. Автоматизация настройки систем научной визуализации на специфику разнообразных источников данных // Научная визуализация. Национальный исследовательский ядерный университет МИФИ. – 2016. – С. 1-14.
  17. Трифонова Т. А., Мищенко Н. В., Краснощеков А. Н. Геоинформационные системы и дистанционное зондирование в экологических исследованиях // М.: Академический проект. – 2005. – Т. 352. – С. 4.
  18. Cuenca E. et al. Multistream: A multiresolution streamgraph approach to explore hierarchical time series // IEEE Transactions on Visualization and Computer Graphics. – 2018. – Т. 24. – №. 12. – С. 3160-3173.
  19. Hejlsberg A., Wiltamuth S., Golde P. C# language specification. // Addison-Wesley Longman Publishing Co., Inc., 2003. – 672 с.
  20. Kersting O., Döllner J. Interactive 3D visualization of vector data in GIS // Proceedings of the 10th ACM international symposium on Advances in geographic information systems. – 2002. – С. 107-112.
  21. Liberty J. Programming C#: building. NET applications with C. // « O'Reilly Media, Inc.». – 2005. – 644 с.
  22. Piliczewski B. Application of the Golden Software Surfer mapping software for automation of visualisation of meteorological and oceanographic data in IMGW Maritime Branch // EGS-AGU-EUG Joint Assembly. – 2003. – 383 с.
  23. Risch J. S. et al. Multidimensional structured data visualization method and apparatus, text visualization method and apparatus, method and apparatus for visualizing and graphically navigating the world wide web, method and apparatus for visualizing hierarchies // пат. 7373612 США. – 2008. – 51 с.
  24. Roy D. P. et al. Landsat-8: Science and product vision for terrestrial global change research // Remote sensing of Environment. – 2014. – Т. 145. – С. 154-172.
  25. Shneiderman B., Wattenberg M. Ordered treemap layouts // IEEE Symposium on Information Visualization, 2001. INFOVIS 2001. – IEEE, 2001. – С. 73-78.
  26. Ивлиева Н. Г., Росяйкина Е. А. Обработка данных дистанционного зондирования Земли в ГИС-пакете ArcGIS // Огарёв-online. – 2015. – №. 4 (45).
  27. Law M., Collins A. Getting to know ArcGIS. // Redlands : ESRI press. – 2015. – С. 768.
  28. Graser A. Learning Qgis // Packt Publishing Ltd. – 2016. – 185 С.
  29. Сорокин В. А., Демин А. Ю. Разработка программного комплекса для визуализации 3d модели рельефа поверхности // Перспективы развития информационных технологий. – 2014. – С. 353-354.
  30. Симакова С. И. Инфографика: визуализация цифрового контента // Вестник Волжского университета им. ВН Татищева. – 2012. – 8 C.
  31. Callahan S. P. et al. VisTrails: visualization meets data management // Proceedings of the 2006 ACM SIGMOD international conference on Management of data. – 2006. – С. 745-747.
  32. Manovich L. Inside photoshop // Computational Culture. – 2011. – 1 С.
  33. Дьяконова Т.А. Программа для формирования сложной пространственной структуры нерегулярного рельефа русла для задач гидродинамического моделирования. // Диссертация. – 2018. – 168 С.
  34. Johnson B. Professional visual studio 2012. // John Wiley & Sons. – 2012. – 1108 С.
  35. Amann S. et al. A study of visual studio usage in practice // 2016 IEEE 23rd International Conference on Software Analysis, Evolution, and Reengineering (SANER). – 2016. – Т. 1. – С. 124-134.

# Приложение А Полученные при выполнении работы компетенции

При выполнении работы был проведен поиск информации по теме «Проектирование информационной модели программы для визуализации геопространственных данных» и ее критический анализ, обработка и переосмысление полученной информации, что соответствует компетенции

УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач.

В ходе выполнения работы в устной и письменной формах осуществлялась коммуникация с научным руководителем, а также посредством дистанционного обучения, что соответствует компетенции

УК-4 Способен осуществлять деловую коммуникацию в устной и письменной формах на государственном языке Российской Федерации и иностранном(ых) языке(ах).

В ходе выполнения работы были созданы и проанализированы требования к разрабатываемому программному обеспечению, что соответствует компетенции

ПК-4 Способен создавать и анализировать требования на разработку программно-информационных систем и подсистем.

При выполнении работы была создана информационная модель для разрабатываемого приложения, что соответствует компетенции

ПК-5 Способен осуществлять концептуальное, функциональное и логическое проектирование программно-информационных систем.