

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное учреждение науки
Институт прикладной математики
Дальневосточного отделения Российской академии наук
(ИПМ ДВО РАН)

На правах рукописи



СОЛОВЬЁВ Игорь Владимирович

РЕКОНСТРУКЦИЯ ТРЕХМЕРНОЙ МОДЕЛИ ГОРОДСКОЙ ОБСТАНОВКИ ПО ТОПОГРАФИЧЕСКОМУ ПЛАНУ

Специальность: 2.3.5 – Математическое и программное обеспечение
вычислительных систем, комплексов и компьютерных сетей

Диссертация
на соискание ученой степени
кандидата технических наук

Научный руководитель:
к.т.н., Кудряшов Алексей Павлович

Владивосток – 2021

Оглавление

Введение	4
Глава 1. Обзор методов реконструкции, основные положения, требования	11
1.1. Анализ проблемы	11
1.2. Методы реконструкции	12
1.3. Метод распознавания по топопланам	14
1.4. Методы представления трехмерных моделей	18
1.5. Методы выделения контуров	22
1.6. Методы распознавания текста	24
1.7. Обзор существующих программных систем	26
1.8. Выводы по главе	36
Глава 2. Выделение контуров на топоплане	37
2.1. Основные определения	37
2.2. Схема реконструкции	39
2.3. Волновой алгоритм	46
2.4. Анализ быстродействия и модификация	54
2.5. Выводы по главе	58
Глава 3. Классификация контуров и объектов	59
3.1. Выделение зданий	60
3.2. Векторизация контура здания	62
3.3. Распознавание внутренней информации здания	69
3.4. Выделение символов	71
3.5. Применение алгоритмов	80
3.6. Выделение дополнительных объектов топоплана	82
3.7. Выводы по главе	87
Глава 4. Трехмерная модель и текстурирование	88
4.1. Построение трехмерной модели здания	88
4.2. Географическая привязка и рельеф	89

4.3. Текстурирование	94
4.4. Программная реализация	105
4.5. Примеры реконструкций.....	109
Заключение.....	114
Список литературы	115
Приложение А. Свидетельство о государственной регистрации программы для ЭВМ	126
Приложение Б. Акт о внедрении.....	127
Приложение В. Описание разработанного программного комплекса.	128

Введение

Актуальность темы исследования. С развитием компьютерной техники стало реальностью создание масштабных трехмерных моделей реальных объектов окружающего мира. Появились технические средства для трехмерной реконструкции, такие как, трехмерные сканеры, лазерные дальномеры и др. Однако для протяженных реальных сцен применения этих устройств ограничено из-за относительно небольшого радиуса их действия. Поэтому, по-прежнему, одной из актуальных задач компьютерной графики на сегодняшний день является разработка систем трехмерной реконструкции для моделирования протяженных сцен, в частности, одной из важных практических задач является трехмерная реконструкция объектов городской обстановки и создание виртуальной городской среды. В качестве исходных данных, для реконструкции подобных сцен, обычно выступают существующие топографические планы местности, спутниковые данные, а также фотоизображения. К тому же актуальность трехмерного моделирования городского пространства объясняется не только ограниченностью радиуса действия существующих технических средств реконструкции, но и наличием на топопланах исчерпывающей семантической информации об объектах. Компьютерная трехмерная сцена города может использоваться при проектировке новых строений в существующей городской застройке, для визуальной ориентации и навигации в городском пространстве различными службами, а также для туристической и развлекательной сфер деятельности.

При этом значительная роль уделяется скорости процесса реконструкции, точности полученных моделей, их простой идентификации в городской обстановке. На практике приходится соблюдать баланс между скоростью и точностью реконструкции, т. к. эти параметры зависят от уровня требуемого конечного результата и степени вовлеченности оператора в каждый из этапов процесса реконструкции.

Степень разработанности проблемы. Последние несколько десятилетий значительно преобладает создание цифровых форм топопланов, представляющих единую векторную модель. Тем не менее, остается актуальной задача по векторизации существующих топопланов на бумажном носителе и последующее его представление в трехмерном виде для целей визуализации, ориентации и т. д. Данная процедура не автоматизирована, поэтому требуется участие оператора на всех этапах обработки данных.

Реализация подобной системы трехмерной реконструкции требует решения ряда комплексных задач, основными из которых являются следующие задачи:

- выделение и распознавание на изображениях зданий и других объектов (деревьев, светофоров, фонарей и т.д.);
- построение трехмерных моделей найденных объектов;
- построение трехмерного рельефа местности;
- текстурирование сцены с использованием фотоснимков реальных объектов либо типовых текстур.

Эти задачи, в некоторой степени, могут быть решены с помощью уже реализованных программных систем, таких как, например, Esri CityEngine, AutoCad, PhotoModeler, MapInfo, ГИС Карта, Autodesk 3ds Max и др. Первые две системы достаточно универсальны, но для узкоспециализированных задач, таких как реконструкция объектов городской застройки, эти программные комплексы не обладают всей полнотой необходимого функционала. В этих программных продуктах требуется значительная доля участия оператора при реконструкции, т.к. инструменты автоматизации построения сцены по топоплану отсутствуют.

Проведенное исследование показало, что существующие системы реконструкции трехмерных сцен городского пространства не в состоянии полностью обеспечить автоматизированный процесс построения трехмерной модели с использованием топоплана, модели рельефа и фотоизображений. При наличии практической задачи по реконструкции крупномасштабной сцены городской обстановки требуется максимально автоматизировать процесс, однако

упомянутые выше программные комплексы в первую очередь основываются на принципах ручного редактирования сцены. Это свидетельствует о том, что данная проблема актуальна на сегодняшний день и требует создания методов, алгоритмов и программных средств для автоматической трехмерной реконструкции объектов и сцен городской обстановки.

На сегодняшний день существует ряд исследований и прикладных работ в области компьютерного моделирования, машинного зрения и трехмерной реконструкции, которые могли бы помочь в решении обозначенных задач. Они связаны с такими именами, как Бобков В.А. [52, 53, 54, 55, 56, 57, 58], Andrew Zisserman [39, 49], Daniel Cremers [16, 17, 22, 23], А.С. Конушин [12, 28, 29, 30, 31, 45, 46, 66, 69, 70, 103, 104], David Forsyth [98] и многими другими.

Цель и задачи исследования. Целью диссертационной работы является разработка моделей, методов и программных средств для построения объемных сцен городской обстановки с использованием топографического плана, спутниковых данных о рельефе и фотоизображений местности.

Для достижения указанной цели решаются следующие **задачи**:

- 1) Анализ существующих методов и программных систем для построения объемных сцен городской обстановки.
- 2) Разработка метода выделения и распознавания информации на зашумленном бинарном изображении топографического плана.
- 3) Разработка алгоритма моделирования объектов городской обстановки (по распознанной информации) и рельефа (по существующим базам данных).
- 4) Разработка метода текстурирования зданий по набору фотоснимков.
- 5) Реализация программной информационной системы, использующей разработанные методы и алгоритмы для построения объемных сцен городского пространства.

Научная новизна диссертационной работы:

- 1) Предложен новый подход к реконструкции трехмерной сцены городского пространства, основанный на комплексном использовании топографического плана, спутниковых данных и фотоизображений зданий.
- 2) Предложен новый метод выделения замкнутых контуров на бинарном изображении, основанный на модификации волнового алгоритма. Он позволяет применять данный метод в других областях исследований.
- 3) Разработана архитектура программной среды, состоящая из настольного и мобильного приложения и обеспечивающая поддержку разработанных методов. Отличительной особенностью данной программной среды является её специализация на автоматизации всего процесса реконструкции с минимальным участием оператора.

Теоретическая и практическая значимость диссертационной работы.

Теоретическая значимость диссертационной работы заключается в создании модели процесса реконструкции трехмерных сцен городского пространства по топографическому плану. Разделение процесса на независимые модульные элементы способствуют привлечению к данной области исследователей из разных областей знаний, специализирующихся на конкретных этапах данного процесса.

Практическая значимость результатов диссертационной работы заключается в ориентации разработанных средств на решении практических задач, связанных с моделированием объектов городской застройки по разнородным данным и применением компьютерных моделей в проблемно-ориентированных приложениях. В диссертации разработаны методы и алгоритмы построения трехмерных сцен и объектов по топографическому плану, набору фотоснимков, геолокационным и спутниковым данным. Все поставленные требования реализованы в виде программной среды (Приложение А. Свидетельство о государственной регистрации программы для ЭВМ, [91]), состоящей из двух программных продуктов, один из которых предназначен для съемки

фотоизображений, а второй – для реконструкции. Разработанная программная среда решает задачу реконструкции как комплексно, так и использовать её для решения локальных задач, имеющих связь с данной тематикой. Участие оператора заключается в предоставлении программе исходных данных и корректировке результата в случае необходимости, т. к. практически все шаги трехмерной реконструкции выполняются автоматически. Полученная трехмерная сцена может быть экспортирована в распространенный компьютерный формат данных («.X» [20]) и впоследствии может быть использована в сторонних программных продуктах. Помимо этого, отдельно могут быть экспортированы автоматически выделенные на топоплане контуры зданий в формате «.SVG» [42].

Результаты диссертационной работы использовались в специализированной проектной компании (СПК) «Восток» (г. Владивосток) для создания 3D модели городской застройки г. Владивостока (Приложение Б. Акт о внедрении). Разработанные методы, алгоритмы и программные средства также могут применяться в других организациях, где решаются задачи построения и использования компьютерных моделей объектов городской застройки.

Решение задач диссертационной работы выполнялось в рамках гранта ИПМ ДВО РАН «Выполнение научно-исследовательских работ по теме в рамках реализации постановления Правительства Российской Федерации от 9 апреля 2010 г. № 220 по Соглашению № 075-15-2019-1878 от «5» декабря 2019 г.»

Методология и методы исследования. Объектом исследования в данной работе является процесс реконструкции трехмерной модели городской обстановки по топографическому плану. Предмет исследования – современные методы и средства разработки программных систем на всех этапах реконструкции трехмерной модели на основании цифрового изображения.

В работе использовались методы компьютерного зрения, обработки изображений, компьютерной графики, математический аппарат линейной алгебры и теории вероятности. Для метода распознавания текста на топографическом плане применялись статистические методы.

Область исследования соответствует специальности 2.3.5 (ранее: 05.13.11) – «Математическое и программное обеспечение вычислительных систем, комплексов и компьютерных сетей» по п. 7 «Человеко-машинные интерфейсы; модели, методы, алгоритмы и программные средства машинной графики, визуализации, обработки изображений, систем виртуальной реальности, мультимедийного общения».

Основные положения, выносимые на защиту:

- 1) Метод выделения и распознавания информации на зашумленном бинарном изображении топографического плана.
- 2) Алгоритм моделирования городской обстановки по топографическому плану.
- 3) Метод текстурирования зданий по набору фотоснимков с привязкой их к топографическому плану.
- 4) Программная информационная система, использующая разработанные методы для реконструкции городских сцен.

Достоверность полученных результатов базируется на корректной постановке задач, сопоставлении предлагаемых теоретических решений с результатами экспериментов с применениями современных методик моделирования и программных вычислительных средств. Правильность полученных результатов подтверждается использованием их в практических целях. Достоверность применения методов и средств исследования подтверждается апробацией основных положений работы в рецензируемых журналах, входящих в перечень журналов ВАК и свидетельствами о регистрации программного комплекса, включающими результаты диссертационного исследования.

Апробация работы. Основные положения диссертационной работы докладывались и обсуждались: на 23-й (г. Владивосток, 2013), 25-й (г. Протвино, 2015) и 27-й (г. Брянск, 2017) Международных конференциях по компьютерной графике и зрению (Графикон) [78, 80, 95].

Публикации. По теме диссертации опубликовано 14 работ. Из них 6 входящих в перечень научных журналах ВАК, 3 – в материалах конференций, 3 – в других изданиях, получено 2 свидетельства о государственной регистрации программы для ЭВМ [91, 92].

Личный вклад автора. Все результаты, составляющие основное содержание диссертации, получены автором самостоятельно. Основные задачи поставлены научным руководителем автора к.т.н. А.П. Кудряшовым [71, 72, 73, 74, 75, 85, 86]. В работе [84] автору принадлежит разработка метода реконструкции по топографическому плану, в работах [76, 77, 81] – идея метода выделения контуров и текстовой информации на топографическом плане, в работах [79, 82, 83] – идея и программная реализация метода текстурирования сцены и построение рельефа.

Структура и объем работы. Диссертация состоит из введения, 4 глав, заключения, списка литературы и 3 приложений. Содержание работы изложено на 133 страницах (включая 8 страниц приложения). Список литературы включает 104 источника. В работе содержится 112 изображений и 6 таблиц.

Содержание диссертации.

В первой главе приведен анализ существующих методов реконструкции. Представлен метод реконструкции городской сцены по топографическому плану.

Во второй главе представлен волновой алгоритм Ли, предложена его модификация и применение его на топографическом плане.

В третьей главе представлен метод классификации контуров. Предлагается метод распознавания текста на топографическом плане.

В четвертой главе представлен метод построения рельефа сцены и метод текстурирования зданий. Приводятся примеры реконструкций.

В приложениях представлено свидетельство о регистрации предложенного алгоритма, акт о внедрении и описание программного комплекса.

Благодарность. Автор выражает благодарность своему научному руководителю – к.т.н. Кудряшову Алексею Павловичу за идею работы и участие.

Глава 1. Обзор методов реконструкции, основные положения, требования

1.1. Анализ проблемы

Реконструкция объектов окружающего мира является важной задачей компьютерного моделирования объектов реального мира. Много областей человеческой деятельности использует их. К наиболее популярным можно отнести архитектуру, робототехнику, медицину, а также сферу развлечений. Некоторые области в современном мире трудно представить без какого-либо компьютерного моделирования. Например, современная архитектура ориентируется не только на то, как будет выглядеть создаваемый объект сам по себе, но и на то, как он будет вписан в существующую окружающую среду [93]. Виртуализация окружающего пространства становится важной частью навигации в городе, например, в автомобильных навигаторах. Сфера развлечений также интенсивно развивается. В частности, создаются компьютерные игры, в которых действия происходят в реально существующих локациях.

Вместе с тем, одновременно с ростом быстродействия современных вычислительных устройств, компьютеров, смартфонов и др., ежегодно растут и требования к качественным виртуальным моделям. При этом, желательно, чтобы время на создание таких высоко полигональных и фотореалистичных сцен не увеличивалось.

Реконструкция городского пространства является частным случаем реконструкции окружающего мира. В качестве инструментов для решения подобного рода задач выступают методы компьютерного зрения, распознавания образов, машинного моделирования, обработки изображений и т.д. При этом требуется полная или частичная автоматизация всего процесса моделирования. Задача реконструкции городской сцены уже решалась в той или иной степени множеством исследователей. Разработан ряд подходов, которые в той или иной степени решают подобные задачи. Все они отличаются по уровню вовлеченности

оператора в процесс реконструкции, необходимости в дополнительном оборудовании, стоимости процесса моделирования, затрачиваемых временных ресурсов и т.п. Далее будут рассмотрены основные подходы из этих вышеперечисленных.

1.2. Методы реконструкции

Метод машинного зрения. Наиболее популярный метод реконструкции.

Принято выделять следующие основные этапы обработки данных:

- 1) предобработка изображений;
- 2) сегментация;
- 3) выделение геометрической структуры;
- 4) определение относительной структуры и семантики.

На первом этапе происходит подготовка изображений к дальнейшей обработке. Изображения подвергаются фильтрации простых шумов, гистограммной обработке и т.д.

На этапе сегментации осуществляется выделение необходимой информации из изображения и классифицирование этих частей. Это один из наиболее сложных этапов, т.к. не существует единого подхода из-за различных исходных данных и целей выделения. За последние годы был достигнут значительный прогресс по проблемам соотношения точек и фрагментов изображений, выделения признаков внутри малых фрагментов, высокой точности 3D-позиционирования точек, выделения простых геометрических структур типа «точка», «граница», «пятно», «прямая», «угол» и др.

Выделение геометрической структуры также одна из нетривиальных задач. В настоящее время известно несколько основных алгоритмических подходов и математических формализмов, используемых при разработке практических систем анализа изображений. Это гистограммные преобразования, анализ проекций, линейная и нелинейная фильтрация изображений, яркостная и текстурная

сегментация, корреляционное обнаружение и согласованная фильтрация, морфологический подход Пытьева [90], математическая морфология Серра [40], метод "нормализации фона", преобразование Хафа [25, 18], структурно-лингвистический подход и ряд других.

На последнем этапе происходит «осмысление» модели. Одним из часто встречающихся инструментов в последнее время являются нейронные сети.

Метод распознавания образов на спутниковых картах. Распознавание образов на спутниковых картах осложняется рядом причин:

- 1) нечеткие границы объектов;
- 2) искажение снимков, т.к. не всегда они делаются под прямым углом к поверхности земли;
- 3) наличие шумов в виде облаков или других природных объектов.

Существует ряд методов, которые решают в определенной степени такую задачу.

Метод лазерного сканирования. Преимуществами данного метода являются высокая скорость и точность создания трехмерных моделей. Однако в этом случае формируется облако трехмерных точек. Поэтому в автоматическом режиме невозможно выделить характерные объекты из сцены. Например, поверхность рельефа и фасады зданий представляют в такой системе единое целое. Из-за этого трехмерным данным нельзя назначить атрибуты и использовать в качестве физических моделей, имеющих определенные свойства. Кроме того, лазерное сканирование характеризуется большим объемом получаемых данных, что предъявляет высокие требования к вычислительным возможностям систем.

Метод сравнения спектров графов. Данный метод имеет противоположную от лазерного метода структуру. Информация формируется не в монолитные данные, а представляется в виде графов. Для сравнения структур используются спектры графов [93]. Преимуществом структурных методов и алгоритмов является то, что они позволяют анализировать большое множество элементов на основе малого количества простых составляющих и правил

формирования графической модели. Также структурные методы позволяют описать те характеристики объекта, которые исключают его отнесение к другому классу, что повышает надежность распознавания.

С помощью методов выделения отрезков на изображении (например, преобразования Хафа) формируется исходный набор отрезков. Далее эти отрезки соединяются между собой, если их концы достаточно близки. Соединенные отрезки формируют графы. Графы описываются в виде матриц смежности. Сравнение графов состоит в поиске соответствий между структурами графов на основе более или менее строгих ограничений. Используются методы сравнения на основе спектральной теории графов. Приводится схематичная реконструкция (Рисунок 1.1)

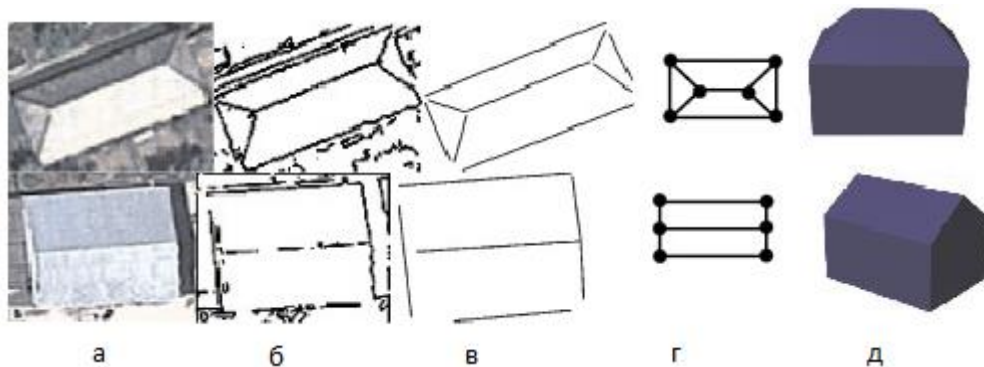


Рисунок 1.1 Реконструкция крыш зданий: а – спутниковые снимки реконструируемых объектов; б – результаты выделения контуров с использованием преобразования Хафа; в – векторизованные изображения; г – подходящий граф; д – реконструируемые модели

1.3. Метод распознавания по топопланам

Данный метод трехмерной реконструкции заключается в восстановлении сцены по топографическому условному изображению местности. Как правило совместно с этим используются дополнительные средства, которые позволяют восполнить тот необходимый набор информации, который в полной мере не может дать топоплан.

Векторизация контуров. Векторизация контуров применяется после выделения на изображении массива точек. Эти точки не упорядочены, а,

следовательно, их сложнее анализировать. Для устранения этой проблемы применяется один из методов векторизации, который направлен на создание единого контура с четкой последовательностью точек.

Метод наименьших квадратов. Один из основных методов векторизации контуров. Он заключается в том, чтобы найти коэффициенты некоторой кривой, которая бы подходила для части контура с наименьшей ошибкой отклонения. Например, для стены здания в большинстве случаев это будет прямая с двумя коэффициентами. Контур представляется в виде отдельных частей. Каждая такая часть может быть получена путем нахождения его граничных точек.

Предыдущие работы с реконструкцией по топоплану. Существует ряд отечественных и зарубежных работ, посвященных методам выделения информации с топографических планов. В работе [62] авторами дается описание метода реконструкции зданий без автоматизированного алгоритма распознавания. Трехмерная модель городской сцены строится с помощью коммерческих продуктов и на каждом этапе реконструкции требуется участие оператора. Здания на топоплане выделяются вручную, что значительно увеличивает время реконструкции сцен больших размеров. Выделенные вручную контуры зданий являлись основой трехмерных моделей. Высота каждой модели соответствует данным, полученным в результате наземного лазерного сканирования территории (Рисунок 1.2). Дополнительно, авторы осуществляют цифровую фотосъемку местности для последующего текстурирования зданий. Фотосъемка осуществляется в условиях максимально приближенной к центральной проекции при определенных погодных условиях. Все фотоснимки обрабатываются вручную в графическом редакторе. Итоговые изображения вручную наносятся на трехмерные объекты сцены.

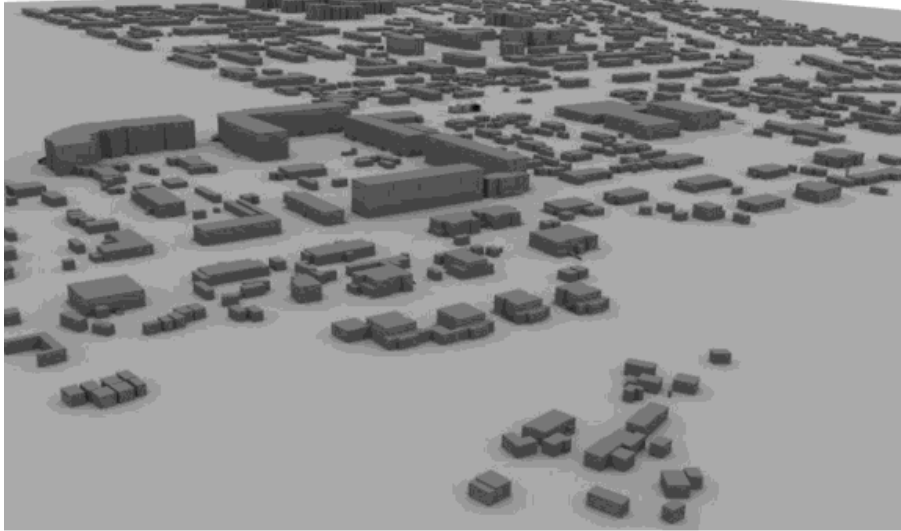


Рисунок 1.2 Итоговая сцена

В статье [102] предлагается метод распознавания контуров на основании построения так называемых «остов», по которым строится линейно-узловая структура в виде графа с вершинами – точками ветвления и ребрами – ломаными между ветвлениями.

Авторы проводят предварительную обработку остова (замыкание некоторых линий) и его упрощение (удаление мелких полигонов, петель, выпрямление углов контуров, уменьшение количества точек ломаных). Внутри потенциальных контуров зданий ищется надпись, состоящая из символов, обозначающих количество этажей, огнестойкость («К», «П», «Д»), заселенность («Н», «Ж»). В результате получается набор векторных контуров, к которым привязана информация о надписях внутри зданий (Рисунок 1.3).



Рисунок 1.3 Фрагмент исходных данных и результат распознавания

Автоматический алгоритм реконструкции городской сцены предлагается в статье [33]. Метод основан на использовании системы лазерного сканирования трехмерного пространства и сопоставлении полученного облака точек с топографическим планом на основании элементов тензорного анализа. Данные LiDAR обеспечивают точное представление поверхностей зданий, но в них содержится плохая информация о текстуре. К тому же затруднено извлечение границ зданий на основании этих данных. По этим причинам авторы используют топопланы.

Облако точек, полученное в результате использования LiDAR, обрабатывается для извлечения точек крыш. Для этого используются элементы тензорного анализа. Описание геометрических особенностей точек крыш зданий осуществляется с помощью метода тензорного голосования (Tensor voting method / TVM). Для этого используется тензор второго порядка (1.1):

$$T = [e_1 \ e_2 \ e_3] \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix}, \quad (1.1)$$

где e_1, e_2, e_3 – базисные ортогональные вектора; λ_1, λ_2 и λ_3 – собственные значения для базисов. Для каждой точки облака происходит «голосование» за то, что является ли она частью отдельного сегмента сцены. Голосование осуществляется с помощью тензора, который рассчитывается следующим образом (1.2):

$$T = \sum_{i=1}^m w_i T_i^p, \quad (1.2)$$

где w – Гауссова функция затухания, зависящая от расстояния между рассматриваемой точкой и места «съемки», m – размерность пространства (равна 3), p – вектор координат точки.

Сегментированные данные вместе с топопланом объединяются в одну систему координат (Рисунок 1.4).

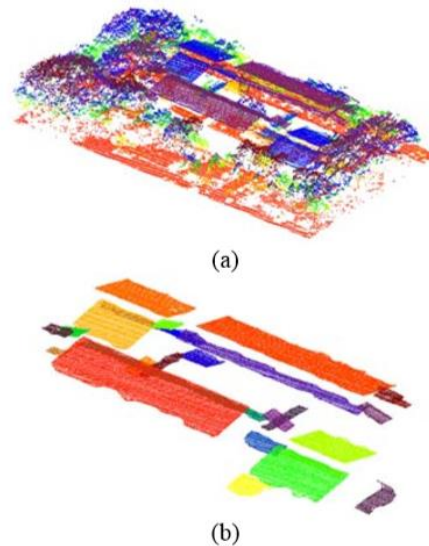


Рисунок 1.4 (a)ОблакоLiDAR точек и (b) результат его сегментации

Метод не предусматривает автоматического текстурирования моделей, к тому же требует наличия сложного измерительного прибора.

1.4. Методы представления трехмерных моделей

Трехмерная модель – виртуальная совокупность точек, граней, текстур визуального объемного образа. Существует множество как открытых, так и закрытых форматов хранения трехмерных моделей. Часть из них позволяет содержать только геометрию объектов, другая часть дополнительно может хранить текстуры, остальные позволяют хранить иную информацию о сцене, например, анимацию, освещение и т.п. Многие программные продукты, связанные с редактированием трехмерных моделей, как правило, имеют собственный оптимизированный формат. Однако, существуют и универсальные форматы, которые могут быть использованы большинством программных продуктов. Наиболее распространенные форматы хранения трехмерных моделей:

- «.OBJ» – открытый формат файла. Содержит позицию каждой вершины, связь текстур с вершинами, нормали вершин, полигоны. Является одним из самых популярных форматов.

- «.X» – открытый формат файла. Разработан компанией Microsoft для использования на платформе DirectX. Помимо информации о геометрии сцены и её текстурах может содержать анимацию. Широко используемый формат данных.
- «.STL» – формат файла, в котором сцена описывается набором треугольных граней и их нормалей. Несмотря на то, что формат распространен среди программных продуктов, тем не менее, описанные в нем модели имеют невысокую точность, а для сложных моделей размер файла занимает большой объем данных.
- «.3DS» – закрытый формат файла, разработанный компанией Autodesk. Содержит большой набор информации о сцене.
- «.DWG» – основной формат бинарных файлов для некоторых САПР-программ, например, AutoCAD. Позволяет хранить информация как о трехмерной модели, так и о двухмерной.

Существует широкий спектр различных методов представления трехмерных сцен как моделей городского пространства [36, 63], так и произвольных моделей [27, 87].

Полигональное моделирование в трехмерных редакторах. Данный метод реконструкции трехмерных моделей является одним из самых первых, появившийся в начале эры компьютерного моделирования. Основной принцип данного подхода заключается в представлении моделируемого объекта в виде набора полигонов. Каждый полигон задается набором вершин и ребер между ними (Рисунок 1.5). Простота данного метода, одновременно, является и достоинством, и недостатком данного метода, т.к. несложные объекты с небольшим количеством граней могут быть восстановлены за короткое время. При этом детализация высокоточных моделей требует огромного количества полигонов. Создание и обработка сотен тысяч и миллионов полигонов в таких моделях требует значительных производственных мощностей от платформы и оборудования.

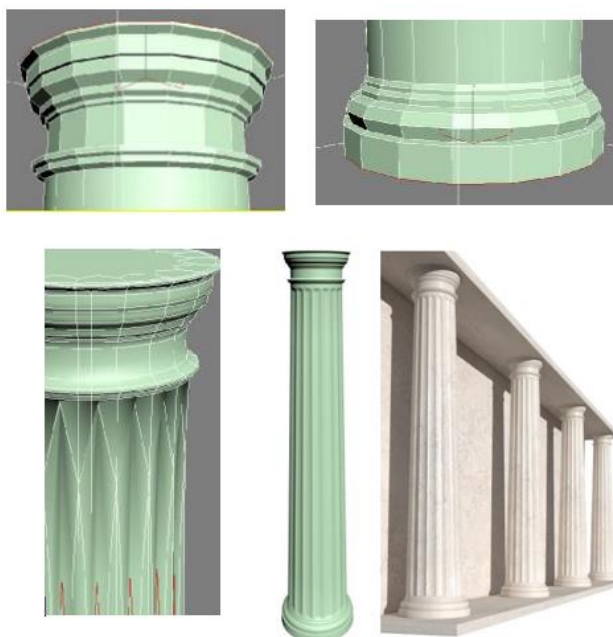


Рисунок 1.5 Полигональная модель

Полигональное моделирование благодаря своей простоте является неотъемлемым инструментом большинства информационных сред трехмерной реконструкции.

Триангуляционная поверхность. Трехмерная модель при таком подходе строится по набору изображений. Среди изображений этого набора ищутся соответствия, т.е. точки, которые являются общими для нескольких изображений. Образуется набор точек объекта. По этим точкам строится карта глубин, на основании которой происходит построение трехмерной модели.

Существенным достоинством данного метода является его универсальность к применимому объекту реконструкции. Объект может содержать мелкие и детализированные части. Данный метод достаточно точно восстанавливает эти детали.

Результатом реконструкции с помощью данного метода является триангуляционная поверхность (Рисунок 1.6). Все объекты сцены представляются в виде непрерывной последовательности сегментов. Выделить из неё отдельные объекты весьма затруднительно. Метод особенно полезен при реконструкции рельефа местности, т.к. итоговая модель повторяет все неровности ландшафта.

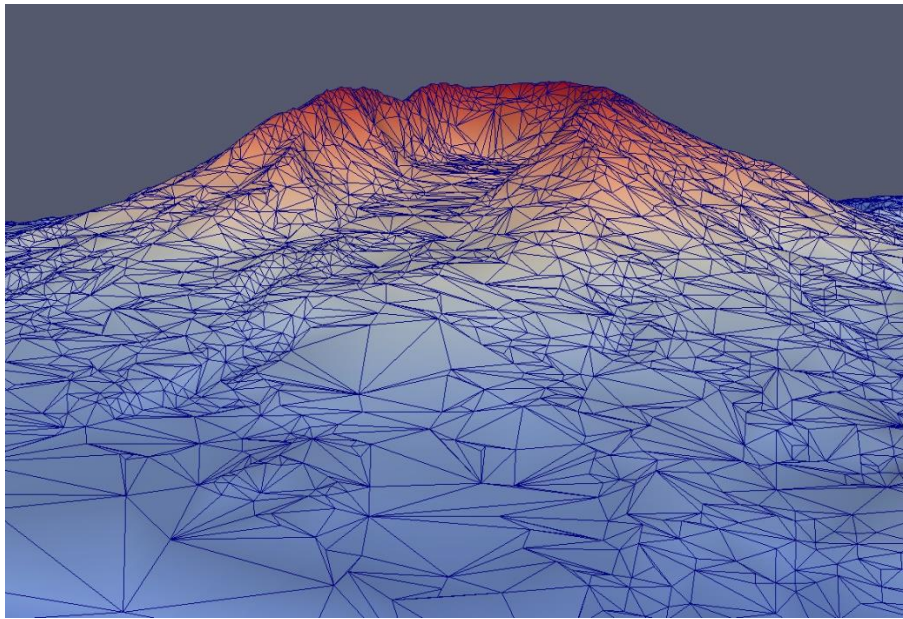


Рисунок 1.6 Рельеф, построенный с помощью триангуляционной поверхности

Воксельное моделирование. Воксельное представление трехмерного объекта заключается в том, чтобы преобразовать некоторое тело к набору упорядоченных кубов (вокселей). Пространство разбивается на набор кубов, который представляет из себя трехмерный массив данных. Каждый элемент массива обладает некоторым признаком того, содержится ли внутри его куба часть трехмерного объекта. Каждый куб в зависимости от плотности тела или текстуры может изменяться по показателю прозрачности и содержать свой свет (Рисунок 1.7).

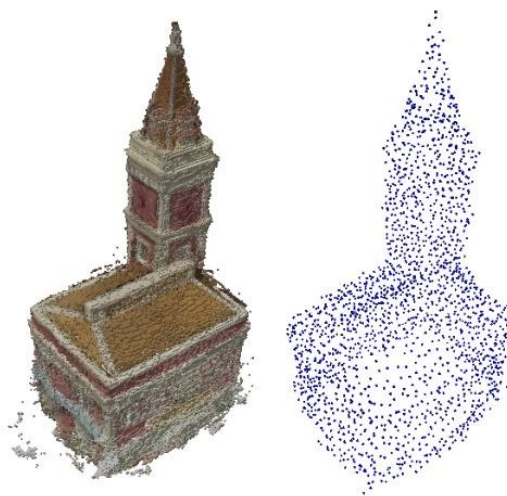


Рисунок 1.7 Воксельное представление объекта (из [19])

Точность приближения набора вокселей к реальному объекту зависит от размера вокселей. Чем меньше размер каждого из них, тем точнее модель. В связи с этим возрастает требуемый объем данных, который будет пропорционален кубу разрешения трехмерной сетки. Воксельное представление полезно при реконструкции сплошных тел. Объемные сцены представляются в виде простого набора данных. Однако, данный метод дает неточное представление модели, если стороны воксельных кубов не параллельны частям объекта.

Воксельный метод прост в реализации и может быть достаточно успешно применен для объектов со сложной геометрической формой. Сложность структуры полученной трехмерной модели является одним из важнейших недостатков методов.

Все объекты сцены представляются в виде непрерывной последовательности сегментов.

1.5. Методы выделения контуров

Реконструкция сцены по некоторому набору изображений [14, 15, 24, 37, 38, 48] – частый случай. Изображения могут быть разного рода. Это могут быть аэрофотоснимки [5, 6, 7], топографический план, фотоснимки сцены «изнутри» [43, 44], последовательность кадров при видеосъемке [65, 100] и т.д. В ряде случаев трехмерные модели строятся на основании выделения геометрии объекта с изображений. На таких изображениях находятся различные контуры. Из них оставляются только те, которые имеют отношение к реконструируемой модели. По особенностям контуров строятся трехмерные модели.

Использование детекторов. Данный подход заключается в создании некоторого фильтрующего окна. Оно, как правило, квадратное и небольшого размера. Фильтрующее окно перемещается от точки к точке изображения. В каждой точке вычисляется так называемый отклик, т.е. совокупность значений некоторой функции во всех точках окна. На основании этих значений

рассчитывается скорость изменения значений. Чем выше эта скорость, тем четче граница на изображении.

Существует ряд различных операторов, которые используют фильтрующее окна. К наиболее известным из них относится оператор Собеля [26, 41], Робертса, Превитта, Канни [8, 9, 10, 11]. Существуют другие альтернативные функции, как, например, в [99].

К недостатку данного метода можно отнести неспособность получения на выходе упорядоченных последовательностей точек, а также чувствительность к шуму на изображении.

Данный метод пользуется большой популярностью во многих практических задачах, т.к. прост в реализации и позволяет быстро обрабатывать изображения.

Отслеживающие алгоритмы. Данный тип методов заключается в том, чтобы найти любую точку контура объекта и последовательно идти от неё к другой до тех пор, пока все точки контура не закончатся (Рисунок 1.8). При таком подходе у алгоритма возникают трудности с выделением внутренних контуров, однако их реализация одна из самых простых.

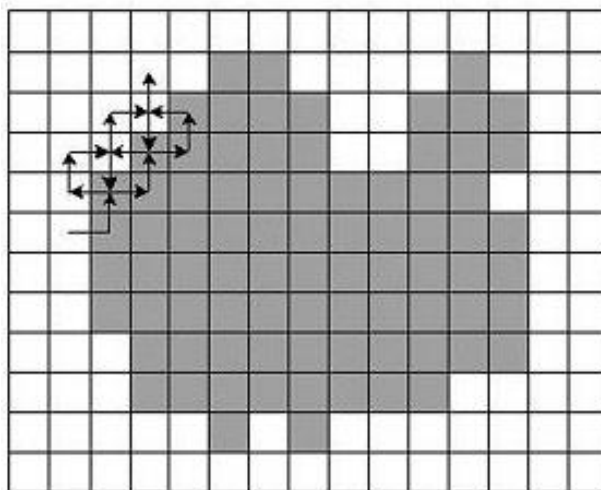


Рисунок 1.8 Схема работы алгоритма

1.6. Методы распознавания текста

Трехмерная реконструкция сцены, в которой требуется распознавание текста в исходных изображениях, достаточно редкий случай, т.к. часто распознаются именно трехмерные модели. Данные методы часто используются во многих прикладных задачах. Часто применяется на производстве, в сфере безопасности, при оцифровке книг, при контроле дорожного трафика и других. Распознаваемый текст может быть на любом из языков, написан различными шрифтами, прописью или печатно.

Использование шаблонов. Суть данного метода заключается в том, чтобы подготовить заранее все возможные изображения символов (эталон), которые могут встречаться на изображении [59, 60]. Часть изображения, в которой есть буква, сравнивается с каждым эталоном. В результате выбирается тот эталон, с которым изображение совпадает наибольшим образом. Способы сравнения изображения могут быть разные.

Одним из наиболее простых и популярных является метод кросс-корреляции. Он заключается в том, чтобы найти коэффициент корреляции между матрицами изображений (1.3):

$$r = \frac{\sum_m \sum_n (A_{mn} - \bar{A})(B_{mn} - \bar{B})}{\sqrt{\sum_m \sum_n (A_{mn} - \bar{A})^2 \sum_m \sum_n (B_{mn} - \bar{B})^2}}, \bar{A} = \sum_{m,n} \frac{A}{mn}, \bar{B} = \sum_{m,n} \frac{B}{mn}, \quad (1.3)$$

где \mathbf{m} , \mathbf{n} – размеры матриц.

В каждом элементе таких матриц хранится число, которое характеризует соответствующую точку на изображении. Как правило, это яркость пикселя. Чем выше по модулю коэффициент корреляции, тем сильнее похожи изображения.

Подход с использованием эталонов оправдан в тех случаях, когда количество использованных на изображении символов невелико и заранее известно, что шрифт, размер и наклон текста будут таким же, как и у эталона. В противном случае придется иметь эталоны под разные варианты написания символа, что сильно

замедлит процесс распознавания, т.к. потребуется больше времени, чтобы обработать все эталоны для каждого символа.

Нейронные сети. Данный метод достаточно общий для различных информационных дисциплин. Нейронные сети хорошо себя зарекомендовали в распознавании образов и, в частности, в распознавании текста. Общий подход заключается в создании структуры, пропускающей сигнал, который преобразуется по формулам на элементах сети и формирует ответ [50].

Нейронная сеть используется как классификатор при распознавании текста. Данный классификатор может быть обучен, путем настраивания коэффициентов на элементах сети. Лучшее распознавание текста достигается путем подбора наиболее подходящих параметров на элементах сети.

Недостатком данного метода является необходимость его обучения. Успех зависит от корректности установленных данных и от их количества.

Статистические характеристики. Суть данного метода состоит в определении некоторых характеристик, которые были бы уникальны для каждого символа [51]. Каждому возможному на изображении символу ставится в соответствие некоторая совокупность числовых значений. В качестве исходных данных берется набор точек, из которых состоит анализируемый символ. Если числовые характеристики символа и эталона близки, то можно утверждать, что символ распознан верно. Одна из наиболее известных методик использует вычисление центральных моментов.

Все точки символа используются для вычисления следующих значений (1.4)

$$m_{pq} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} (x - \bar{x})^p (y - \bar{y})^q f(x, y), \quad (1.4)$$

Где M, N – размеры изображения символа; x, y – координаты пикселя; \bar{x} и \bar{y} "с чертой" – координаты центра тяжести; $f(x, y)$ – яркость пикселя; $p, q \in \{0, 1, \dots, \infty\}$ – порядки относительно осей координат. Порядки служат некоторой мощностью, на которой соответствующий компонент взят в относительно других.

Путем определенных преобразований центральные моменты нормализуются и в результате распознавание текста становится более инвариантным к повороту, масштабированию и смещению.

1.7. Обзор существующих программных систем

На сегодняшний день существует множество программных систем, позволяющих на определённом уровне решить поставленные задачи. Однако, ни одна из рассмотренных ниже систем не способна создавать трёхмерную модель по топографическому плану автоматически. Тем не менее, представленные системы имеют мощный набор инструментов для такого вида реконструкций.

3D-генераторы. Ниже представлены некоторые наиболее известные программные системы, позволяющие создавать трехмерные реконструкции как с использованием топоплана, так и без него.

Esri CityEngine [13]. Данный продукт позволяет создавать протяженные модели городского пространства. Система дает возможность генерировать городское пространство по заданным правилам. При необходимости оператор может вручную вносить изменения в необходимый объект сцены (Рисунок 1.9).

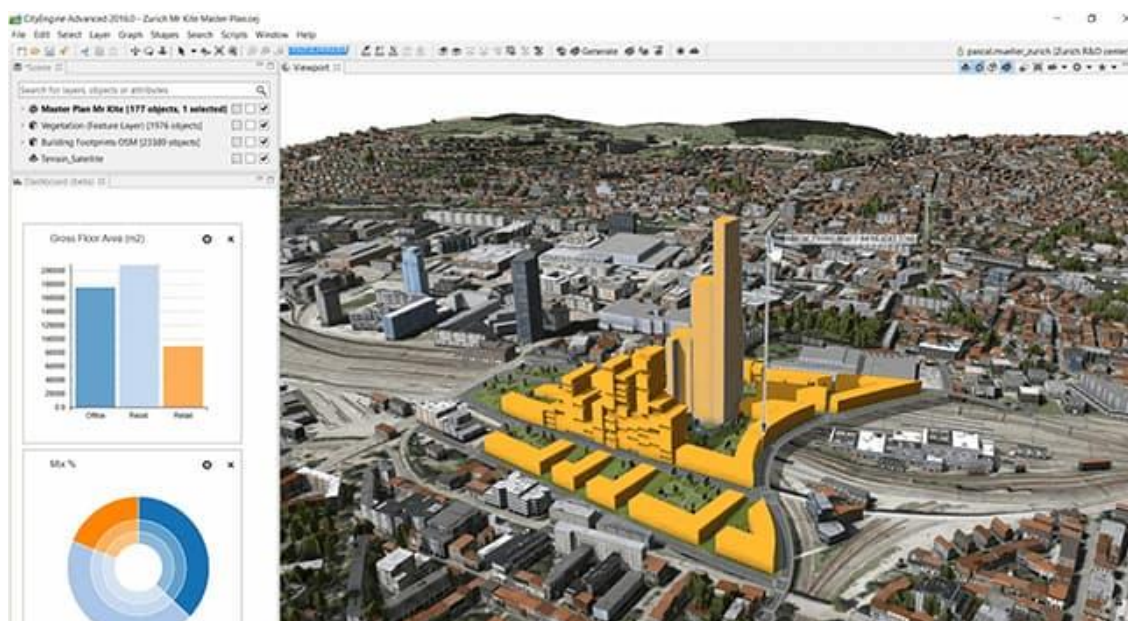


Рисунок 1.9 Моделирование городского пространства в Esri CityEngine

CityEngine позволяет профессиональным пользователям ГИС, САПР и 3D:

- быстро создавать 3D-города из существующих двумерных ГИС-данных;
- осуществлять 3D-проектирование на основе ГИС-данных и процедурных правил;
- эффективно моделировать городскую среду для анализа, прогнозирования и улучшенного визуального представления.

CityEngine предоставляет возможность прямого экспорта созданных 3D-моделей в другие программные пакеты:

- для анализа или управления данными (например, ArcGIS);
- для трехмерного редактирования (Maya, 3D Max);
- для 3D-визуализации (ArcGlobe, Google Earth);
- в игровые движки (Unity, Unreal);
- в решения рендеринга (визуализации) высокого качества (RenderMan);
- в качестве готовых облачных сервисов (RealityServer)

CityEngine не предлагает решений для автоматического построения 3D-городов по перспективным снимкам или лидарным данным. Основное назначение CityEngine – создание 3D-данных и расширенные возможности их экспорта. В результате данные из CityEngine могут быть использованы в различных коммерческих и частных решениях для 3D-визуализации.

MapInfo [34]. Мощный программный комплекс, который имеет множество возможностей применения для работы с картографической информацией (Рисунок 1.10). Предназначение данного программного комплекса – сбор, хранение, отображение, редактирование и анализ геоинформационных данных. MapInfo имеет внушительный набор инструментов для создания и редактирования топопланов и дизайна карт. Система имеет встроенный язык высокого уровня, что позволяет значительно расширить её возможности.

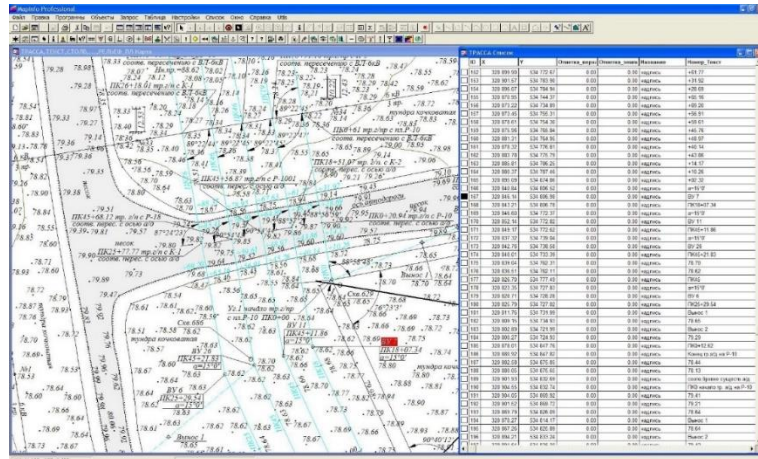


Рисунок 1.10 Редактирование топоплана в MapInfo

Одним из главных недостатков системы является её концентрация на разработке карт, а не на их анализе. Хотя дополнительные плагины позволяют её расширять в плане большей визуализации территории. Система не производит обработку топопланов, которые были загружены в неё не в векторных форматах. Для выделения объектов на таком топоплане оператору необходимо затратить достаточное количество времени.

На основании ручного создания изолиний и внесения географических данных о высоте в определенных точках карты есть возможность построения трехмерного рельефа территории. На основании указанных данных выполняется триангуляция методом Делоне и строится поверхность (Рисунок 1.11).

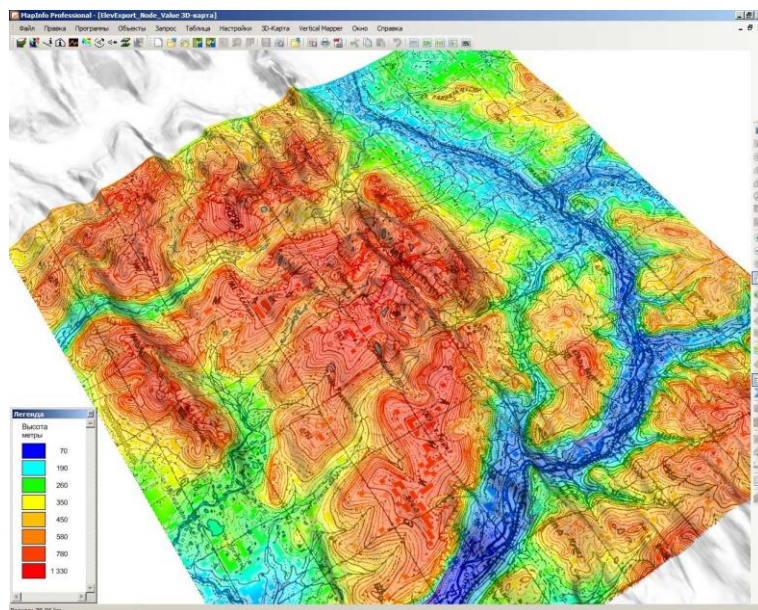


Рисунок 1.11 Трехмерный рельеф, построенный в MapInfo

Достоинством данного программного комплекса является наличие широкого инструментария по разработке и редактированию карт. Все объекты, наносимые на карту, могут быть сохранены в отдельной базе данных, к которой могут происходить обращения с помощью запросов.

AutoCAD Map 3D [2]. Одна из систем автоматизированного проектирования. Данное программное обеспечение обладает широкими возможностями создания и редактирования как двумерных, так и трехмерных чертежей (Рисунок 1.12).

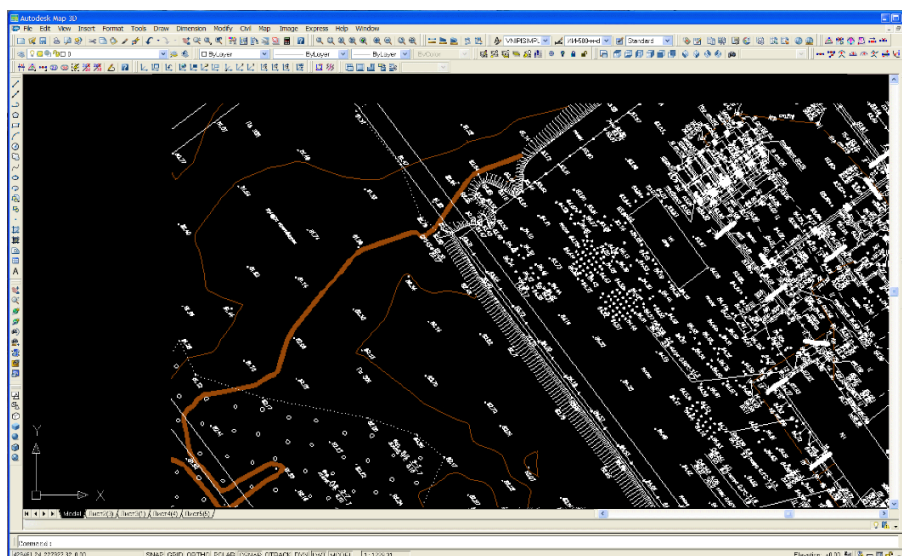


Рисунок 1.12 Редактирование топоплана в AutoCAD

Система позволяет работать с облаками точек (Рисунок 1.13). Огромное количество данных может быть визуализировано с помощью импорта информации о лазерном 3D сканировании/LiDAR.

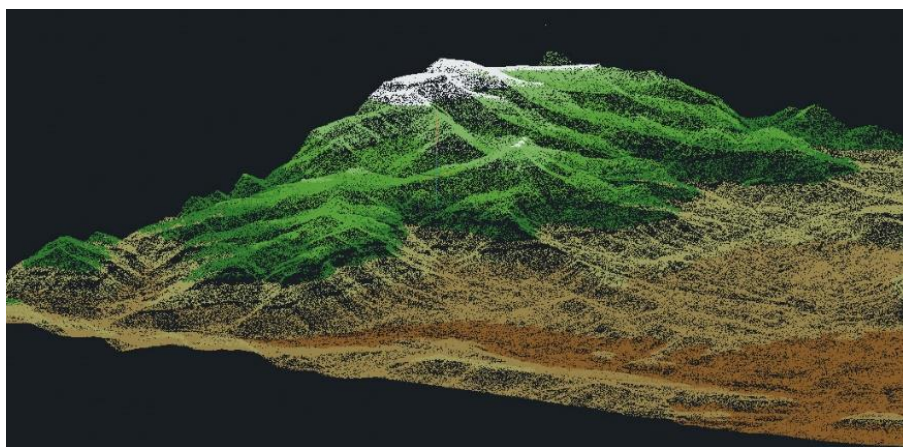


Рисунок 1.13 Реконструкция рельефа

К недостатку данной системы можно отнести высокую степень участия оператора при реконструкции сцены, т.к. система используется для отображения уже имеющихся данных и их редактирования.

ГИС Карта [61]. Пользовательские трехмерные модели создаются на основе введенных оператором данных. Система может использовать в качестве исходных данных следующую информацию:

- векторная карта в формате «.SXF» или «.SIT»
- классификатор карты в формате «.RSC»
- библиотека трехмерных изображений в формате «.P3D»

Возможно также использование:

- матрицы высот «.MTW»
- растровой информации «.RSW»

Для создания наиболее реалистичных моделей оператор должен обладать фотоснимками местности, что приводит к необходимости полевых работ, а конкретно фотографированию необходимых объектов или их частей.

Данный программный комплекс обрабатывает матричные данные о местности, представленные в открытом формате MTW. Матрицы высот могут содержать абсолютные высоты рельефа местности, относительные высоты объектов местности или сумму названных высот (Рисунок 1.14).

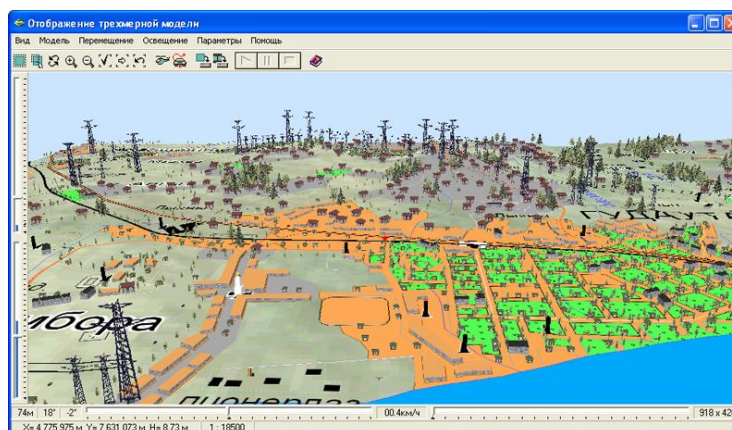


Рисунок 1.14 Сцена с рельефом и дополнительными объектами

Программа обладает богатым встроенным набором трехмерных моделей, которые могут быть использованы для условного или приближенного обозначения реальных дополнительных объектов сцены.

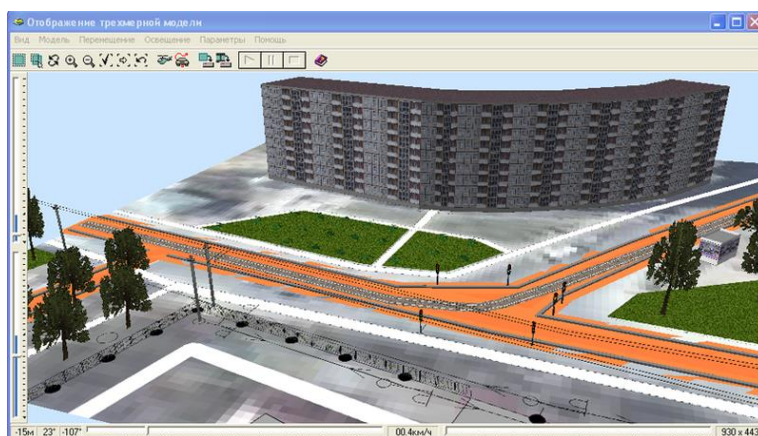


Рисунок 1.15 Результат реконструкции сцены

Autodesk 3ds Max [4]. Данная программа является одним из самых распространённых редакторов трёхмерной графики. Её инструментарий может быть использован как для реконструкции, так и для создания совершенно новых сцен. Интерфейс программы изображен ниже (Рисунок 1.16).

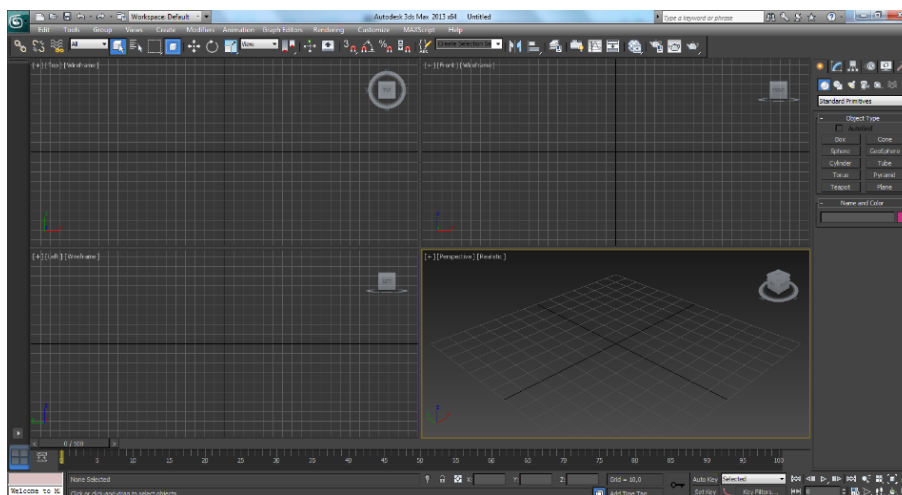


Рисунок 1.16 Главное окно

Одним из полезнейших инструментов данной программной системы для реконструкции городских объектов по топографическому плану является модификатор Extrude. С его помощью можно произвести «выдавливание» полигонов или линий, образуя таким образом стены объекта. На рисунках

ниже (Рисунок 1.17, Рисунок 1.18) изображён созданный контур и «выдавленный» трёхмерный объект.

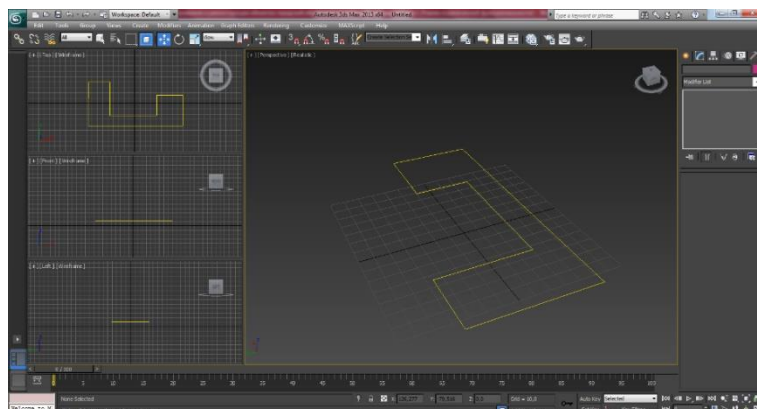


Рисунок 1.17 Созданный контур

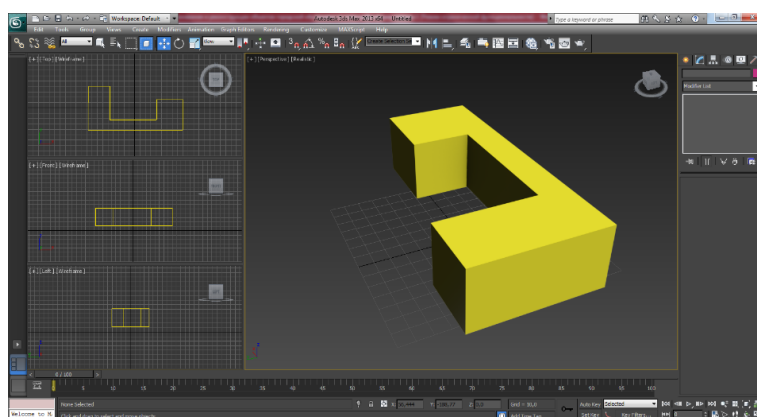


Рисунок 1.18 Построенная модель здания

Рассмотренным выше способом создается вся трёхмерная сцена и производится её текстурирование. Все действия не могут быть произведены автоматически, в том числе и текстурирование (Рисунок 1.19). Поэтому, для реконструкции протяженной сцены требуется большое количество времени.

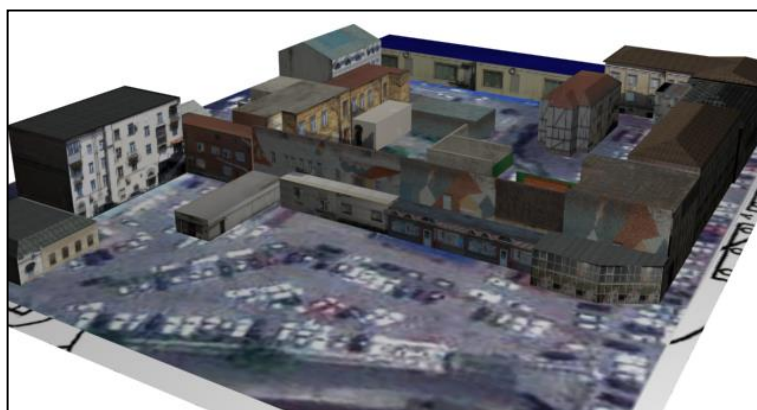


Рисунок 1.19 Текстурированная сцена, созданная в 3ds Max

Photomodeler [35]. Данная программа позиционируется разработчиками как программа для трехмерной реконструкции на основе моделирования для точного измерения и визуализации 3D-моделей в области машиностроения, архитектуры, кино, судебной медицины, и многих других.

Работа с программой начинается с загрузки фотоизображений объекта. Их должно быть как минимум два. Далее необходимо откалибровать камеру и изображения. Для калибровки камеры необходимо сфотографировать лист бумаги под углом в 45 градусов, а на полученном изображении выделить границы этого листа, по этим данным программа рассчитает фокус и центральную точку. Далее на загруженных изображениях находим угловые точки и связываем их гранями. Затем вручную устанавливаем сопоставления найденных точек на всех изображениях. Потом построенные грани объединяем в полигоны (Рисунок 1.20).

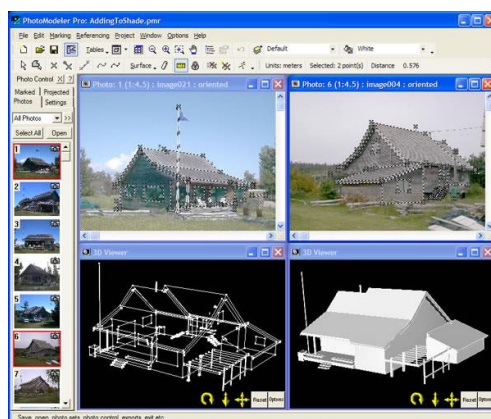


Рисунок 1.20 Общий вид программы Photomodeler

Текстурирование происходит автоматически. Привязку к физическим размерам объекта производим, выделяя любую грань и указывая ее длину в метрах, миллиметрах и т.д. Программа позволяет просматривать и сохранять полученные трехмерные объекты.

Вывод: это программа, предназначенная для реконструкции любых объектов, достаточно универсальна, но предполагает значительный объем ручной работы даже для объектов простой геометрической формы.

Векторизаторы. Векторизаторы – программное средство для выполнения растрово-векторного преобразования данных. Векторизаторы, как правило,

включают функции обработки растровых изображений, ручного ввода и полуавтоматической трассировки контуров и линий объектов. Не существует универсального метода и программного комплекса, позволяющего распознать все объекты топоплана.

Ниже представлены некоторые наиболее популярные векторизаторы для ГИС, позволяющие выполнять векторизацию топоплана.

Easy Trace [21]. Данный продукт позволяет обрабатывать реальные, низкокачественные и сильно изношенные материалы (Рисунок 1.21). К тому же пакет позволяет распараллеливать работу с гарантией последующей интеграции данных в единое целое.

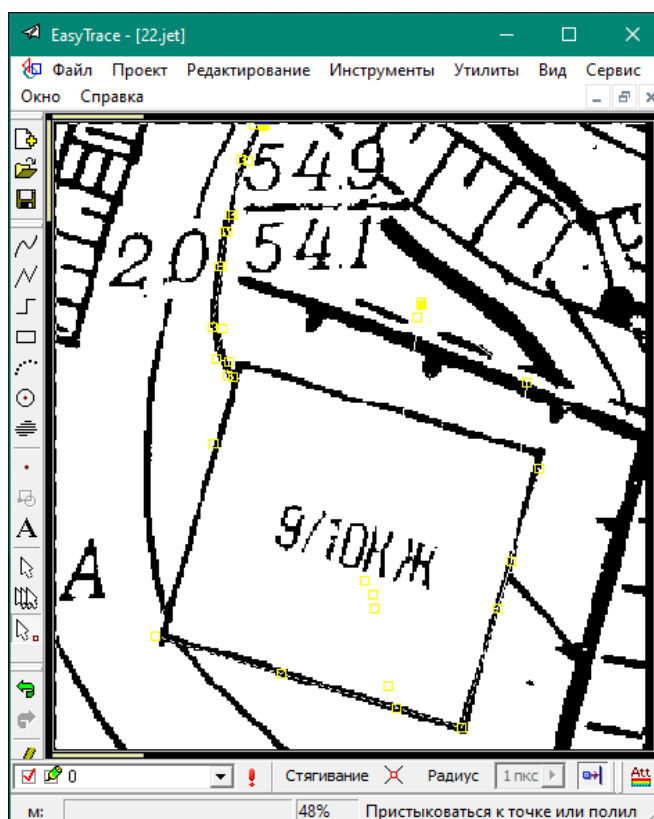


Рисунок 1.21 Общий вид программы Easy Trace

В основе технологии, реализованной в пакете, лежит мозаичное растрово-векторное поле практически неограниченных размеров. Многослойная растровая мозаика может состоять из произвольной комбинации растров различной цветности и масштаба. Количество векторных слоев не ограничено, в свою очередь, каждый слой может содержать до миллиона объектов. В пакете присутствуют

инструменты для подготовки топоплана к векторизации, которая может выполняться как в автоматическом, так и в полуавтоматическом режиме.

VPmap [47]. VPmap – серия продуктов компании Softelec, предоставляющая оператору интерактивные инструменты для создания, редактирования пространственных данных, обновления информации, полученной из сканированных карт, спутниковой, аэрофотосъемки и переноса её в ГИС.

Пакет даёт возможность получать детальные данные с отсканированных цветных и черно-белых картографических материалов для использования в проектах кадастровых работ, землеустройства, проведения топографических съемок, геологических и инженерных изысканий и т. п. Инструментарий позволяет решать широкий круг задач векторизации растровых изображений с автоматическим замыканием полигонов, пропиской их цветового оформления, контролем совмещения границ, присваивать объектам атрибуты и решать ряд других задач. Пакет способен значительно снизить расходы на реализацию ГИС-проектов, опирающихся на доступные бумажные картографические материалы, обеспечивает возможность сшивки растровых карт, цветоделения, создания мозаик, преобразования систем координат и проекций. В нем поддерживаются основные ГИС-форматы, такие как GeoTIFF, DWG, SHP, MIF и DGN.

AutoCAD Raster Design [3]. AutoCAD Raster Design позволяет очищать изображения от случайных пикселей, появившихся в результате сканирования нечетких рисунков и светокопий (Рисунок 1.22). В пакете присутствует инструментарий для исправления искажений пропорций, дорисовки растра, улучшения качества изображений на основе регулирования параметров изображения и тоновой настройки, нелинейного трансформирования изображений.

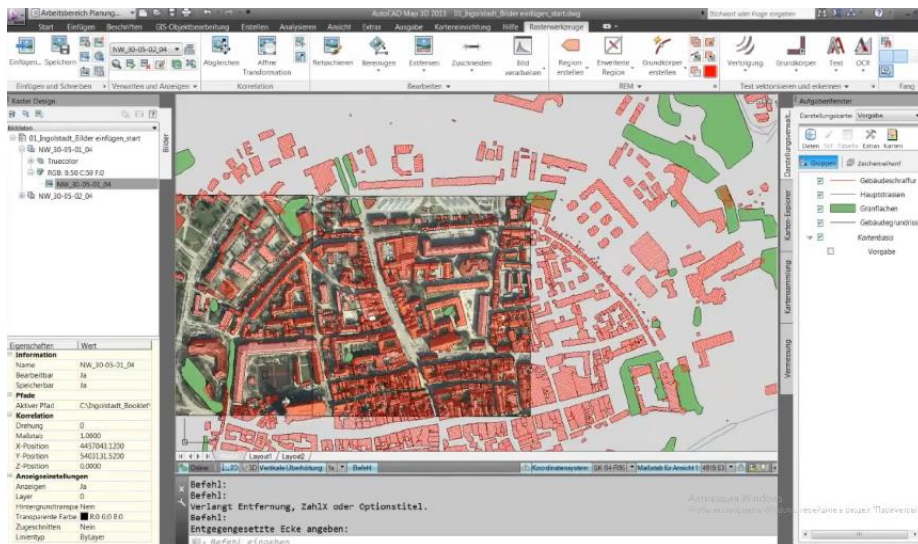


Рисунок 1.22 Общий вид программы AutoCAD Raster Design

Пакет обладает широким спектром функциональности и позволяет экспортировать данные в ГИС-системы.

1.8. Выводы по главе

Описанные выше методы и присутствующие в настоящее время на рынке программные среды не могут быть в полном объеме применены для реконструкции сцены городского пространства по топоплану. Они не позволяют обрабатывать протяженные сцены с помощью автоматического выделения контуров с топоплана. В условиях, которые диктуются современной практикой реконструкции, вышеизложенные методы требуют значительного вклада ресурсов оператора в процесс моделирования. Не во всех программных продуктах присутствует возможность соотносить и наносить текстуру из фотоснимков реальных объектов на здания сцены.

В связи с этим поддерживается актуальность проведения дальнейших исследований и разработок, направленных на повышение эффективности и точности разрабатываемых систем в рассматриваемой области.

Глава 2. Выделение контуров на топоплане

В данной главе представлена методика выделения информации на бинарном изображении. Описывается алгоритм, с помощью которого происходит распознавание объектов топоплана. Предлагается модификация алгоритма.

2.1. Основные определения

Топоплан – это уменьшенное подобное изображение горизонтальной проекции участка поверхности Земли с находящимися на ней объектами [67]. План не составляется на очень большую территорию, т.к. сферическая поверхность Земли не может быть развернута в плоскость без складок и разрывов. Изображение Земли на плоскости, уменьшенное и искаженное вследствие кривизны поверхности, называют картой. Таким образом, и план, и карта – это уменьшенные изображения земной поверхности на плоскости. Различие между ними состоит в том, что при составлении карты проецирование производится с искажениями поверхности за счет влияния кривизны Земли, а на плане изображение получают практически без искажения. Топографические планы создают в масштабах 1:5000, 1:2000, 1:1000 и 1:500 [68, 101].

На топоплане изображаются чертежи рельефа местности и прочих объектов с их метрическими и техническими характеристиками, выполненные в утвержденных условных знаках (Рисунок 2.1). Если известно, в каком масштабе выполнен топоплан, то на нем возможно измерить координаты и линейные размеры объектов.

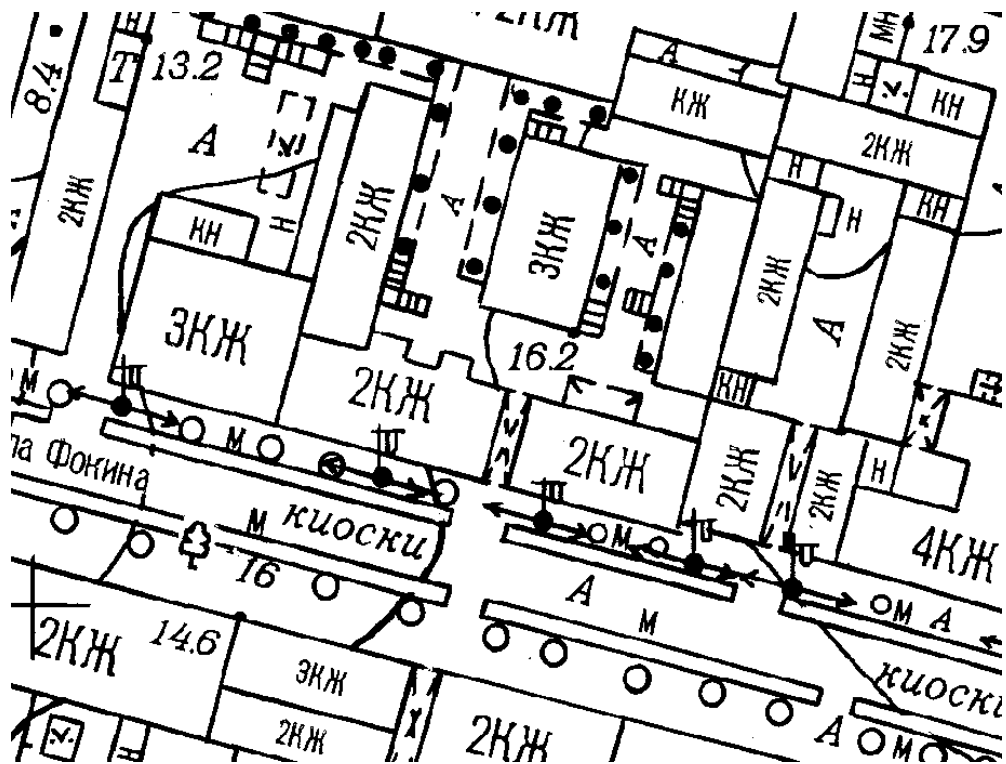


Рисунок 2.1 Фрагмент топоплана

Топографические планы являются основными исходными данными на всех этапах архитектурно-планировочного и инженерного проектирования [88, 89, 94]. С их наличием появляется возможность произвести анализ и оценку природных особенностей района. С помощью измерений по топографическому плану [64] можно получить координаты изображенных объектов, расстояния между ними, размеры и другие параметры, измерение которых в реальных условиях представляет трудоемкую задачу.

Основные объекты [97], изображаемые на топоплане, это:

- Здания и сооружения. Все здания и сооружения имеют подписи с информацией о количестве этажей, материале и принадлежности к жилым постройкам.
- Растительность и грунт. На топоплане указывают кустарники, деревья, клумбы и т.д. Растительность может обозначаться вплоть до указания названия дерева (дуб, береза и т.п.). Грунт обязательно обозначается, если растительность отсутствует.

- Дорожная сеть. Указываются автомобильные, железнодорожные, трамвайные или пешие пути.
- Инженерные сети. Наземные и подземные коммуникации: ЛЭП, газ, водопровод, канализация, кабели связи и т.д.

Контур – упорядоченная последовательность точек топоплана, где каждая точка связана с двумя другими. Никакие две точки не образуют пересечения соединяющих их отрезков. При этом, соединенные последовательно точки должны максимально точно описывать форму объекта. Количество вершин может сильно различаться в зависимости от того, к какому объекту он принадлежит. Как правило, контуры зданий имеют небольшое количество точек из-за того, что в большинстве случаев в действительности у них относительно небольшое количество стен. В силу своей геометрии точки на стенах не несут смысловой нагрузки, поэтому от них необходимо избавляться. В тоже время другие объекты, такие как условные обозначения или текстовая информация, содержат больше точек контура, т.к. они плотно расположены друг к другу и каждая точка важна для понимания формы объекта.

2.2. Схема реконструкции

В ходе проведенного исследования по обзору существующих методов, подходов, программных продуктов для реконструкции протяженных сцен городской обстановки был выработан следующий подход для решения подобной задачи для топопланов с масштабом 1:2000:

1. Выделение контуров. На данном этапе происходит поиск контуров всех объектов, присутствующих на топоплане. Поиск контуров осуществляется с помощью волнового алгоритма.
2. Поиск среди них контуров зданий, для этого необходимы следующие методы:

- 2.1. Фильтрация. Полученный набор контуров необходимо обработать, чтобы оставить только контуры тех объектов, которые представляют некоторую ценность для реконструкции (здания, текст).
- 2.2. Распознавание текста. Информация о типе здания может быть получена только из его внутренней информации. Поэтому внутри зданий ищется текст, который дальше преобразуется в характеристики здания. Для выделения текста также используется волновой алгоритм.
- 2.3. Логический анализ текста. Выделенный текст подвергается синтаксическому анализу для его проверки на допустимые значения. К тому же на данном этапе текст может дополнительно корректироваться.
3. Выделение дополнительных объектов (фонари, светофоры и др). На топоплане помимо зданий присутствует дополнительная информация о других объектах, которые обозначаются условными знаками. По определенным рассчитанным характеристикам зданий производится поиск дополнительных объектов среди найденных контуров.
4. Построение трехмерной модели сцены по полученной информации. Здания в итоговую сцену добавляются в виде прямоугольных призм. Дополнительные объекты представлены в виде заранее заготовленных трехмерных моделей.
5. Получение данных о горизонтальной поверхности сцены. На основании пары точек, сопоставленных с парой точек топоплана определяются географические координаты множества точек топоплана. Высота над уровнем моря в каждой такой точке определяется с помощью Google Maps Elevation API сервиса.
6. Подшивка рельефа. Для того, чтобы здания и другие объекты естественно соприкасались с рельефом, происходит поднятие объектов до нужного уровня.

7. Получение фотоснимков с дополнительной информацией. Для текстурирования используются фотоснимки зданий. Оператор производит съемку с помощью специально разработанного приложения, позволяющего определить направление съемки, задать область текстуры здания на снимке, а также привязать место съемки к географическим координатам.
8. Нанесение реальных текстур. На основании привязанных к снимкам географических координат определяется место съемки на топоплане, а также направление. Происходит автоматический поиск здания, которое было сфотографировано на текстуре.
9. Нанесение стандартных текстур. Здания, которые не содержат текстур с фотоснимков покрываются стандартными текстурами. Т.е. такими, которые соответствуют определенному типу строения.
10. Вывод полученной сцены в распространенный формат. Трехмерная текстурированная сцена экспортируется в файл с расширением «.x», который впоследствии можно просмотреть в редакторе трехмерной графики.

В настоящей работе применяется следующая последовательность шагов для реконструкции сцены городского пространства:

Подготовка топоплана. На данном этапе происходит преобразование топоплана в файловый формат, который будет подан на обработку алгоритма выделения контуров. Топографическое изображение может иметь различные форматы (jpeg, bmp, png). По своей структуре изображения топопланов могут отличаться. Некоторые могут содержать легенду, название, обрамляющую рамку и т.п. Эта информация должна быть убрана с изображения, как и мелкие шумы. Топоплан должен быть преобразован в черно-белое изображение формата bmp. Это обеспечит более точный поиск контуров на нем.

Отсканированные изображения имеют ряд проблем. В результате сканирования могут появляться дефекты в виде черных пятен между объектами

или, наоборот, разрывы в линиях. Из-за этого часто требуется ручная обработка, которая позволяет подготовить топоплан к дальнейшей обработке. В предложенном ПО (Приложение В. Описание разработанного программного комплекса) есть весь необходимый инструментарий, однако, такая обработка возможна в любом стороннем графическом редакторе. Исходный топоплан и результат его обработки показан ниже (Рисунок 2.2).

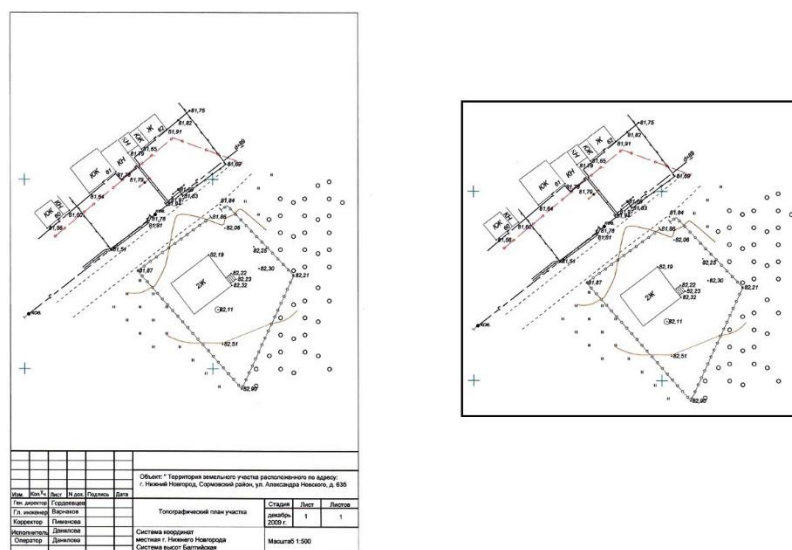


Рисунок 2.2 Топоплан исходный (слева) и топоплан обработанный (справа)

Выделение контуров на топоплане. Основная информационная ценность топоплана – это расположение зданий с информацией о них и других объектов, таких как электросети, фонари, светофоры, дороги, деревья и т.д. Поэтому важно найти все объекты на сцене и определить их тип. Результатом данного этапа должен быть массив контуров, каждый из которых является упорядоченной последовательностью точек.

Существует ряд алгоритмов и методик выделения контуров объектов на цифровых фотоснимках. Часть этих методов (методы Превитта, Собеля, Робертса, Канни-детектора) основаны на фильтрации изображения некоторым окном-матрицей и нахождении отклика в каждой точке изображения. Достоинствами данных методов является их универсальность, т.к. они позволяют обрабатывать не только черно-белые изображения, но и цветные. Однако, данные алгоритмы не

могут быть применены к выделению зданий на топоплане, т.к. они не способны получить упорядоченную последовательность точек контура.

Получить такую последовательность возможно с помощью волнового алгоритма, который предназначен для поиска пути выхода из лабиринта. С помощью, предложенной автором модификации этого алгоритма, будет происходить выделение всех точек контура объекта, а также их упорядочивание. Последовательно во всех возможных точках топоплана будет запущен волновой алгоритм, который сформирует массив всех контуров на изображении. Каждый контур изначально будет представляться в виде набора неупорядоченных точек. Но после повторного применения волнового алгоритма к этим точкам, будет сформирован готовый контур. Подробно этот процесс описан в п. 2.3.

Классификация контуров по городским объектам. Выделенные на топоплане контуры могут принадлежать различным объектам. Часть из них может являться результатом выделения незначающей информации. Например, изолинии рельефа, линии электропередач, а также их пересечения с дорогами или линиями контуров дадут ложные замкнутые контуры, которые нельзя классифицировать как отдельный объект. Такие контуры необходимо исключить из общего набора контуров. Остальные требуется рассортировать по видам, т.к. они могут быть контурами зданий, других объектов городской обстановки или контурами текстовой информации.

На данном этапе требуется определить уникальные характеристики каждого класса объектов. Некоторые требования к каждому классу объектов можно поставить исходя из условий реального окружения. Так, например, здания могут иметь строго определенную форму. Т.е. количество прямых углов в контуре должно преобладать, количество вершин контура более трех, но не более какого-то заданного количества.

Распознавание внутренней информации зданий. После классификации всех выделенных контуров требуется определить дополнительную информацию, которая располагается внутри контуров зданий. Эта информация содержит данные

о том, сколько этажей у здания, из чего оно сделано и жилое ли оно. Количество этажей здания напрямую влияют на его геометрическую форму на сцене. Остальные параметры внутренней информации требуются для определения внешнего вида здания, т.е. текстуры, которой оно должно обладать.

Внутренняя информация о зданиях состоит из условных символьных обозначений. Количество этажей обозначается цифрами, а другая информация буквами. Выделение символов осуществляется также с помощью волнового алгоритма. В результате работы алгоритма образуются отдельные объекты, каждый из которых необходимо будет классифицировать как определенный символ. После распознавания всей надписи внутри здания требуется проверить её на соответствие шаблонам, которые приняты для обозначения внутренней информации. В результате проверки надпись будет скорректирована и на её основании будет получена информация о количестве этажей и типе здания.

Построение 3D-модели сцены. Реконструированная сцена должна содержать трехмерные модели всех выделенных на ней объектов. Топоплан не дает информации о точной геометрической трехмерной форме зданий, поэтому в качестве объемной модели здания будет использоваться прямая призма, с основанием, обозначенным на топоплане. На основании полученных данных о количестве этажей и условной высоте одного этажа будет определяться высота всей модели.

Другие объекты сцены, такие как деревья, кусты, светофоры и уличные фонари, обозначаются условными знаками. В связи с этим установить их реальные размеры и геометрическую форму не представляется возможным. Геометрическая форма таких объектов неизвестна, поэтому каждому из них сопоставляется определенная условная шаблонная трехмерная модель. Линейные размеры таких моделей могут быть заданы относительно высоты одного этажа здания.

Одним из главных компонентов сцены, которые придают ей максимальную информативность и реалистичность, является рельеф местности. На топопланах, изображающих большую территорию местности, перепады уровня высот особенно

заметны и значимы. Топоплан содержит информацию об уровне высот. На нем присутствуют геодезические линии и числа, обозначающие высоту над уровнем моря. Однако, из-за наличия большого количества объектов на топоплане, выделить эти данные проблематично. Но, для построения рельефа, можно воспользоваться сторонними базами данных (Google Maps, GMTED2010, ASTER GDEM2, NextMap World 30). Поэтому топоплан может быть привязан к географическим координатам, обеспечив тем самым данные об уровне его высот. Информация о рельефе местности получается на основании двух заранее известных географических координат, которые соответствуют известным точкам на топоплане. На основании этих данных строится триангуляционная поверхность всей территории. Высота в каждой точке этой поверхности вычисляется путем обращения к сервису *Google Maps Elevation API*. В сервис передаются рассчитанные по двум исходным географическим точкам координаты каждой вершины поверхности.

Текстурирование сцены. Поскольку реконструируемые здания будут построены весьма условно, то для увеличения реалистичности сцены будет использоваться текстура. Для это требуется, чтобы каждый объект сцены имел текстуру, максимально приближенную к реальному объекту.

Внешний вид здания обладает большей информативностью, чем его геометрическая форма, т.к. на зданиях, в большинстве случаев, присутствует дополнительная информация, которая отсутствует на топоплане (названия, адреса, цветовая гамма и т.п.). Эта информация служит дополнительным ориентиром на территории сцены. В качестве текстур здания могут служить фотографии его сторон, т.к. в подавляющем большинстве случаев они имеют прямоугольные формы. В условиях городской постройки не всегда есть возможность получить идеальные фотоснимки стен зданий, которые можно было бы использовать для текстурирования. На фотографиях необходимо выполнить коррекцию перспективы, выделив на них только те участки, которые относятся к стенам

зданий. Каждый такой снимок должен быть обработан перед нанесением на модель здания.

Современные технологии (использующие GPS, ГЛОНАСС) позволяют получить фотоснимок с географическими координатами точки, в которой он был сделан, а также информацию о направлении съемки. На основании этих данных, возможно сопоставление этих фотоснимков со стенами конкретных зданий на топоплане.

Если нет возможности сделать снимки стен зданий, то текстурирование таких строений осуществляется по-иному. Все здания обладают информацией о своем типе. Каждый тип, как правило, характеризуется определенным внешним видом. Таким образом, каждому типу здания можно сопоставить определенный набор условных текстур, которые однозначно бы характеризовали этот тип (Рисунок 4.20).

На рельеф сцены также может быть нанесена текстура. На основании уже имеющихся географических координат точек на топоплане наносится текстурная карта территории, полученная со спутника, путем обращения к сервису *Google Maps API*.

2.3. Волновой алгоритм

Топоплан представляет собой бинарное черно-белое изображение, содержащее очертания зданий, улиц, информацию о числе этажей в зданиях, данные о ландшафте, различных городских объектах и т.д. Каждый объект является совокупностью черных и белых точек. Такие последовательности будут называться контурами. Волновой алгоритм Ли [32] позволяет обработать топоплан таким образом, что в результате его работы будут получены все внешние и внутренние точки, опоясывающие все объекты.

Волновой алгоритм Ли изначально предназначен для поиска кратчайшего пути в дискретном лабиринте, однако, принцип его работы может быть

использован и для того, чтобы выделить контуры объектов на топоплане. Принцип работы волнового алгоритма заключается в распространении волны в дискретной области из некоторой точки и выделении всех точек, с которыми эта волна столкнулась.

Основные понятия, используемые в работе волнового алгоритма:

- *Карта* – дискретная область, содержащая числовые значения в каждой своей ячейке.
- *Ячейка* – элемент карты. Содержит числовое значение, которое характеризует её.
- *Волна* – совокупность действий по переходу от одной ячейки к другой.
- *Пустая ячейка* – ячейка, в которой значение равно 0. По пустым ячейкам происходит распространение «волны».
- *Ячейка шага* – ячейка, которая была пустой, но которая изменила свое значение при распространении волны.
- *Стена* – ячейка, в которой содержится значение 1. Эти ячейки не участвуют в распространении волны, а являются барьером для неё.
- *Точка контура* – ячейка стены, с которой столкнулась волна. При этом значение в ячейке стене будет меняться на 2. Эти точки являются составляющей того множества, которое необходимо определить на этапе выделения контуров объектов.
- *Исходная ячейка* – ячейка начала распространения «волны». Её значение равно 3.
- *Текущее значение волны* – это максимальное значение среди всех ячеек.

Выполнение волнового алгоритма

Шаг 1. Работа волнового алгоритма начинается с подготовки карты. Карта должна содержать только пустые ячейки и стены. Карту необходимо окружить стенами, чтобы волна не распространялась за её пределы.

Шаг 2. На карте необходимо указать исходную ячейку, от которой будет распространяться волна. Исходная ячейка должна быть изначально пустой. Текущее значение волны становится равным значению исходной ячейки, т.е. 3 (Рисунок 2.3).

0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0
0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0
0	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
0	0	1	0	1	0	1	0	1	1	0	0	1	0	0	0
0	0	1	0	1	1	1	0	1	0	1	0	1	0	0	0
0	0	1	0	0	0	1	0	1	0	1	0	1	1	1	1
0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0
0	0	1	1	1	1	1	1	0	0	3	1	0	0	0	0
0	1	0	0	0	0	0	0	1	0	0	0	0	1	0	0
1	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0
0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0

Рисунок 2.3 Фрагмент карты со значениями в ячейках

Шаг 3. Осуществляется поиск всех ячеек на карте, значение которых равно текущему значению волны.

Шаг 4. Выбираются соседние ячейки вокруг тех, которые были найдены на шаге 3. Волна может распространяться двумя способами: по четырем направлениям или по восьми. В первом случае в качестве соседних ячеек выбираются те, которые расположены ближе всего к найденным на шаге 3 по горизонтали и вертикали. Во втором случае в качестве соседних берутся еще и диагональные ячейки.

Осуществляется проверка соседних ячеек относительно исходной. Если соседняя ячейка является пустой, то в неё записывается значение, равное значению пустой ячейки + 1. Если соседняя ячейка является стеной, то она становится точкой контура и в неё записывается значение 2.

Шаг 5. Текущее значение волны увеличивается на 1.

Шаг 6. Если на шаге 4 была найдена хотя бы одна пустая ячейка, то алгоритм повторяется с шага 3. Таким образом будут обработаны все пустые ячейки в

контуре, внутри которого находится исходная точка. Процесс распространения волны показан ниже (Рисунок 2.4).

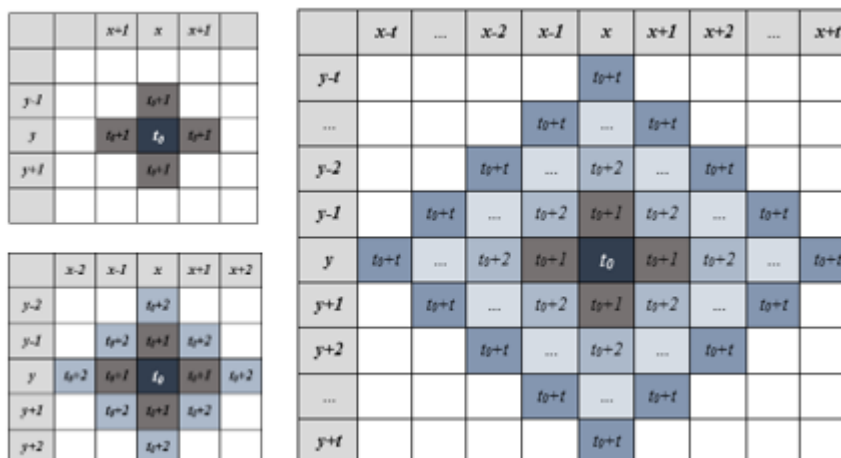


Рисунок 2.4 Процесс распространения волны на карте.
 t_0 , x , y – значения и координаты исходной ячейки на карте

Шаг 7. Волна не сможет распространяться дальше, если на её пути нет пустых точек. Результатом будет карта, на которой будут присутствовать ячейки контуров (Рисунок 2.5). Выделенные таким образом ячейки могут образовывать несколько разделенных групп ячеек контуров. Две группы будут считаться разделенными, если никакая ячейка одной группы не является соседней для любой ячейки из другой группы. При этом каждая группа ячеек в итоге является контуром. Один из таких контуров всегда будет объединяющим для остальных.

0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
1	1	1	2	2	2	2	2	2	2	2	2	1	0	0	0
0	0	2	16	15	14	13	12	13	12	11	10	2	0	0	0
0	0	2	15	2	15	2	11	2	13	1	9	2	0	0	0
0	0	2	14	2	16	2	10	2	2	9	8	2	0	0	0
0	0	2	13	2	2	2	9	2	7	2	7	2	0	0	0
0	0	2	12	11	10	2	8	2	6	2	6	2	1	1	0
0	0	2	11	10	9	8	7	6	5	4	5	2	0	0	0
0	0	1	2	2	2	2	2	5	4	3	2	0	0	0	0
0	1	0	0	0	0	0	2	6	5	4	5	2	0	0	0
1	0	0	0	0	0	0	2	7	6	5	6	2	0	0	0
0	0	0	0	0	1	1	1	2	2	2	2	1	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0

Рисунок 2.5 Результат распространения волны на карте

Все выделенные ячейки контуров необходимо упорядочить, чтобы они образовывали явную последовательность точек, по которым можно восстановить

форму объекта на топоплане. Для построения последовательности точек контура также используется волновой алгоритм с некоторыми дополнениями.

Примеры выделения контуров. Волновой алгоритм был применён к топоплану размером 277x177 пикселей на Рисунок 2.6а. Была указана начальная точка **S** внутри контура здания. Белые точки соответствуют пустым ячейкам, а чёрные – стенам. В результате был получен набор из 771 точки. На Рисунок 2.6б красными точками помечены найденные, синими – точки волны. Время работы алгоритма менее одной десятой секунды.

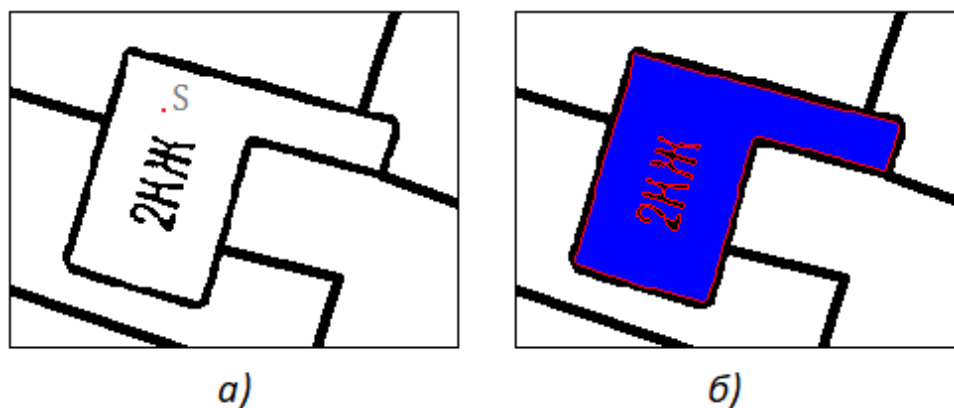


Рисунок 2.6 Применение волнового алгоритма. а) Исходный топоплан. б) Топоплан с применённым алгоритмом

Применение волнового алгоритма к топоплану на Рисунок 2.7, размеры которого 751x1054 пикселей, позволило выделить 5573 точки за время 0.21 секунд.

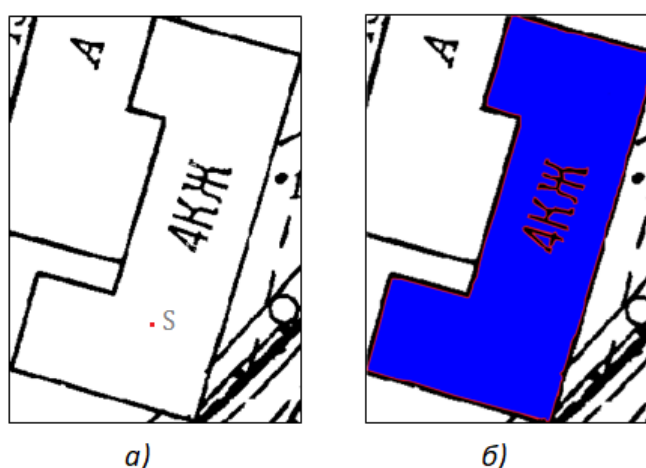


Рисунок 2.7 Применение волнового алгоритма. а) Исходный топоплан. б) Топоплан с применённым алгоритмом

Алгоритм восстановления контура по набору точек. Восстановление контура происходит на основании того набора данных, который был получен в результате работы волнового алгоритма для одного объекта. На топоплане пока присутствует только набор точек, который пока не упорядочен в один контур. Для группировки точек используется следующий алгоритм.

Шаг 1. На топоплане выбирается ячейка контура (значение в ней равно 2). Ячейка должна быть выбрана из внешней группы точек, т.е. группы, которая объединяет другие группы, если их несколько. Выбрать ячейку из внешней группы можно путем последовательного перебора строк и столбцов топоплана. При таком подходе первая попавшаяся ячейка контура будет гарантированно принадлежать внешней группе. Выбранная ячейка становится исходной для выполнения волнового алгоритма. Однако в данном случае, её значение будет равно не 3, а -3. Это позволяет отличить эту исходную ячейку и ту, которая была выбрана при выделении ячеек контура. Новая исходная ячейка становится текущим значением волны.

Шаг 2. На топоплане ищутся ячейки, значение в которых равно текущему значению волны.

Шаг 3. Для каждой ячейки, обнаруженной на шаге 2, запускается волна по восьми направлениям. Запуск волны по восьми направлениям обусловлен тем, что данный вариант обеспечивает гарантированное распространение волны по ячейкам, которые соседствуют только по диагонали.

В данном случае роль стен уже будут играть все ячейки со значением отличным от 2. А всеми пустыми ячейками будут считаться ячейки со значением 2. Осуществляется проверка соседних ячеек относительно найденных. Если соседней ячейкой окажется ячейки со значением 2, то её значение будет изменено на текущее значение волны - 1. Обратный счет также обусловлен тем, чтобы значения волны, участвующие при выделении ячеек контура, не совпали со значениями волны, участвующими при построении контура.

Шаг 4. Текущее значение волны уменьшается на 1.

Шаг 5. Если на шаге 3 была обнаружена хотя бы одна ячейка со значением 2, то повторяется шаг 2.

Шаг 6. Волна перестанет распространяться, когда на её пути не окажется ни одной ячейки со значением 2 (Рисунок 2.8). В этом случае необходимо спуститься по обратным путям из той ячейки, на которой остановилась волна до исходной ячейки. Таких путей будет два. На данном шаге запоминается текущая ячейка, на которой остановилась волна.

0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
1	1	1	-3	-4	-5	-6	-7	-8	-9	-10	-11	1	0	0
0	0	-4	16	15	14	13	12	13	12	11	10	-12	0	0
0	0	-5	15	2	15	2	11	2	13	2	9	-13	0	0
0	0	-6	14	2	16	2	10	2	2	9	8	-14	0	0
0	0	-7	13	2	2	2	9	2	7	2	7	-15	0	0
0	0	-8	12	11	10	2	8	2	6	2	6	-16	1	1
0	0	-9	11	10	9	8	7	6	5	4	5	-17	0	0
0	0	1	-10	-11	-12	-13	-14	5	4	3	-18	0	0	0
0	1	0	0	0	0	0	-14	6	5	4	5	-19	0	0
1	0	0	0	0	0	0	-15	7	6	5	6	-20	0	0
0	0	0	0	0	1	1	1	-16	-17	-18	-19	1	0	0
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0

Рисунок 2.8 Распространившаяся волна по контуру

Шаг 7. Около текущей ячейки выбирается любая соседняя, у которой значение больше значения текущей на 1.

Шаг 8. Координаты текущей ячейки запоминаются и добавляются в последовательность точек контура.

Шаг 9. В текущую ячейку устанавливается значение 1. Это обеспечит невозможность выбора той же самой ячейки при повторном спуске.

Шаг 10. Текущей ячейкой становится выбранная соседняя ячейка на шаге 7.

Шаг 11. Если значение в текущей ячейке равно тому значению, которое было равно у исходной ячейки перед началом распространения волны (т.е. равно -3), то спуск заканчивается.

Шаг 12. Выполняется второй спуск из точки, где закончилось распространение волны (Шаг 6). Для этого повторяются шаги с 7 по 11. На шаге 8 координаты текущей ячейки должны записываться в начало последовательности, чтобы эта последовательность была непрерывной.

В результате работы алгоритма будет получена последовательность точек, которая будет образовывать контур фигуры на топоплане. Несмотря на то, что при спусках ячейкам контуров были установлены другие значения, на топоплане всё еще могут оставаться ячейки контуров со значением 2. Это возможно при наличии нескольких групп ячеек контуров.

По окончании выделения контуров необходимо очистить топоплан от ячеек, которые были посещены волной. Для этого всем ячейкам, у которых значение отличается от 1 или 0 устанавливается значение 1. Это поможет при дальнейшей обработке топоплана не обрабатывать посещенные ранее волной ячейки.

После обработки области топоплана, в котором была указана исходная точка будет образован набор контуров. В наборе может быть и один контур. Это будет означать, что он не имеет каких-либо вложенных контуров. Таким образом, указанной точке внутри топоплана будет ассоциирован набор вида:

Внешний контур

- Внутренний контур №1

- Внутренний контур №2

- Внутренний контур №N.

Подобно тому, как были выделены контуры на основании одной указанной на топоплане точки, выделяется и все остальные. Для этого на топоплане последовательно перебираются ячейки. Если среди них найдена пустая ячейка, то она становится исходной для волнового алгоритма. Из неё запускается процесс поиска контуров.

В результате обработки топоплана будет получено множество контуров и их внутренние контуры.

2.4. Анализ быстродействия и модификация

Произведем анализ быстродействия алгоритма. Рассмотрим неидеальный вариант, когда топоплан имеет квадратную форму ($n \times n$), а в качестве исходной точки выбирается точка в одном из её углов.

На каждом шаге движения волны происходит анализ всего топоплана. Тогда на каждом шаге происходит анализ n^2 точек. Для заполнения всего топоплана волне необходимо сделать количество шагов, равное $2n-1$.

Таким образом, чтобы заполнить квадратный топоплан размером n^2 требуется проанализировать P_1 точек (2.1).

$$P_1 = (2n-1)n^2 = 2n^3 - n^2 \quad (2.1)$$

где P_1 – общее число точек для анализа, t – число шагов для того, чтобы достигнуть границы, n – размер области по горизонтали и вертикали.

Как видно из проиллюстрированных данных (Рисунок 2.9 и Рисунок 2.10), число анализируемых точек высокое даже при небольших размерах исходной области.

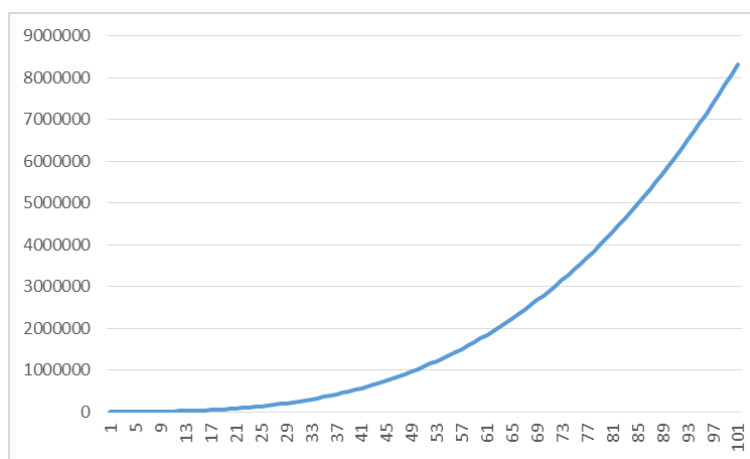


Рисунок 2.9 Зависимость количества анализируемых точек от размера топоплана. Подпись снизу – ширина и высота (n) области. Подпись слева – число анализируемых точек (P_1).

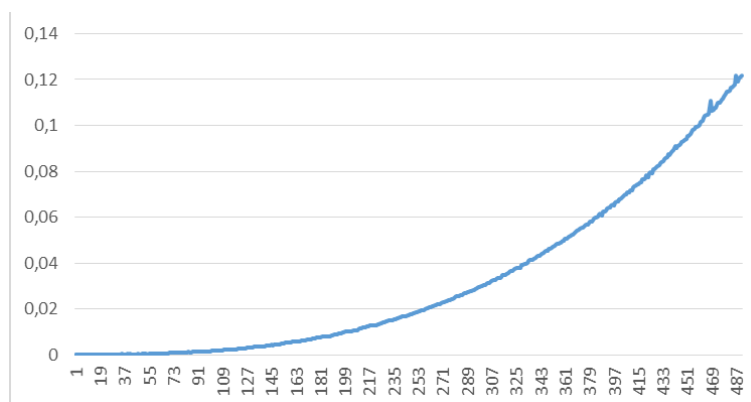


Рисунок 2.10 Зависимость времени (в секундах) работы алгоритма от длины и ширины исходной области

Поскольку алгоритму предстоит обрабатывать изображения большого размера, для уменьшения времени обработки была предложена модификация. Во время выполнения алгоритма приходится анализировать все точки области. Однако, можно заметить, что в этом нет необходимости, если рассматривать только те, которые были рассмотрены на прошлых шагах.

Пусть область Q – все ячейки внутри некоторого контура топоплана. Зададим область q , причём $q \in Q$. Её размеры будут регулироваться во время работы алгоритма по мере необходимости четырьмя величинами L, T, R, B . Параметры L и R устанавливают горизонтальные границы области q ($L < R$), а T и B – вертикальные ($T < B$) (Рисунок 2.11).

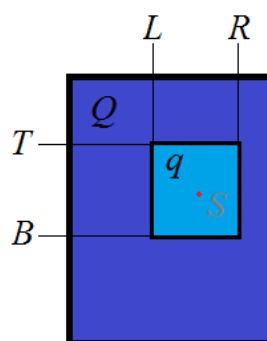


Рисунок 2.11 Область Q и находящаяся в ней q

При выборе точки S_{xy} начальные значения параметров будут равны (2.2)

$$L = x; R = x; T = y; B = y. \quad (2.2)$$

При таком подходе на каждом шаге будут рассматриваться только ячейки Q_{ij} , при $i \in [L-1, R+1]$ и $j \in [T-1, B+1]$. Если рассматриваемая ячейка Q_{ij} окажется

стеной, то проверяется, выходит ли одна из её координат за границы области q . Если так, то соответственная граница сдвигается до такой координаты.

Таким образом, если координата i рассматриваемой точки меньше x , то будет сдвинута граница L , если больше, то R . Аналогично с вертикалью, если текущая координата j меньше y , то будет сдвинута граница T , иначе B (Рисунок 2.12).

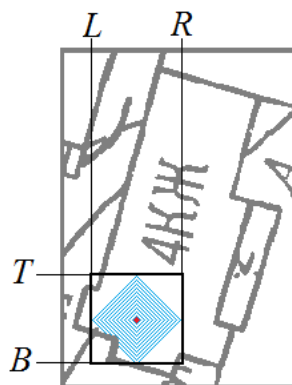


Рисунок 2.12 Распространение волны в ограниченной области

Рассчитаем выигрыш во времени для модификации алгоритма.

Рассмотрим квадратную область размера $n \times n$. На каждом шаге распространения волны граница сдвигается на одну единицу. Для модифицированного алгоритма худшим вариантом распространения волны будет распространение из угла топоплана. На первом шаге придётся проанализировать 2×2 точек, на втором – 3×3 и т.д. На распространение по всему топоплану алгоритму понадобится n шагов. В результате при модифицированном алгоритме будет проанализировано P_2 точек (2.3):

$$P_2 = \sum_{k=1}^n k^2 + (n-1)n^2 = \frac{1}{6}(8n^3 - 3n^2 + n) \quad (2.3)$$

где t – число шагов для достижения границы области; P_2 – общее число точек, которое будет проанализировано. Зависимость анализируемых данных от количества точек и времени показана ниже (Рисунок 2.13, Рисунок 2.14).

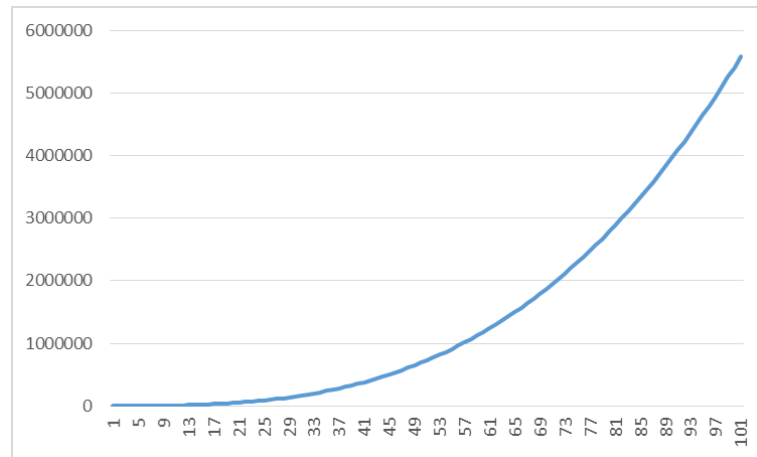


Рисунок 2.13 Число анализируемых точек, зависящее от размера области

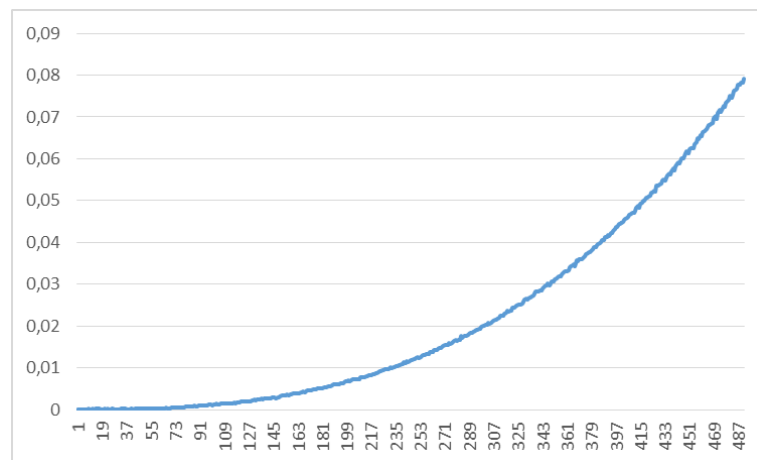


Рисунок 2.14 Зависимость времени работы алгоритма от размера области Q

Рассчитаем ускорение a , получаемое при работе двух алгоритмов. Ускорение будет рассчитываться по формулам (2.4) и (2.5):

$$a = \frac{P_1}{P_2} = \frac{2n^3 - n^2}{\frac{1}{6}(8n^3 - 3n^2 + n)} \quad (2.4)$$

$$\lim_{x \rightarrow \infty} \frac{2n^3 - n^2}{\frac{1}{6}(8n^3 - 3n^2 + n)} = 1,5 \quad (2.5)$$

Ниже (Рисунок 2.15) показана зависимость ускорения от размера топоплана.

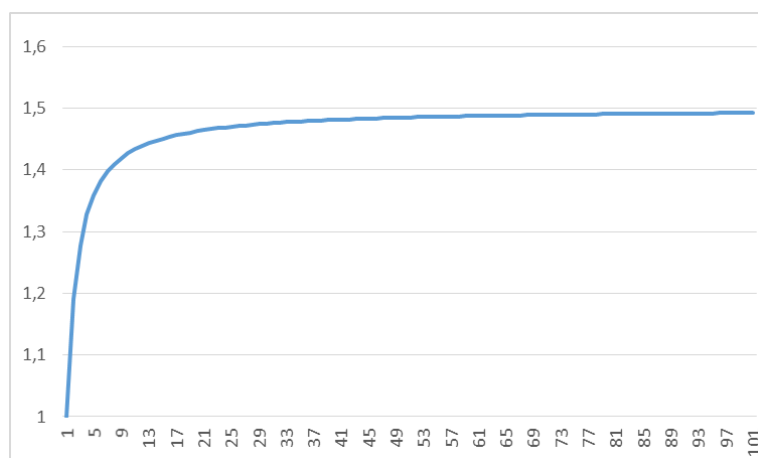


Рисунок 2.15 Ускорение модифицированного алгоритма.
Подпись снизу – размер топоплана. Подпись слева – ускорение.

Таким образом, модификация волнового алгоритма позволяет улучшить его в полтора раза.

2.5. Выводы по главе






Предложена методика выделения контуров на бинарном изображении, которая основывается на волновом алгоритме Ли. Методика позволяет находить все контуры на всем изображении. При этом каждый контур имеет известное положение и может быть отделен от остальных. Полученные контуры максимально точно описывают форму объекта. Это позволит далее классифицировать его более точно к одному из типов городских объектов. Особенностью методики является упорядочивание последовательности точек контуров, которая производится с помощью волнового алгоритма. Модификация волнового алгоритма позволила повысить скорость его работы в 1,5 раза.

Глава 3. Классификация контуров и объектов

После обработки топоплана с помощью разработанного волнового алгоритма получен набор контуров. Некоторые из них являются контурами определенных объектов на топоплане, остальные – это ложные контуры фигур, образованных в результате наложения некоторых объектов друг на друга. Так, например, пересекающиеся линии дорог и изолинии тоже образуют замкнутую фигуру, однако, никаким физическим объектом она не является. Информация о контуре этой фигуры не несет никакой полезной информации для реконструкции, потому необходимо из всего найденного массива контуров выделить только те, которые могут быть квалифицированы как определенные объекты городской сцены.

Топоплан содержит подробную информацию о территории местности. Каждый объект на нем обозначается одним или несколькими условными знаками. Ниже (Таблица 3.1) представлены некоторые объекты с их условными обозначениями.

Таблица 3.1. Объекты топоплана и их условные обозначения

Объект топоплана	Условное обозначение
Здание	
Уличный фонарь (два варианта)	
Светофор	
Клумба	
Дерево	

Каждый из приведенных объектов имеет свои уникальные характеристики, по которым возможно его классифицировать.

3.1. Выделение зданий

Здания – это основной тип объектов на топоплане. Определяется как замкнутый контур со служебной информацией внутри. Здание, как физический объект имеет некоторые ограничения и допустимые значения.

Форма основания здания. Выделим признаки того, что объект на топоплане является зданием:

- Форма объекта – это многоугольник. В реальности встречаются объекты иных форм (круглые, полукруглые и т.д.), однако всех их можно свести к форме многоугольника путем увеличения количества вершин.
- Количество сторон многоугольника больше трех и не больше какого-то определенного числа. В окружающем мире могут встречаться здания с тремя сторонами. Однако, такие объекты достаточно редки. А вот вероятность встретить на топоплане ложные объекты, которые в результате пересечения с другими объектами имели бы треугольную форму, достаточно высока. Поэтому будем пренебрегать данным частным случаем ради сокращения количества ложных распознаваний. Аналогичная ситуация со зданиями, у которых количество стен (равно как и вершин многоугольника) слишком большое. На практике было выявлено, что количество стен зданий обычно не превышает 20 штук.
- Количество прямых углов в многоугольнике значительно больше других. Если здание не имеет круглых форм, то в большинстве случаев его стороны перпендикулярны друг другу. Количество острых и тупых углов значительно меньше прямых. На практике установлено, что здания должно содержать больше 80% прямых углов.
- Внутри объекта обязательно есть символы, которые обозначают его внутреннюю информацию. Никаких других объектов внутри контура быть не может. Единственное исключение – замкнутый контур с буквой «А» – это не здание, а участок, покрытый асфальтом.

Примеры зданий различных форм на топопланах (Рисунок 3.1):

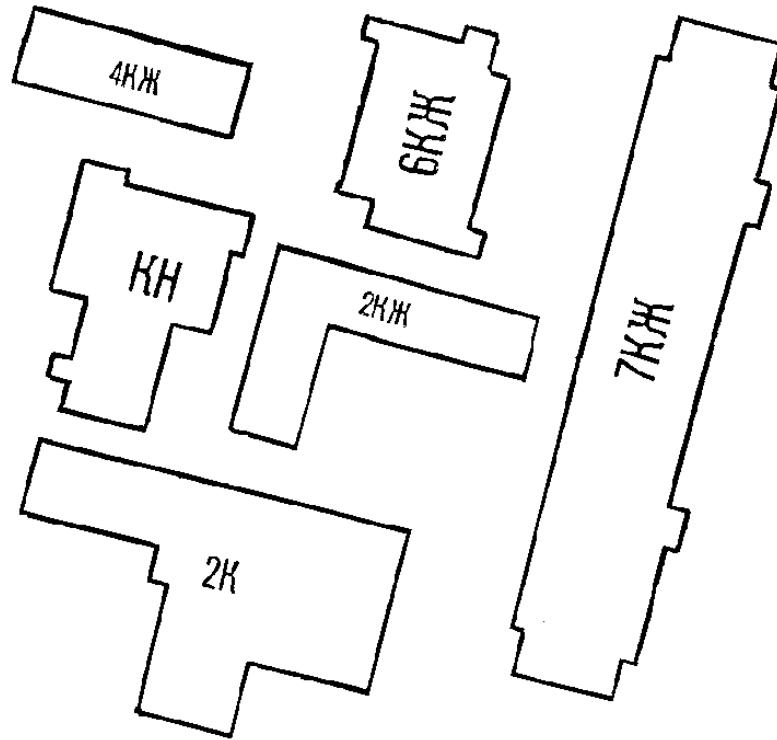


Рисунок 3.1 Примеры форм зданий на топопланах

Служебная информация внутри контура. Типы зданий характеризуются материалом постройки и заселенностью. Виды материалов зданий встречаются следующие: кирпичное (обозначается буквой “К”), металлическое (“М”), смешанное (“СМ”). По заселенности здания могут быть жилыми (“Ж”) и не жилыми (“Н”). Информация внутри здания является комбинацией этих типов. Примеры надписей внутри зданий представлен ниже (Таблица 3.2).

Таблица 3.2 Примеры надписей и их значения

Надпись	Количество этажей	Тип материала	Заселенность
2КЖ	2	Кирпич	Жилое
КЖ	1	Кирпич	Жилое
МН	1	Металл	Нежилое
К	1	Кирпич	-
10СМЖ	10	Смешанное	Жилое

Отсутствие подобной внутренней надписи внутри объекта означает, что это не здание.

3.2. Векторизация контура здания

Каждый контур, полученный с помощью волнового алгоритма, состоит из множества точек. Для того, чтобы построить многоугольник, на который можно заменить этот контур, необходимо произвести так называемую векторизацию данного множества точек контура. Векторизация контура является процессом, в результате которого из множества точек контура остаются только те, которые имеют наибольшую значимость для определения формы контура, т.е. узловые точки, соединённые друг с другом векторами. Необходимо избавиться от промежуточных точек, без которых форма контура не претерпит серьёзных изменений. Для этого пропустим контур через ряд фильтров, которые отсеют необязательные точки.

Сокращение последовательности коллинеарных векторов. Точки в контуре располагаются рядом с друг к другу. Это не позволяет верно оценить количество стен здания и найти углы между ними. В виду того, что топоплан имеет сеточное дискретное пространство, возможных направлений между двумя соседними точками может быть 8 (Рисунок 3.2).

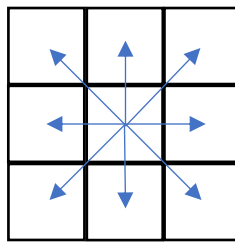


Рисунок 3.2 Направления к соседним ячейкам

В первую очередь из контура должны быть удалены точки, которые являются общими для подряд идущих коллинеарных векторов. Для примера рассмотрим последовательность точек, условно образующих замкнутый контур здания (Рисунок 3.3).

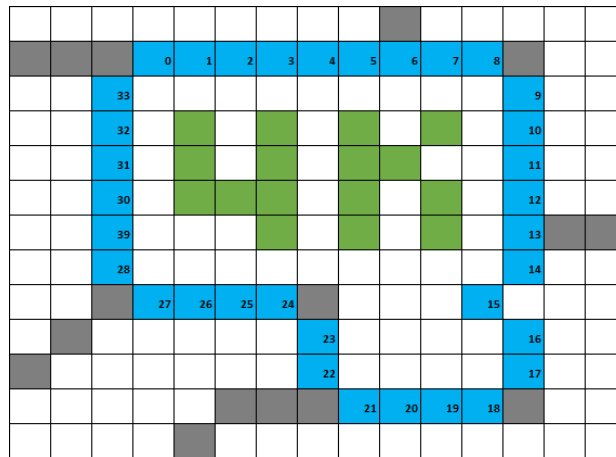


Рисунок 3.3 Упорядоченные точки контура

Пусть указанный контур является массивом точек P , который содержит точки, лежащие на одной прямой, поэтому, с целью минимизировать их число, такие точки могут быть удалены из массива.

Пусть любая точка массива P_k имеет координаты x_k и y_k . Если выполняются соотношения (3.1), то точка P_k лежит между P_{k-1} и P_{k+1} на одной прямой, а значит она может быть удалена.

$$\begin{aligned} x_k - x_{k-1} &= x_{k+1} - x_k \\ y_k - y_{k-1} &= y_{k+1} - y_k \end{aligned} \quad (3.1)$$

Где x_{k-1} , y_{k-1} , x_{k+1} , y_{k+1} – соответственные координаты точек P_{k-1} и P_{k+1} . Таким образом, после обработки всех точек массива, количество данных уменьшится, причем без потери геометрической формы контура (Рисунок 3.4).

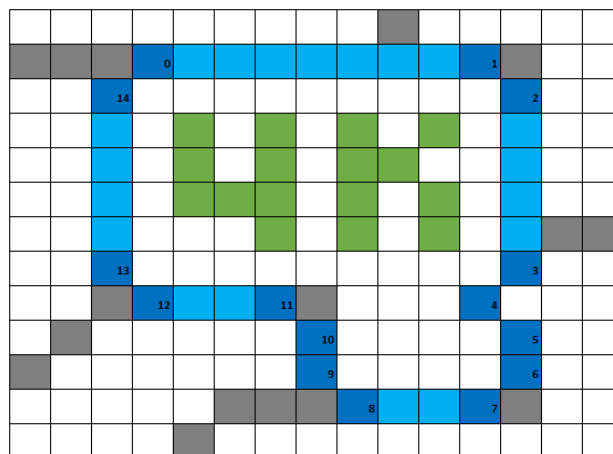


Рисунок 3.4 Массив P после удаления повторов

После удаления точек, лежащих на одной прямой, образовались небольшие группы связанных точек. Такие группы точек могут быть объединены в одну.

Возьмём любые две соседние точки P_k и P_{k+1} . Если между их координатами одновременно верны соотношения (3.2), то эти точки могут быть объединены в одну.

$$\begin{aligned} |x_{k+1} - x_k| &= 1 \\ |y_{k+1} - y_k| &= 1 \end{aligned} \quad (3.2)$$

где $x_k, y_k, x_{k+1}, y_{k+1}$ – соответственные координаты точек P_k и P_{k+1} .

Соотношения выше говорят о том, что точки должны располагаться по диагонали. Отсюда следует, что есть два варианта того, как их объединить: удалить одну из точек или удалить обе и создать новую между ними. При выборе первого варианта геометрия контура резко изменит свою форму. В то время, как при выборе второго варианта будет создаваться точка не внутри контура, тем самым, сохранив его форму максимально точно (Рисунок 3.5).

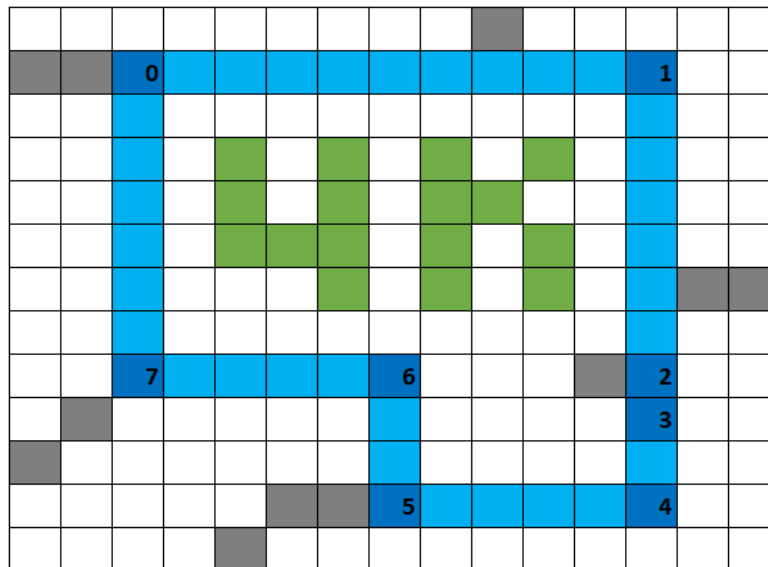


Рисунок 3.5 Массив P после объединения точек

Данный алгоритм рекуррентно применяется к контуру до тех пор, пока на каждом шаге будет удаляться хотя бы одна точка.

Выпрямление сторон контура. Количество точек может быть минимизировано в массиве P благодаря удалению тех точек, которые близко расположены к отрезку, соединяющему две соседние с ней точки.

Рассмотрим точку P_k и две соседние с ней P_{k-1} и P_{k+1} (Рисунок 3.6).

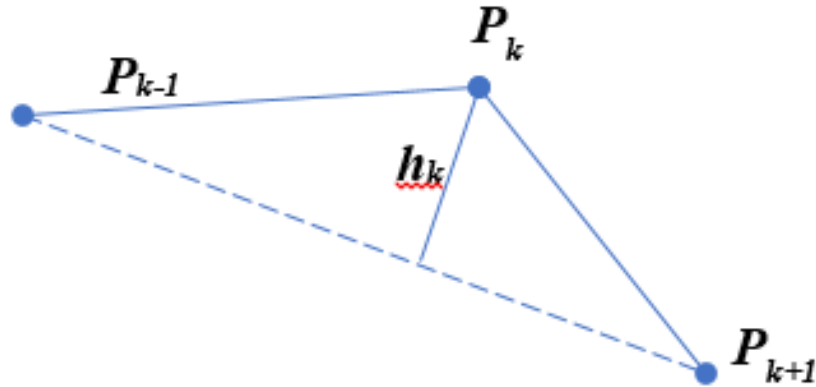


Рисунок 3.6 Высота, опущенная из вершины P_k на отрезок

Найдём значение высоты через площадь треугольника (3.3), которая будет равна половине площади параллелограмма, образованного на векторах $\overrightarrow{P_k P_{k-1}}$ и $\overrightarrow{P_k P_{k+1}}$.

$$S_{mp} = \frac{1}{2} \begin{vmatrix} x_{k-1} - x_k & y_{k-1} - y_k \\ x_{k+1} - x_k & y_{k+1} - y_k \end{vmatrix}, \quad (3.3)$$

где S_{mp} – площадь треугольника (половина площади прямоугольника, построенного на векторах).

Также, площадь треугольника может быть определена через высоту и сторону (3.4), на которую она опущена.

$$S_{mp} = \frac{1}{2} h_k |P_{k-1} P_{k+1}|, \quad (3.4)$$

Где h_k – длина высоты, опущенной на прямую $P_{k-1}P_{k+1}$ с вершины P_k .

В силу того, что выражения (3.3) и (3.4) тождественно равны, то верно следующее равенство (3.5):

$$\frac{1}{2} \begin{vmatrix} x_{k-1} - x_k & y_{k-1} - y_k \\ x_{k+1} - x_k & y_{k+1} - y_k \end{vmatrix} = \frac{1}{2} h_k |P_{k-1} P_{k+1}| \quad (3.5)$$

Отсюда следует, что высота будет определяться согласно (3.6):

$$h_k = \frac{\begin{vmatrix} x_{k-1} - x_k & y_{k-1} - y_k \\ x_{k+1} - x_k & y_{k+1} - y_k \end{vmatrix}}{|P_{k-1} P_{k+1}|} \quad (3.6)$$

При таком подходе нахождения высоты число умножений и суммирований сведено к минимуму. Это позволяет уменьшить время работы алгоритма.

Для каждой точки P_k ищется высота, опущенная из неё на отрезок, соединяющий соседние с ней точки. В том случае, если эта высота окажется меньше некоторого установленного значения h_{max} , то можно считать, что эта точка практически лежит на одной прямой с двумя соседними и её можно удалить.

Если хоть одна точка на этом этапе была удалена, то шаг поиска высот повторяется снова. Таким образом, получается конечный массив точек, по которым можно построить фигуру, близкую по геометрии к реальному контуру.

Примеры работы алгоритма. Предложенный алгоритм был применён к топоплану на Рисунок 3.7. Размер топоплана 1070x914 пикселей.

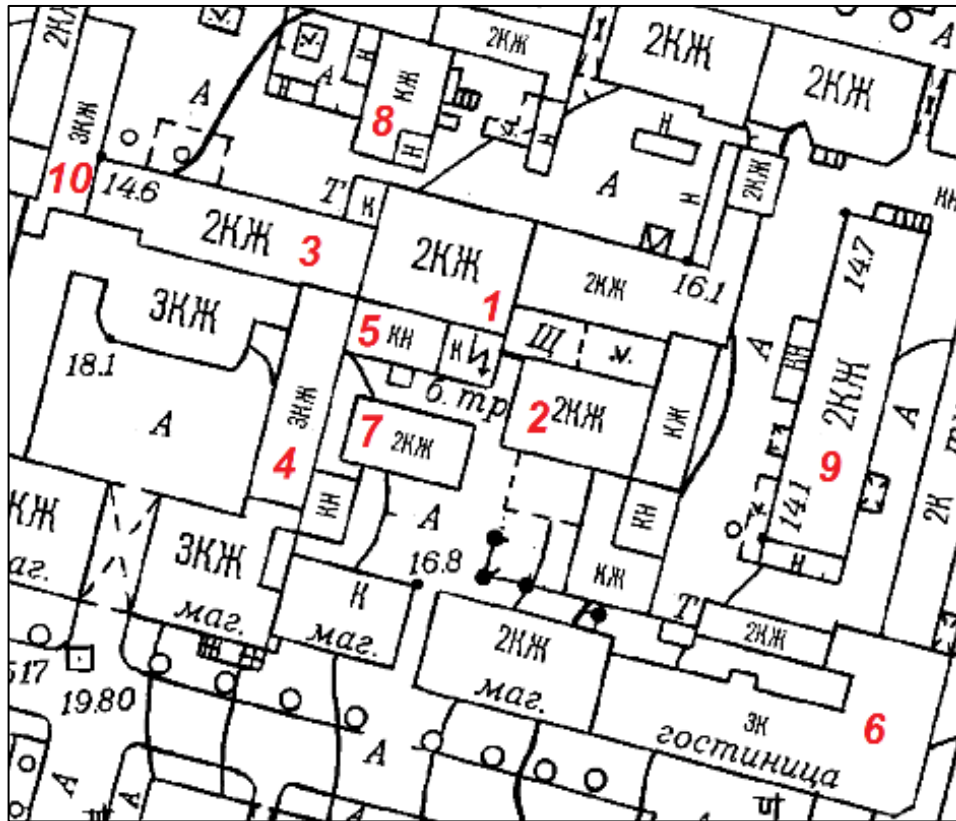


Рисунок 3.7 Топоплан. Красными числами помечены здания, для которых применялся алгоритм

Количественные характеристики некоторых зданий указаны ниже (Таблица 3.3).

Таблица 3.3. Результат распознавания зданий на топоплане

Номер здания	Число точек на текущем этапе			Реальное число точек
	Первоначальное выделение	Удаление повторов и объединение	Удаление точек с отрезка	
1	572	111	4	4
2	480	99	4	4
3	764	146	8	6
4	586	122	4	4
5	308	63	5	4
6	1142	235	16	13
7	394	80	5	4
8	414	81	6	6
9	892	176	4	4
10	594	124	8	8

Ниже (Рисунок 3.8, Рисунок 3.9, Рисунок 3.10) приведены иллюстрации шагов по выделению некоторых контуров. Показаны точки при первоначальном выделении, после удаления повторных точек и после удаления повторных точек и точек, искажающих прямолинейность стороны контура. Показан результат выпрямления углов (Рисунок 3.10), алгоритм которого будет описан ниже.

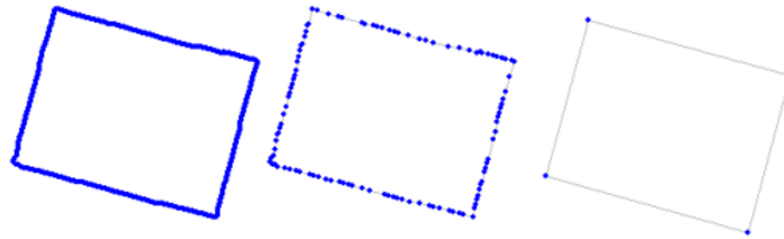


Рисунок 3.8 Выделение контура для здания 1

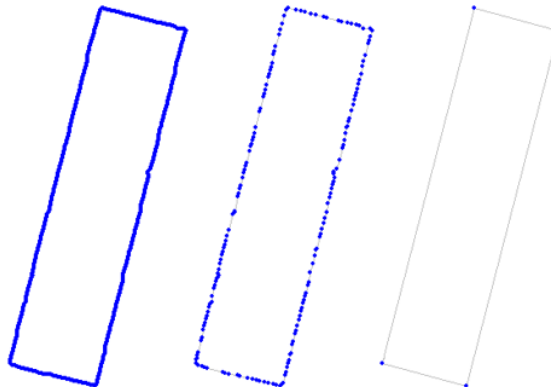


Рисунок 3.9 Выделение контура для здания 9

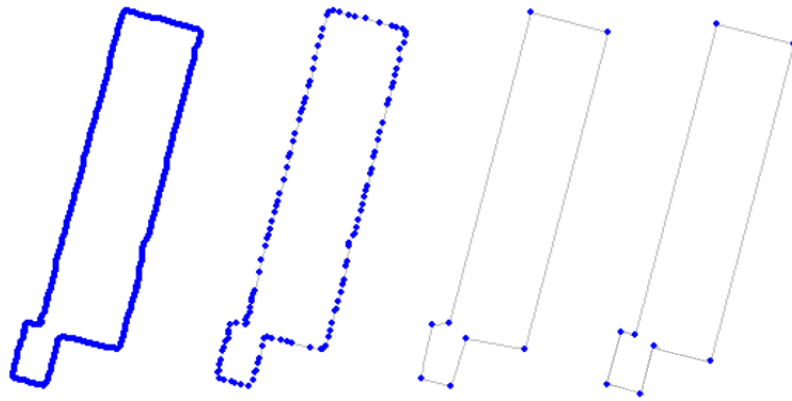


Рисунок 3.10 Выделение контура для здания 10

Границы контуров зданий на топоплане могут содержать различные дефекты, которые приводят к тому, что стены распознанных зданий не располагаются под прямым углом, как это должно быть в реальности. При условии, что у всех зданий стены ортогональны, то углы между сторонами контура приводятся к прямым углам.

Для этого определяется основная сторона контура – это наибольшая по длине сторона. Рассматриваются соседние с ней стены. Из них выбирается наибольшая. Она становится перпендикулярной к основной стороне. Далее эта стена будет считаться основной. Относительно новой основной стены рассматриваются другие стены, которые ранее не рассматривались. Процесс повторяется до тех пор, пока не останется одна стена, которая еще не была приведена к прямому углу с соседними (Рисунок 3.11). Она делается перпендикулярной обоим.

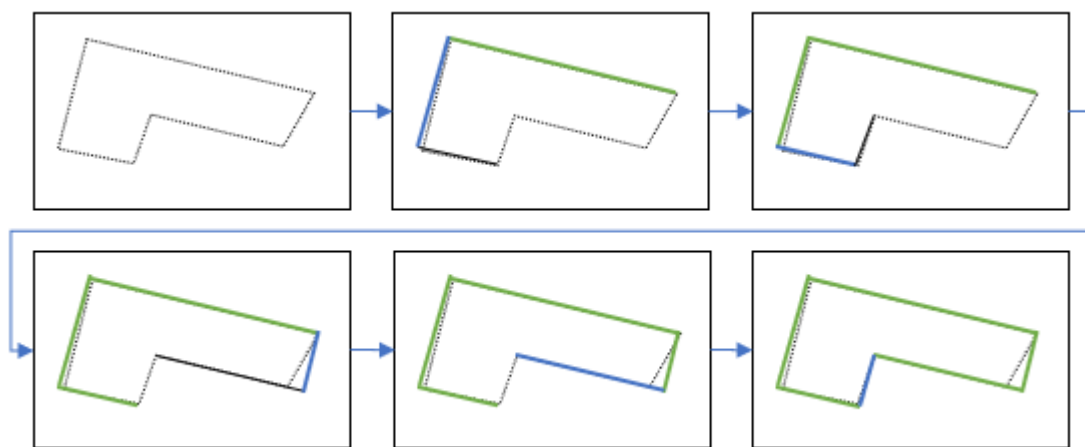


Рисунок 3.11 Схема спрямления углов в контуре

3.3. Распознавание внутренней информации здания

Каждый контур здания обязательно внутри себя содержит служебную информацию о количестве этажей в здании, а также дополнительные данные о его структуре (жилое, нежилое, кирпичное и т.д.). Информация о количестве этажей является необходимой для трёхмерной реконструкции. При наличии данных об этажности здания, а также масштабе топоплана, может быть установлена высота городского объекта (Рисунок 3.12).

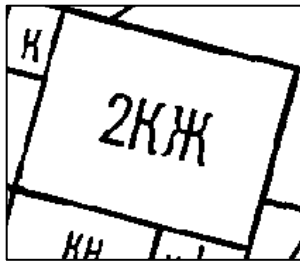


Рисунок 3.12 Пример текста внутри контура

Другие данные тоже полезны, потому что они характеризуют внешний вид здания (они будут учитываться при текстурировании).

Существует целый ряд методик по распознаванию текстовой информации на цифровом изображении. Он включает распознавание с использованием нейросетей, скелетизации текста и т.д.

Угол наклона надписи вычисляем. На топоплане текст внутри зданий имеет ту же ориентацию, что и одна из его стен. Подробно о его нахождении будет представлено в п. «Определение угла наклона надписи». Размеры символов могут варьироваться. Исходя из этих положений в качестве способа распознавания текста выбран простой метод корреляции пикселей. Далее будет описан алгоритм распознавания такого текста.

Эталонные символы. Топоплан содержит ограниченное количество возможных символов (Рисунок 3.13). Все возможные текстовые символы внутренней информации здания представлены на рисунке ниже:

0123456789ЖКМНС/

Рисунок 3.13 Служебные символы, встречающиеся на топопланах

Каждый эталонный символ представляет из себя матрицу (Рисунок 3.14), заполненную нулями и единицами (ноль соответствует белому цвету на изображении, а единица – чёрному).

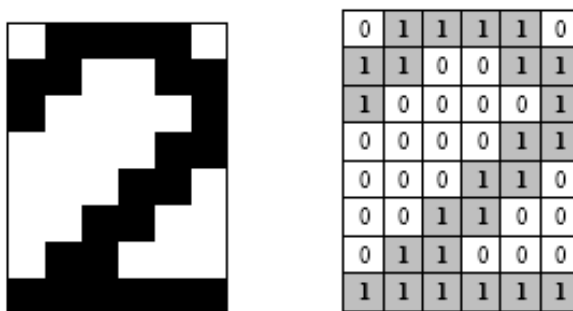


Рисунок 3.14 Эталон в виде изображения (слева) и матрицы (справа)

Задачу распознавания осложняет тот факт, что текст обычно располагается под углом. Однако, тот факт, что этот угол всегда совпадает с углом наклона одной из стен здания позволяет иметь небольшое количество возможных вариантов поворота. Для здания с N количеством сторон нужно N раз повернуть внутреннюю надпись на угол, который совпадает с углом, образованным между прямой, на которой лежит сторона и горизонтальным основанием топоплана. Каждый символ надписи должен будет сравнен с эталонными символами. Однако, ниже (п. «Определение угла наклона надписи») будет предложен подход по определению необходимого угла наклона.

Далее распознанная надпись будет подвергнута синтаксическому анализу для проверки на соответствие с правилами нанесения обозначений на топографический план. В результате будет получено N надписей, из которых будет выбрана та, которая наиболее соответствует эталонным символам и результатам синтаксического анализа.

3.4. Выделение символов

При определении границы контура была необходимость в разделении точек границы контура от точек надписи. Пусть M – матрица, полученная после работы волнового алгоритма внутри контура. Выделим в матрице M подматрицу U и поместим в неё все элементы, равные -1 (Рисунок 3.15, Рисунок 3.16).

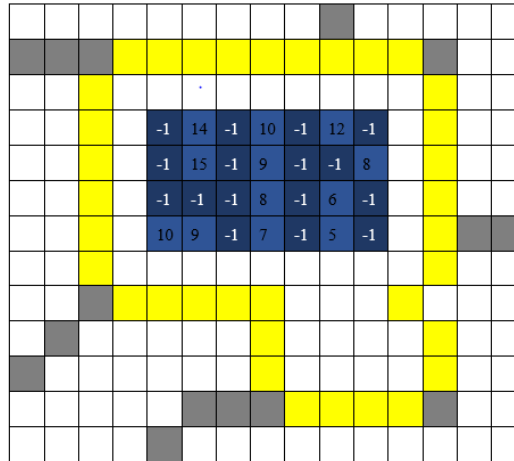


Рисунок 3.15 Матрица M , внутри которой находится матрица U

-1	14	-1	10	-1	12	-1
-1	15	-1	9	-1	-1	8
-1	-1	-1	8	-1	6	-1
10	9	-1	7	-1	5	-1

Рисунок 3.16 Матрица U отдельно от матрицы M

Матрица U содержит группы соседствующих элементов. Каждая такая группа является отдельным символом. Необходимо выделить в этой матрице все символы по отдельности.

Выберем любой элемент в первом столбце матрицы U . Далее из него будет запущен волновой алгоритм со следующими особенностями:

- Волна распространяется по восьми направлениям;
- «Свободными» ячейками будут те, значение которых равно -1;
- «Стены» - любые ячейки, значения которых не равно -1;

- Начальной точке присваивается значение -2 , а каждой следующей на единицу меньше.

В результате работы такого алгоритма образуется набор элементов, значение которых меньше -1 . Совокупность этих элементов является символом. Выделим в матрице U подматрицу J_1 , в которую полностью входят элементы меньше -1 матрицы U (Рисунок 3.17).

-2	14	-6	10	-1	12	-1
-3	15	-5	9	-1	-1	8
-4	-4	-5	8	-1	6	-1
10	9	-5	7	-1	5	-1

-2	14	-6
-3	15	-5
-4	-4	-5

Рисунок 3.17 Выделенный символ в матрице U (слева) и матрица J_1 (справа)

Изменим значения в матрице J_1 следующим образом:

- Элементам, у которых значение больше -2 , задаётся значение 0 ;
- Если значение элемента матрицы меньше -1 , то оно меняется на 1 .
- После проведённых действий матрица J_1 примет следующий вид (Рисунок 3.18):

1	0	1
1	0	1
1	1	1
0	0	1

Рисунок 3.18 Матрица J_1 после изменений

Те элементы матрицы U , у которых значение меньше -2 меняют своё значение на 1 (Рисунок 3.17). Таким образом, в матрице U стало на один символ меньше. Далее, в следующих столбцах ищется элемент, значение которого равно -1 и процесс повторяется снова.

После всех действий будет образован набор из C матриц. Пусть этим множеством будет массив J , элементы которого J_k , где $k=1..C$.

На Рисунок 3.19 приведены примеры служебных символов внутри контуров. С помощью прямоугольников выделены найденные символы.



Рисунок 3.19 Примеры выделения символов на топопланах

Следует также отметить, что внутреннее содержание контура может содержать небольшие группы точек, которые являются помехами и не несут никакой полезной информации. Поэтому, зададим некоторое значение J_{min} – минимальное число элементов в любой из матриц J равных 1. Если число таких элементов матрице J_k меньше установленного значения J_{min} , то этот символ удаляется из массива J .

Определение угла наклона надписи. Угол наклона надписи определяется её положением относительно горизонтальной линии изображения.

Надпись может содержать один или несколько символов. Если надпись содержит несколько символов, то направление надписи будет определяться по прямой, которая проходит между двумя наиболее удаленными друг от друга центрами описанных вокруг символов прямоугольниками (Рисунок 3.20).

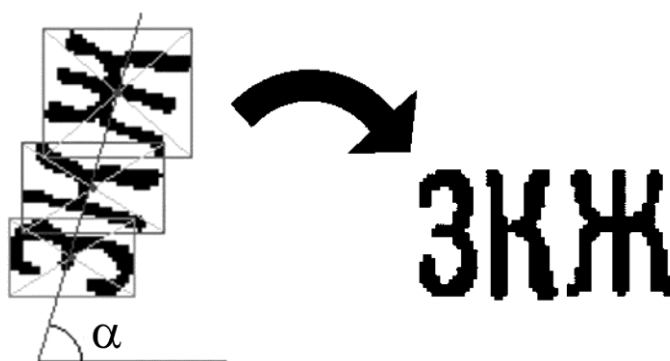


Рисунок 3.20 Угол между направлением надписи и горизонтальной стороной изображения

Если надпись состоит из одного символа, то им может быть или буква «К», или буква «Н». Поэтому распознавание символа будет происходить только для двух эталонов.

Для каждого из эталонных символов предполагается, что:

- Высота символа больше, чем его ширина;
- Описанный вокруг символа прямоугольник, имеющий наименьшую площадь, практически точно совпадает с описанным вокруг символа прямоугольником, стороны которого параллельны сторонам изображения.

Данные предположения позволят точно определить угол наклона каждого символа. Направление надписи будет совпадать с отрезком меньшей длины соединяющего середины противоположных сторон описанного вокруг символа прямоугольника (Рисунок 3.21).

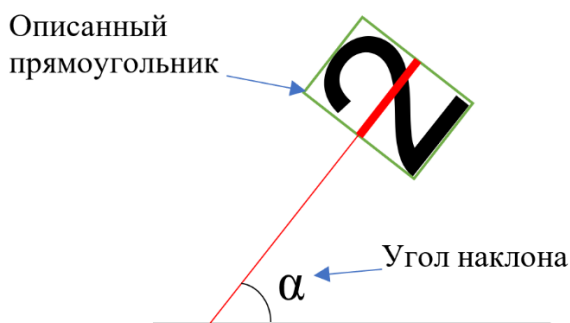


Рисунок 3.21 Угол поворота по меньшей средней

Для того, чтобы получить эту минимальную среднюю линию, необходимо описать вокруг символа прямоугольник меньшей площади. Его размеры и угол относительно осей дают представление о том, как расположен символ.

Т.к. прямоугольник может находиться под некоторым углом к горизонтальной стороне топоплана, то в рамках дискретного пространства нельзя добиться точной формы фигуры с прямыми углами и параллельными линиями. Поэтому описанным прямоугольником будет являться фигура, которая наиболее точно ему соответствует. Это четырехугольник, у которого углы практически прямые, а противоположные стороны практически параллельны. Основной характеристикой этого четырехугольника будет набор из четырех точек. Таким образом задача сводится к нахождению точек пересечения двух пар практически параллельных прямых, которые образуют область наименьшей площади.

Необходимо найти вокруг точек описанную минимальную выпуклую оболочку (МВО). Для этого воспользуемся алгоритмом Джарвиса. Он состоит из трех шагов:

Шаг 1. Происходит поиск самой левой нижней точки, которая гарантированно является вершиной МВО. Эта точка будет являться стартовой.

Шаг 2. Берется точка, которая имеет наименьший положительный угол относительно стартовой точки. Она становится вершиной МВО.

Шаг 3. Среди точек, которые еще не являются вершиной МВО, но включая первую, определяется та, которая образует минимальный косинус с прямой, проходящей через две предыдущие точки МВО. Эта точка становится вершиной МВО.

Таким образом будет сформирован набор точек, который будет образовывать МВО. Т.к. это выпуклая оболочка, то описывающий четырехугольник с минимальной площадью будет в качестве одной из своих сторон содержать какой-нибудь из отрезков МВО.

Перебираются все стороны МВО. Для этого каждый раз берется пара соседних вершин МВО. Эта пара точек образует прямую. Через все остальные вершины МВО проводятся перпендикуляры к образованной прямой. Из всех этих перпендикуляров выбираются два, расстояние между которыми наибольшее. Из всех точек МВО выбирается та, которая больше всего удалена от исходной прямой. Через эту точку проводится прямая параллельная исходной. Таким образом получено четыре прямые, которые образуют практически прямоугольную область. Последовательно идущие прямые пересекаются и дают необходимые четыре точки четырехугольника. По этим четырём точкам рассчитывается его площадь. Четырёхугольник делится на два треугольника и по координатам его вершин рассчитываются длины сторон. На основании их с помощью формулы Герона рассчитывается площадь.

Таким образом, перебираются все отрезки МВО и выбирается та четверка точек, которая образует наименьшую площадь (Рисунок 3.22).

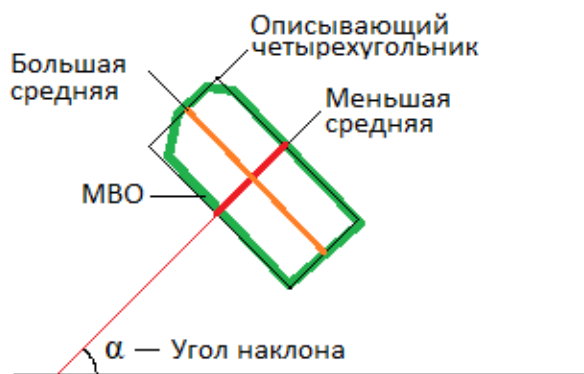


Рисунок 3.22 Характеристики описывающего четырехугольника

Распознавание текста. После определения угла между надписью и горизонтальной стороной изображения может быть выполнен поворот каждого символа на найденный угол. Однако, поворот должен осуществляться таким образом, чтобы в дальнейшем была возможность сравнения символа с эталоном. Для этого необходимо преобразование каждой точки эталона к точке символа. Это будет осуществляться с помощью матричных преобразований (масштабирование, перемещение и поворот).

Матрица масштабирования позволяет выполнить такое преобразование, которое линейно растягивает координаты точки вдоль осей OX и OY на коэффициенты S_x и S_y соответственно (3.7).

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (3.7)$$

Матрица поворота позволяет выполнить такое преобразование, которое поворачивает точку относительно центра координат на указанный угол α (3.8).

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos \alpha & \sin \alpha & 0 \\ -\sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (3.8)$$

Матрица перемещения позволяет выполнить такое преобразование, которое смещает точку относительно своих координат на указанный вектор T с координатами T_x и T_y (3.9).

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & T_x \\ 0 & 1 & T_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (3.9)$$

Поместим некоторый эталонный символ в начало координат таким образом, чтобы его левый нижний угол совпадал с началом координат. Тогда для того, чтобы сопоставить любую точку эталона с координатами x и y , с точкой распознаваемого символа, необходимо выполнить следующее преобразование (3.10):

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos \alpha & \sin \alpha & 0 \\ -\sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & T_x \\ 0 & 1 & T_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (3.10)$$

где S_x, S_y – коэффициенты масштабирования, α – угол поворота символа, а T_x и T_y , вектор на который смещен символ (3.11).

$$S_x = \frac{R_{\min}}{E_w} \quad (3.11)$$

$$S_y = \frac{R_{\max}}{E_h}$$

где R_{\min}, R_{\max} – соответственно длина минимальной и максимальной средней линии описанного около символа прямоугольника минимальной площади; E_w, E_h – соответственно ширина и высота эталонного символа (Рисунок 3.23).

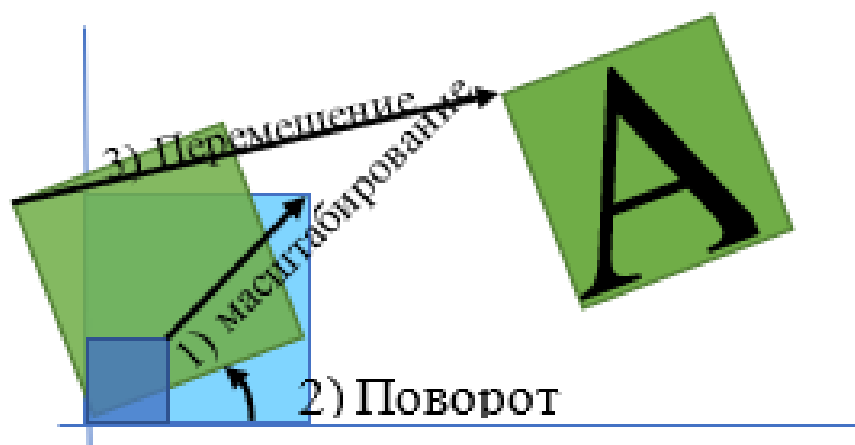


Рисунок 3.23 Преобразование перед сопоставлением изображений

Каждая точка эталона проходит через преобразование 3.11. Все точки можно представить в виде некоторой матрицы A , а все точки символа на изображении

представляются в виде матрицы *B*. Между двумя матрицами находится коэффициент корреляции (1.3).

Для каждого исходного набора точек строится последовательность эталонов, сортированная по коэффициенту корреляции. Например, для символа «Ж» (Рисунок 3.12) таблица процентов корреляции выглядит следующим образом (Таблица 3.4):

Таблица 3.4. Распределение процента корреляции для символа "Ж"

№ эталона	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
Эталон	/	1	0	4	2	7	3	С	6	8	5	9	Н	М	К	Ж
Процент корреляции	7	8	15	16	17	19	22	42	44	45	47	54	60	61	64	88

Анализ синтаксиса. Служебная информация внутри зданий на рассмотренных выше топопланах имеет строгую структуру. В начале текста находятся цифры, указывающие число этажей. Если здание имеет переменную этажность, то после цифр ставится символ «/», а затем пишется другое число этажей. Далее записывается буквенная информация о типе здания, т.е. является он жилым, нежилым, кирпичным и т.д. Возможны несколько комбинаций: «КЖ», «КН», «Н», «К», «СМЖ».

Целью распознавания текста является определение числа этажей в здании, а также его тип. Будем анализировать найденную надпись по каждому символу, используя следующие критерии:

1. Если число этажей здания равно 1, то в таких случаях единица не наносится на топоплан и далее пишутся буквы. Следовательно, если первый символ является буквой, то число этажей равно 1.
2. Если до появления букв не встречается символ «/», то число этажей равно тому числу, что записано перед буквами.
3. Если до появления букв символ «/» встречается, то число этажей равно числу, записанному до этого символа.
4. Если в строке содержится один символ, то им может быть либо «К», либо «Н». В том случае, если распознанный символ не является ни одним из

перечисленных, то в зависимости от процентного сходства тип здания устанавливается как «К» или «Н»

5. Если последний символ распознан как «Ж», то может быть два варианта: «КЖ» или «СМЖ». Тогда тип здания выбирается исходя из числа символов в строке без учёта тех, которые определяют число этажей.
6. Если символ «/» распознан как «1» и с двух его сторон находятся числа, то тогда число этажей не вписывается в реальные параметры здания. В таком случае, в исходной строке «1» заменяется на «/» и происходит повторный анализ.
7. В том случае, если распознанная надпись не подходит ни под один из описанных критериев, то тип здания устанавливается как «КЖ».

Пример обработки служебной информации. Предложенный алгоритм был применён к нескольким контурам на топоплане (Рисунок 3.24).



Рисунок 3.24 Топоплан с отмеченными зданиями

Анализ синтаксиса распознанного текста ниже (Таблица 3.5).

Таблица 3.5. Результат распознавания синтаксиса

Номер здания	Реальная надпись	Распознано	№ критерия	Число этажей	Тип здания
1	1/2КЖ	1/2КЖ	3, 5	1	КЖ
2	3КЖ	3КЖ	2, 5	3	КЖ
3	1/2КЖ	1/2КЖ	3, 5	1	КЖ
4	2КЖ	2НЖ	2, 5	2	КЖ
5	КЖ	НЖ	1, 5	1	КЖ
6	3КЖ	3КЖ	2, 5	3	КЖ
7	2КЖ	2КЖ	2, 5	2	КЖ
8	КН	КН	1	2	КН
9	4КЖ	4КЖ	2, 5	4	КЖ
10	3/4КЖ	314КЖ	3, 5, 6	3	КЖ
11	3СМЖ	3СМЖ	2, 5	3	СМЖ
12	Н	Н	1, 4	1	Н

Как видно из представленных данных, в подавляющем большинстве случаев алгоритм распознал информацию верно. У алгоритма возникли сложности с верным распознаванием символов «К» и «Н» из-за их схожести. Также, может встречаться неверное распознавание для символов «1» и «/» по той же причине. Дополнительный пиксельный анализ в данном случае не сможет дать верный результат. Следовательно, необходимо подойти к проблеме с точки зрения синтаксиса текста.

3.5. Применение алгоритмов

Рассмотренные методы по выделению контуров и распознаванию текста могут быть теперь объединены для нахождения всей необходимой информации о здании на топоплане.

Применим описанные алгоритмы к топоплану на Рисунок 3.25. Размеры исходного изображения 1564x1240 пикселей. На топоплане находится 45 зданий различных типов. Количество служебных символов равно 16. Результат на Рисунок 3.26 и Рисунок 3.27:

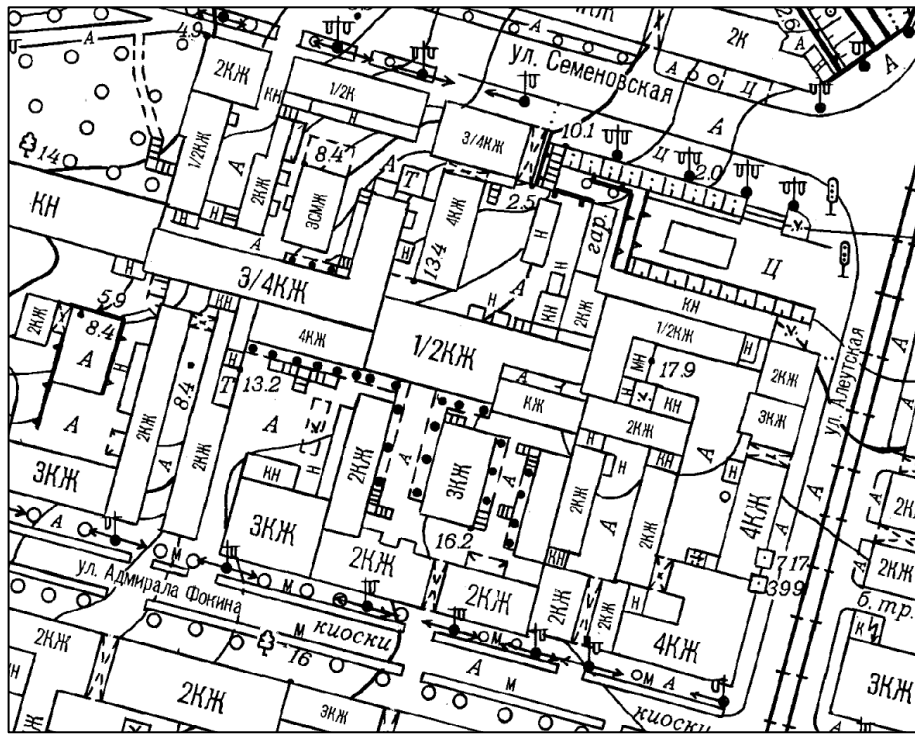


Рисунок 3.25 Топографический план "ул. Адмирала Фокина"



Рисунок 3.26 Выделенные здания и текст

Из-за того, что на топоплане присутствует достаточно много вспомогательных обозначений, таких как контуры изолиний и коммуникаций, объекты топоплана пересекаются с ними. Это делает невозможным применение метода выделения контуров этих объектов по аналогии с поиском символов внутренней информации.

На Рисунок 3.28 уличный фонарь и клумба соприкасаются с контуром дороги. При этом первый еще пересекается со зданием и изолинией. Каждый из объектов обладает внутренней областью. Причем, эта область никогда не пересекается с другими объектами. Это свойство исходит из того, что объекты топоплана нанесены на него поверх других второстепенных обозначений.

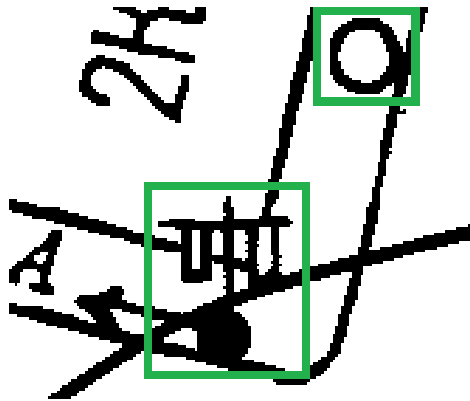


Рисунок 3.28 Пересечение объектов с второстепенными обозначениями

Т.к. внутренняя область не пересекается с другими объектами, то она обладает определенным контуром. На этапе выделения контуров зданий был получен массив всех контуров топоплана, куда были включены также и внутренние контуры дополнительных объектов. У контуров объектов одного типа есть ряд свойств:

- Меньшая и большая средние линии контура примерно совпадают по направлению и имеют практически одинаковые соответствующие длины. Отсюда следует, что отношение большей средней к меньшей средней для объектов одного типа есть некоторая константа.
- Площадь и периметр контуров примерно совпадают. Отсюда следует, что отношение площади к квадрату периметра есть некоторая константа.

Из свойств выше выведено, что контуры объектов одного типа обладают уникальными константами. Каждая такая константа может быть рассчитана на основании эталонных изображений этих обозначений. Однако, константы не дают точной классификации объекта. Для более надежного определения будет использоваться поточечный анализ части топоплана и эталонного обозначения объекта, к константам которого его показатели наиболее близки. Для каждого типа объекта будет выведен определенный порог коэффициента корреляции изображения этого объекта с частью топоплана. Сравнение изображений будет происходить аналогично тому, как это выполнялось при определении символов внутренней информации, за исключением того, что преобразование поворота не будет применяться.

Выделение дополнительных объектов на топоплане необходимо для определения их положения на топоплане. Обозначение каждого такого объекта привязано к определенной точке. Эта точка будет называться опорной точкой дополнительного объекта. Т.к. клумба имеет геометрическую форму окружности, то пусть её опорная точка будет находиться в центре этой окружности. Остальные объекты по своей структуре вытянуты по высоте, поэтому пусть их опорные точки будут находиться по центру снизу изображения.

Фонарный столб. Внешний контур фонарного столба на топоплане может накладываться на какой-нибудь другой объект. В связи с этим анализ внешнего контура будет проблематичен. Фонарный столб имеет внутренние контуры, которые не содержат никаких других. Определив эти контуры, можно будет получить представление о положении фонарного столба на топоплане.

Пусть имеется часть некоторого топоплана (Рисунок 3.29), на котором обозначение фонаря накладывается на некоторую линию.

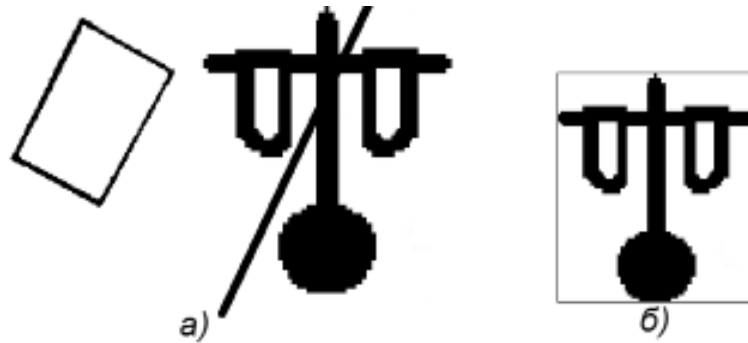


Рисунок 3.29 Часть топоплана (а) и эталонное изображение фонаря (б)

Фонари на топоплане могут обозначаться двумя способами: двойные и одинарные. Для двойного фонаря должны быть два контура. Оба они должны быть подобны контурам эталонного фонаря.

Пусть имеется некоторый исходный контур. Парным контуром для него будет считаться тот, если будут выполняться следующие условия:

- Оба контура имеют примерно одинаковую площадь и периметр
- Отношение высоты и ширины описанных вокруг каждого контура прямоугольников пропорциональны отношению высоты и ширины прямоугольников вокруг контуров на эталоне.
- Центры описанных прямоугольников вокруг этих контуров находятся примерно на одной горизонтали и на расстоянии друг от друга прямо пропорциональном соответствующему расстоянию на эталонном фонаре. При этом коэффициент пропорциональности выбирается, как отношение высот описанных прямоугольников вокруг исходного контура и одного из контуров эталона.

Если для исходного контура будет найден парный, то область топоплана вокруг них будет проверена на соответствие эталонному изображению фонаря. Область выбирается прямоугольной. Размеры ее пропорциональны расстояниям между контурами на топоплане и эталоне. При этом коэффициент пропорциональности выбирается как отношение высот, описанных вокруг контуров прямоугольников на топоплане и эталоне. Для фонаря порог корреляции составляет 0.8.

Светофор. Внутренний контур светофора должен содержать три небольших вложенных контура, центры описанных вокруг них прямоугольников, которых примерно выстроены в вертикальную линию. Площади этих контуров должны быть примерно одинаковыми.

Другой характеристикой класса является то, что отношение ширины и высоты прямоугольника, описанного вокруг внутреннего контура должно быть примерно таким же, как и на шаблоне. Данное условие аналогично условию (1). Значение погрешности также составляет 0.4.

Если условия выше выполняются, то выделяем требуемую область на топоплане и сравниваем ее с шаблоном светофора аналогично тому, как это делалось при выделении фонаря. Если порог корреляции будет больше определенного значения, то можно сделать вывод, что светофор на топоплане обнаружен. Для светофора этот порог составляет 0.8 от количества точек эталона.

Дерево. Для того чтобы задать характеристики класса контуров для дерева, воспользуемся пропорциональным отношением ширины и высоты прямоугольника, описанного вокруг внутреннего контура прямоугольника к соответствующей высоте и ширине контура на шаблоне. Внутренний контур не должен содержать вложенных.

При выполнении данных условий выделим на топоплане требуемую область аналогично тому, как это делалось при распознавании фонаря и светофора, которую сравним с шаблоном дерева. Контур будет определен как дерево, если коэффициент корреляции при сравнении будет больше 0.8.

Клумба. Клумба обозначается на топоплане обычным кругом, внутри которого ничего не должно быть. А это значит, что при некоторой погрешности отношение квадрата периметра контура к его площади будет равно всегда примерно одному и тому же числу, а именно $4/\pi \approx 1.27$. Это свойство помогает решить проблему инвариантности объекта к масштабированию.

3.7. Выводы по главе

Предложенный метод векторизации контура успешно минимизирует количество точек в контурах зданий. Это обеспечит более точное построение трехмерной модели на соответствующем этапе. Дополнительно, данный метод позволяет повысить точность определения признака того, что контур принадлежит зданию. Метод векторизации может быть применен за рамками данной работы в тех случаях, когда требуется минимизировать количество точек контура для некоторого многоугольника.

Предложен метод распознавания текста на топоплане с использованием волнового алгоритма. Метод позволяет распознавать текст эффективно благодаря разделению алгоритмов на анализ однобуквенной надписи и многобуквенной. Для каждого типа надписи производится определение угла наклона. Непосредственная идентификация символов осуществляется с помощью метода корреляции изображений. После определения побуквенной надписи осуществляется синтаксический анализ надписи в целом. После её корректировки определяется число этажей и тип строения. Совокупность этих решений дает надежные результаты при выделении текста внутри зданий.

Предложен метод выделения условных обозначений на топоплане с учетом зашумленности. Метод использует контурные особенности эталонных обозначений и корреляционное сравнение с эталоном. Использование контурного анализа позволяет отсеять случаи перед сравнением изображений, когда заранее известно, что рассматриваемый контур не является каким-либо объектом. Это позволяет применять метод эффективно, что особенно полезно при работе с топопланами больших размеров.

Глава 4. Трехмерная модель и текстурирование

Трехмерная сцена будет построена на основании найденных контуров зданий и дополнительных объектах топоплана. В трехмерную сцену также будет добавлена трехмерная модель рельефа, полученной на основании базы данных высот в Web-сервисе.

4.1. Построение трехмерной модели здания

На основе геометрической формы здания и служебной информации можно построить его упрощенную модель (как говорилось ранее, эта модель будет иметь вид прямоугольной призмы). Зная количество этажей здания $k_{эт}$ и задав некоторое значение высоте одного этажа $h_{эт}$ получим трехмерный каркас строения (Рисунок 4.1).

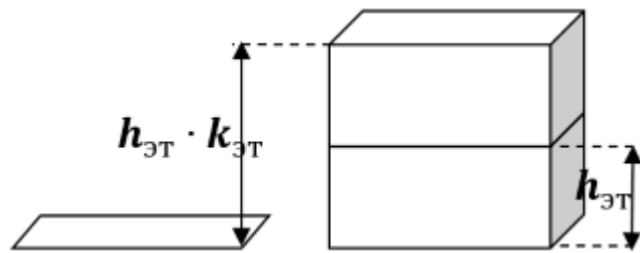


Рисунок 4.1 Образование трехмерного каркаса на основе формы здания

Высота одного этажа – условная вещь. Она не может быть явно определена для каждого здания, поэтому она является установленной величиной, которая может изменяться в зависимости от масштаба топоплана. Стандартная высота потолков в большинстве случаев варьируется в диапазоне 2,5-3,2 метров. Если для топоплана задан масштаб, то линейные пиксельные расстояния могут быть приведены к метрическим.

Для большей визуальной реалистичности генерируется крыша для каждого здания. На основании информации на топоплане данные о крыше получить невозможно, поэтому крыша будет выбираться для каждого здания случайным

образом из определенного набора заготовок. Данный набор заготовленных крыш может иметь один из следующих видов, изображённых на Рисунок 4.2.

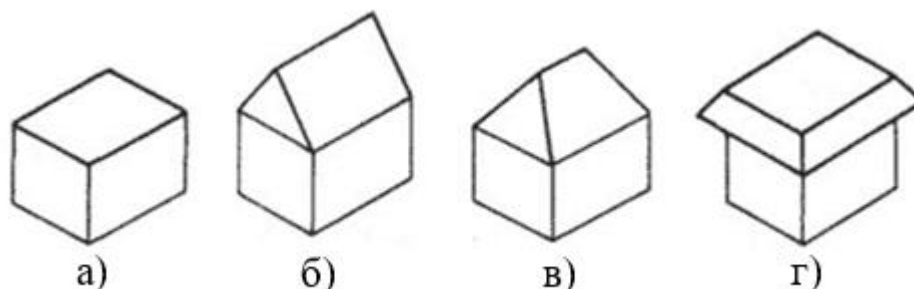


Рисунок 4.2 Заготовки крыш.

а) – Плоская; б) – Двускатная; в) – Вальмовая; г) – Плоская со скатами

Очевидно, что все виды заготовленных видов крыш являются частным случаем плоской со скатами. Выбирая случайным образом её параметры (выступ, высота, размеры верхнего среза) получим набор разнообразных видов крыш. Это позволит придать сцене неформализованный вид (Рисунок 4.3).

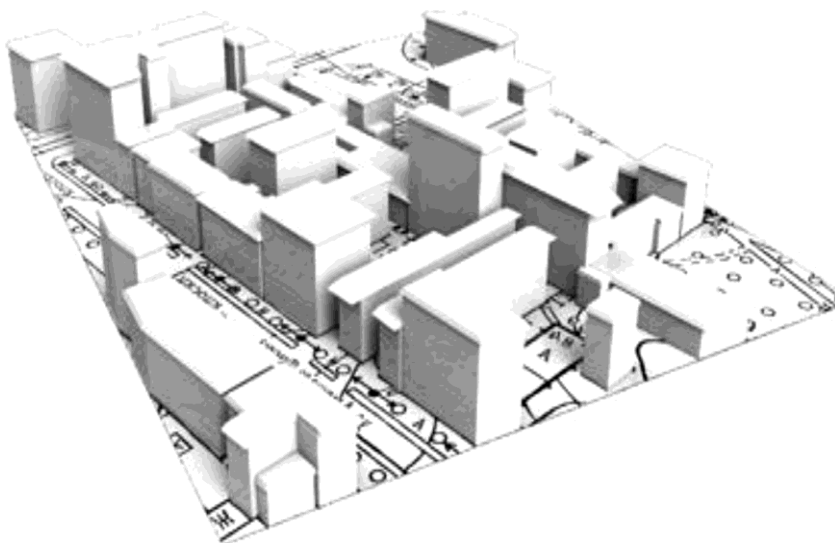


Рисунок 4.3 Пример реконструированной сцены со зданиями

4.2. Географическая привязка и рельеф

Добавление к сцене рельефа придаст ей значительную долю достоверности. Для создания рельефа сначала необходимо установить географические координаты объектов. Если этих данных нет, то можно воспользоваться сервисом *Google Maps*.

Для этого достаточно установить географические координаты двух объектов на топографическом плане. На основании этих данных возможно определить географические координаты любой точки на топоплане с помощью сервиса *Google Maps Elevation API*. Данный сервис позволяет определить высоту над уровнем моря в любой точке Земли на основании широты и долготы в определенной точке.

Сервис *Google Maps Elevation API* позволяет получать информацию о высоте над уровнем моря для массива точек в одном запросе. Количество точек в массиве ограничено только длиной запроса к сервису, который не должен превышать 2000 знаков. Чтобы увеличить количество возможных точек в массиве в сервисе предусмотрена возможность «сжимания» записи точек в более компактный вид. Запрос на получение высот для массива точек будет выглядеть следующим образом:

<https://maps.googleapis.com/maps/api/elevation/xml?locations=58.00720,56.18640|58.00439,56.18580>

После кодирования запрос будет следующим:

https://maps.googleapis.com/maps/api/elevation/xml?locations=enc:p`aJ_||uIpPvB

Что позволяет за один запрос получить информацию сразу о десятках точек.

На основании географических точек также возможно получить текстуру поверхности рельефа. Для этого используется сервис *Google Maps API*. Сервис позволяет получить изображение местности со спутника по географическим координатам точки. Например, следующий запрос:

<http://maps.googleapis.com/maps/api/staticmap?zoom=20&size=640x640¢er=58.00720,56.18640&maptype=satellite>

возвращает текстуру рельефа (Рисунок 4.4)



Рисунок 4.4 Результат запроса получения текстуры рельефа

За один запрос максимально возможное по размеру изображение можно получить с размерами 640x640 пикселей. На топографическом плане строится сетка из точек. Для каждой точки будет получаться изображение. Количество точек по горизонтали и вертикали равны согласно (4.1):

$$K_w = \left\lceil \frac{W}{640} \right\rceil$$

$$K_h = \left\lceil \frac{H}{640} \right\rceil$$
(4.1)

где **W**, **H** – соответственно ширина и высота топографического плана, **K_w**, **K_h** – соответственно минимальное количество точек по горизонтали и вертикали для построения текстуры рельефа.

Для построения всей текстуры рельефа требуется выполнить несколько запросов, количество которых равно **K_wxK_h**. Для определения ширины и высоты каждого изображения, которое будет получаться с сервиса, используется (4.2).

$$w = \left\lceil \frac{W}{K_w} \right\rceil$$

$$h = \left\lceil \frac{H}{K_h} \right\rceil$$
(4.2)

Далее из всех полученных изображений собирается готовая текстура и масштабируется до размеров топографического плана.

Например, для топографического плана с размерами 1248x496. Требуется выполнить 2 запроса. При этом каждый раз запрос получает изображение с

размерами 624x248. Результат получения текстуры рельефа продемонстрирован на Рисунок 4.5.



Рисунок 4.5 Текстура рельефа

Построение рельефа местности требует наличия данных о высотах в каждой её точке. Рельеф для сцены будет представлять из себя карту высот размера $N \times M$. Промежуточные данные будут интерполироваться.

Горизонтальные и вертикальные пиксельные координаты любой точки этой сетки вычисляются по формуле (4.3):

$$\begin{aligned} x &= \left[\frac{W}{N-1} c \right] \\ y &= \left[\frac{H}{M-1} r \right] \end{aligned} \quad (4.3)$$

где W и H – соответственно ширина и высота топографического плана, c – номер колонки точки в сетке $[0..N-1]$, r – номер строки точки в сетке $[0..M-1]$.

Топографические планы строятся для территорий, площадь которых, как правило, не превышает нескольких квадратных километров. При этом построение осуществляется без учета кривизны поверхности Земли. Из этого следует, что пиксельные расстояния между точками топографического плана практически пропорциональны географическим расстояниям между точками местности.

Пусть известны две пары точек \mathbf{a} , \mathbf{b} и \mathbf{A} , \mathbf{B} , в которых географические координаты ($\mathbf{A}_{\text{longitude}}$, $\mathbf{A}_{\text{latitude}}$, $\mathbf{B}_{\text{longitude}}$, $\mathbf{B}_{\text{latitude}}$) соответствуют пиксельным (\mathbf{a}_x , \mathbf{a}_y , \mathbf{b}_x , \mathbf{b}_y). И пусть они не лежат на одной горизонтальной или вертикальной прямой (Рисунок 4.6).

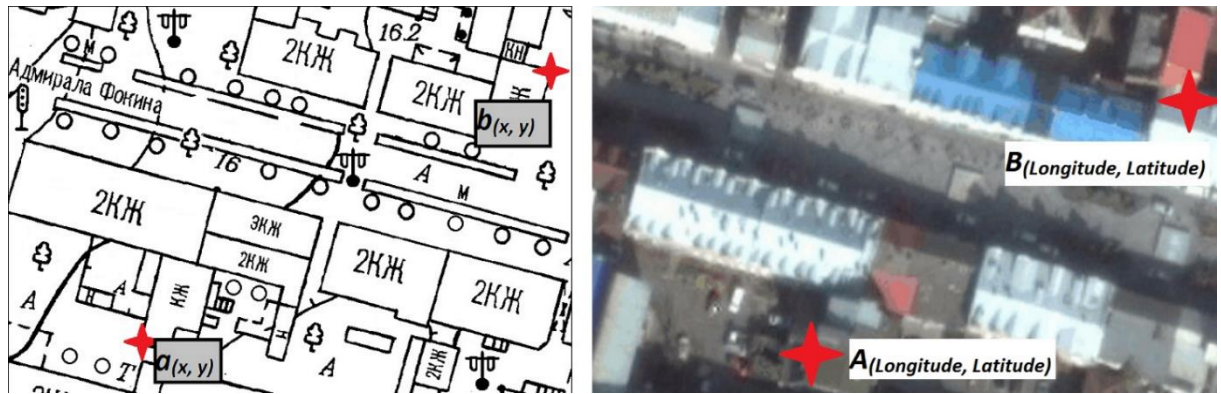


Рисунок 4.6 Точки на топоплане и соответствующие им географические координаты

Тогда географические координаты любой точки C с известными координатами c_x и c_y на топографическом будут определяться по (4.4):

$$C_{longitude} = \frac{c_x - a_x}{b_x - a_x} (B_{longitude} - A_{longitude}) + A_{longitude}$$

$$C_{latitude} = \frac{c_y - a_y}{b_y - a_y} (B_{latitude} - A_{latitude}) + A_{latitude}$$
(4.4)

Обратно, по известным географическим координатам определяется положение точки на топоплане (4.5):

$$C_x = \frac{(C_{longitude} - A_{longitude})}{(B_{longitude} - A_{longitude})} (b_x - a_x) + a_x$$

$$C_y = \frac{(C_{latitude} - A_{latitude})}{(B_{latitude} - A_{latitude})} (b_y - a_y) + a_y$$
(4.5)

Обозначим ширину и высоту топографического плана соответственно W и H . Разобьем его область на отдельные равные прямоугольные участки. Для каждой точки топографического плана p_{ij} , где $i = 0..m$ и $j = 0..n$, определим географические координаты точек P_{ij} с помощью (4.4).

Данные о высоте получаются с помощью сервиса. Для того, чтобы получить информацию о высоте необходимо передать ему список вершин. В результате обработки всех точек строится трехмерная триангуляционная поверхность и получается изображение текстуры рельефа также с помощью сервиса. Таким образом, на основании двух точек на топографическом плане построен текстурированный рельеф (Рисунок 4.7):

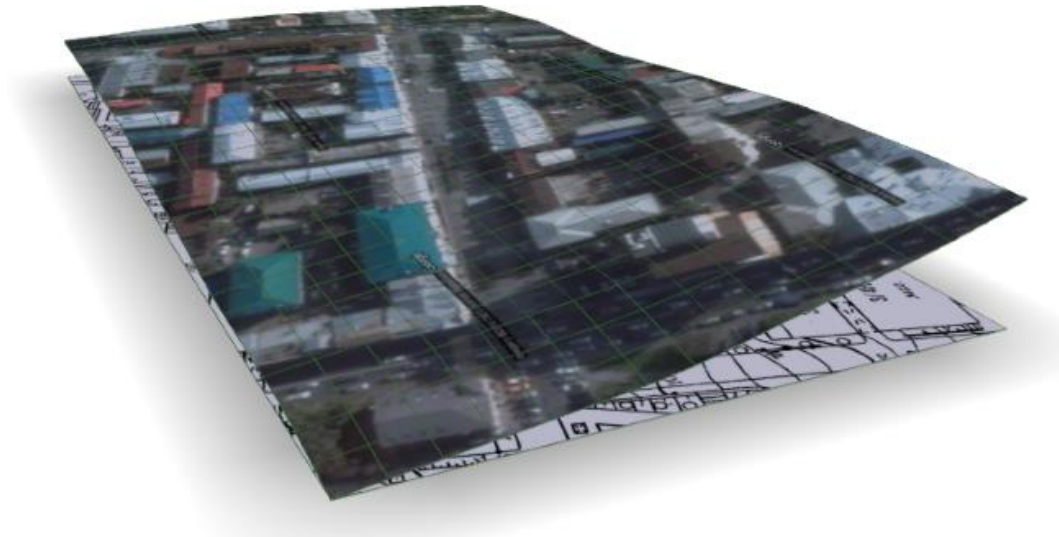


Рисунок 4.7 Текстурированный рельеф сцены

4.3. Текстурирование

Нанесение качественных текстур на сцены значительно увеличивает её реалистичность. Качественные текстуры могут быть получены с фотоснимков строений в расчете на одну сторону здания одна текстура. Однако, в условиях городской обстановки выполнить качественные фотоснимки зданий весьма затруднительно, т.к. для этого требуется сделать фотоснимок стены здания, находясь в определенной позиции съемки, которая может быть преграждена другими объектами.

Наличие неровностей на объекте, таких как окна, балконы, рельефные орнаменты, сильно влияет на результат. Это связано с тем, что некоторые участки на нужной текстуре будут перекрыты.

Рассмотрим пример по извлечению текстуры стороны здания, которая содержит неровности в виде балконов (Рисунок 4.8).

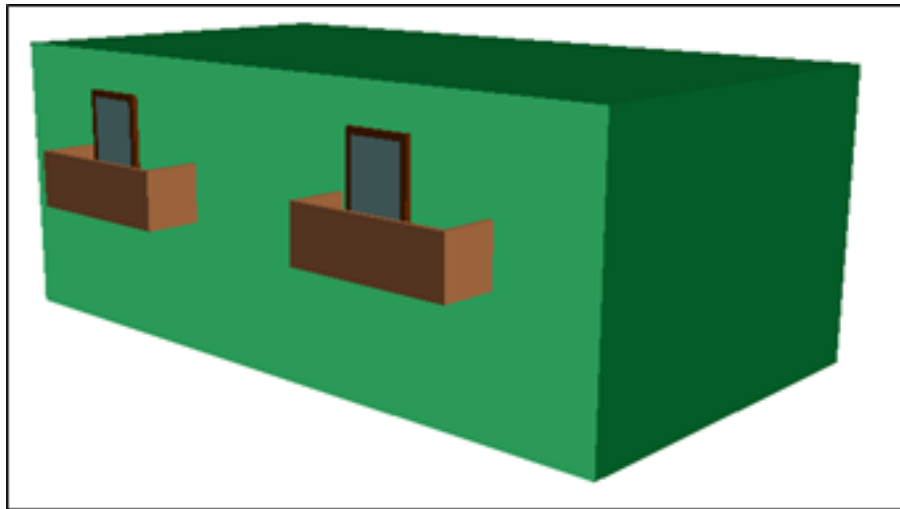


Рисунок 4.8 Условный снимок некоторого здания с балконами

На Рисунок 4.9 показан план съёмки, на котором видно место, с которого делался снимок. Также на нём отмечены области, которые не видны на Рисунок 4.8.

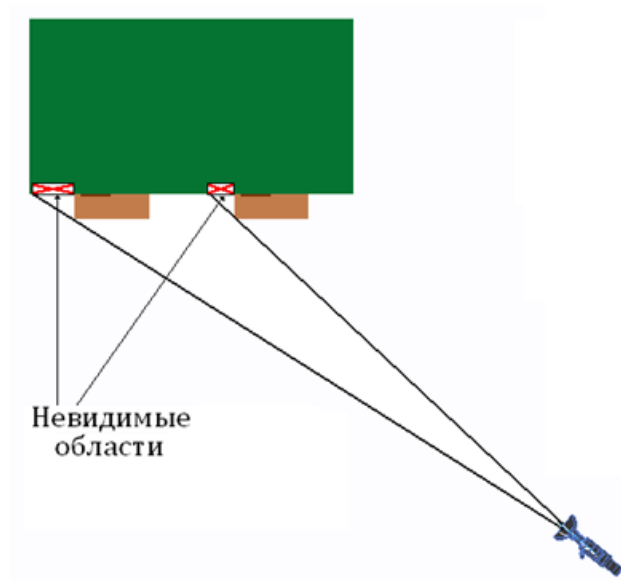


Рисунок 4.9 Вид сверху

Коррекция перспективы. Примерную текстуру стороны здания помогает получить перспективное преобразование снимка. При применении перспективного преобразования для извлечения текстуры той стороны здания, которая содержит балконы, возникает серьёзное искажение. Использование такой текстуры как на Рисунок 4.10а при реконструкции объекта нежелательно, так как дефект очень заметен.

Такие искажения (Рисунок 4.10б) могут быть исправлены «вручную» в различных графических редакторах, но это довольно трудоёмкий и продолжительный процесс.

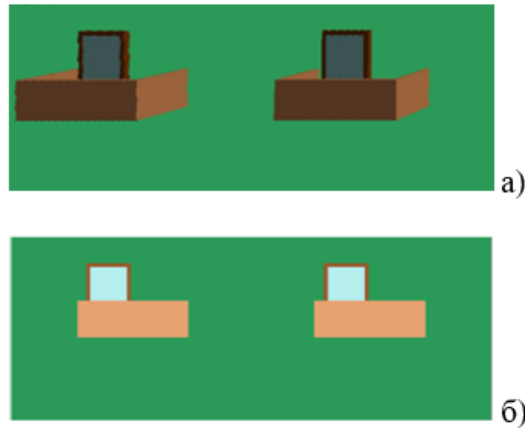


Рисунок 4.10 Искажённая текстура после перспективного преобразования (а);
Текстура без искажения (б)

Чтобы минимизировать искажения и добиться такого результата как на Рисунок 4.10а предлагается выбирать определённый ракурс съёмки. Как правило, наилучшим вариантом будет расположить камеру параллельно плоскости текстуры и совместить их центры (Рисунок 4.11).

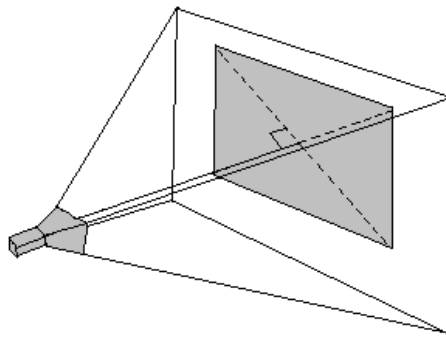


Рисунок 4.11 Наилучшее положение камеры при съёмке

На Рисунок 4.12 и Рисунок 4.13 приведены снимки одного здания снятого с разных точек, а также полученные из них перспективным преобразованием текстуры. Снимок на рисунке Рисунок 4.13 сделан с лучшей позиции, следовательно, текстура будет качественней, чем на рисунке Рисунок 4.12.



Рисунок 4.12 Исходный снимок (слева) и текстура (справа) с заметным искажением

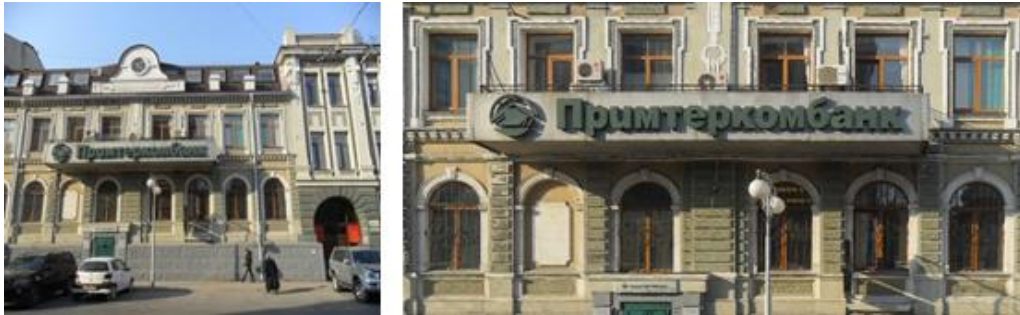


Рисунок 4.13 Исходный снимок (слева) и текстура (справа) с практически отсутствующим искажением

Для проективного преобразования необходимо задать исходную конфигурацию точек и желаемую. Рассмотрим изображение объекта (Рисунок 4.14), наблюдаемого в перспективе. У этого объекта есть грань P , которая в действительности прямоугольная, но на снимке она имеет форму неправильного четырёхугольника. С помощью перспективного преобразования получим реальную геометрию грани P . Вершинами P будут точки p_1 , p_2 , p_3 и p_4 с известными координатами соответственно (u_1, v_1) , (u_2, v_2) , (u_3, v_3) и (u_4, v_4) . Будем считать, начало координат находится в левом верхнем углу снимка, а ось ординат направлена вниз.

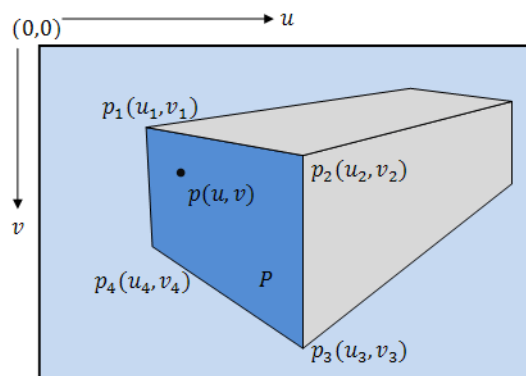


Рисунок 4.14 Условный снимок некоторого объекта

Форма будущей текстуры Q должна быть прямоугольной (Рисунок 4.15). Пусть её высота и ширина равны соответственно H и W . Координаты вершин q_1 , q_2 , q_3 и q_4 будут соответственно равны $(0, 0)$, $(W, 0)$, (W, H) и $(0, H)$. Так как Q является растровым рисунком, то начало координат будет находиться в левом верхнем углу и совпадать с точкой q_1 , а ось ординат будет направлена вниз:

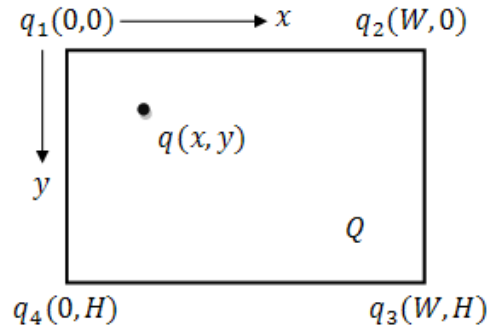


Рисунок 4.15 Q – будущая текстура

Формулы перспективного преобразования. Каждой точке из области Q соответствует другая точка из P . Пусть $p(u, v)$ – любая точка из области P , которая при перспективном преобразовании переходит в точку $q(x, y)$ из Q . Тогда получим следующие формулы перспективного преобразования (4.6):

$$\begin{cases} u = \frac{a_1x + a_2y + a_3}{a_7x + a_8y + 1} \\ v = \frac{a_4x + a_5y + a_6}{a_7x + a_8y + 1} \end{cases} \quad (4.6)$$

В формуле (4.6) есть некоторые числа $a_i (i=1..8)$, которые остаются неизвестными. При перспективном преобразовании вершины области P перейдут в вершины области Q (4.7):

$$p_j(u_j, v_j) \rightarrow q_j(x_j, y_j), j=1..4 \quad (4.7)$$

Следовательно, в общем виде получим четыре пары уравнений (4.8):

$$\begin{cases} u_j = \frac{a_1x_j + a_2y_j + a_3}{a_7x_j + a_8y_j + 1} \\ v_j = \frac{a_4x_j + a_5y_j + a_6}{a_7x_j + a_8y_j + 1} \end{cases}, j=1..4 \quad (4.8)$$

Приведя уравнения системы (4.8) к линейному виду получим (4.9):

$$\begin{cases} u_1 = a_1x_1 + a_2y_1 + a_3 - a_7u_1x_1 - a_8u_1y_1, \\ v_1 = a_4x_1 + a_5y_1 + a_6 - a_7v_1x_1 - a_8v_1y_1, \\ u_2 = a_1x_2 + a_2y_2 + a_3 - a_7u_2x_2 - a_8u_2y_2, \\ v_2 = a_4x_2 + a_5y_2 + a_6 - a_7v_2x_2 - a_8v_2y_2, \\ u_3 = a_1x_3 + a_2y_3 + a_3 - a_7u_3x_3 - a_8u_3y_3, \\ v_3 = a_4x_3 + a_5y_3 + a_6 - a_7v_3x_3 - a_8v_3y_3, \\ u_4 = a_1x_4 + a_2y_4 + a_3 - a_7u_4x_4 - a_8u_4y_4, \\ v_4 = a_4x_4 + a_5y_4 + a_6 - a_7v_4x_4 - a_8v_4y_4. \end{cases} \quad (4.9)$$

Запишем систему уравнений (4.9) в матричном виде (4.10):

$$Fa = b \quad (4.10)$$

где матрицы представляются в виде (4.11):

$$F = \begin{pmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -u_1x_1 & -u_1y_1 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -v_1x_1 & -v_1y_1 \\ x_2 & y_2 & 1 & 0 & 0 & 0 & -u_2x_2 & -u_2y_2 \\ 0 & 0 & 0 & x_2 & y_2 & 1 & -v_2x_2 & -v_2y_2 \\ x_3 & y_3 & 1 & 0 & 0 & 0 & -u_3x_3 & -u_3y_3 \\ 0 & 0 & 0 & x_3 & y_3 & 1 & -v_3x_3 & -v_3y_3 \\ x_4 & y_4 & 1 & 0 & 0 & 0 & -u_4x_4 & -u_4y_4 \\ 0 & 0 & 0 & x_4 & y_4 & 1 & -v_4x_4 & -v_4y_4 \end{pmatrix}, a = \begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \\ a_8 \end{pmatrix}, b = \begin{pmatrix} u_1 \\ v_1 \\ u_2 \\ v_2 \\ u_3 \\ v_3 \\ u_4 \\ v_4 \end{pmatrix} \quad (4.11)$$

После подстановки значений координат точек q_1, q_2, q_3 и q_4 матрица F примет вид (4.12):

$$F = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ W & 0 & 1 & 0 & 0 & 0 & -Wu_2 & 0 \\ 0 & 0 & 0 & W & 0 & 1 & -Wv_2 & 0 \\ W & H & 1 & 0 & 0 & 0 & -Wu_3 & -Hu_3 \\ 0 & 0 & 0 & W & H & 1 & -Wv_3 & -Hv_3 \\ 0 & H & 1 & 0 & 0 & 0 & 0 & -Hu_4 \\ 0 & 0 & 0 & 0 & H & 1 & 0 & -Hv_4 \end{pmatrix} \quad (4.12)$$

Для решения системы можно воспользоваться любым из существующих методов решения СЛАУ.

Точные формулы нахождения коэффициентов приведены в (4.13).

$$\begin{aligned}
a_1 &= \frac{u_2 u_4 (v_3 - v_1) + u_2 u_3 (v_1 - v_4) + u_1 u_4 (v_2 - v_3) + u_1 u_3 (v_4 - v_2)}{W (u_3 v_2 - u_2 v_3 + u_2 v_4 - u_4 v_2 + u_4 v_3 - u_3 v_4)} \\
a_2 &= \frac{u_3 u_4 (v_2 - v_1) + u_3 u_1 (v_4 - v_2) + u_2 u_4 (v_1 - v_3) + u_2 u_1 (v_3 - v_4)}{H (u_3 v_2 - u_2 v_3 + u_2 v_4 - u_4 v_2 + u_4 v_3 - u_3 v_4)} \\
a_3 &= u_1 \\
a_4 &= \frac{(u_2 v_1 - u_4 v_1 - u_1 v_2 + u_4 v_2) v_3 - (u_2 v_1 - u_3 v_1 - u_1 v_2 + u_3 v_2) v_4}{W (u_3 v_2 - u_2 v_3 + u_2 v_4 - u_4 v_2 + u_4 v_3 - u_3 v_4)} \\
a_5 &= \frac{(u_3 v_2 - u_1 v_2 + u_1 v_3 - u_2 v_3) v_4 - (u_3 v_2 - u_4 v_2 - u_2 v_3 + u_4 v_3) v_1}{H (u_3 v_2 - u_2 v_3 + u_2 v_4 - u_4 v_2 + u_4 v_3 - u_3 v_4)} \\
a_6 &= v_1 \\
a_7 &= \frac{(u_4 + u_3)(v_2 - v_1) - (u_1 - u_2)(v_3 - v_4)}{W (u_3 v_2 - u_2 v_3 + u_2 v_4 - u_4 v_2 + u_4 v_3 - u_3 v_4)} \\
a_8 &= \frac{(u_2 + u_3)(v_4 - v_1) - (u_1 - u_4)(v_2 - v_3)}{H (u_3 v_2 - u_2 v_3 + u_2 v_4 - u_4 v_2 + u_4 v_3 - u_3 v_4)}
\end{aligned} \tag{4.13}$$

Теперь известны все коэффициенты a_i ($i=1..8$), а это значит, что формула (4.6) полностью определена и можно приступать к заполнению области Q .

Построение текстуры. Текстура – растровое изображение, состоящее из пикселей. Пиксель характеризуется цветом и координатами на текстуре. Чтобы построить текстуру Q , необходимо каждому пикселю присвоить некоторый цвет, который будет браться с изображения, содержащего область P .

P и Q являются двумерными матрицами, элементами которых являются цвета. Q имеет размерность $[H \times W]$, следовательно, придётся сделать $H \times W$ шагов, чтобы заполнить текстуру.

Для каждого элемента $Q_{i,j}$ ($i=0..W-1; j=0..H-1$) вычисляются координаты u и v на изображении P по (4.6):

После определения координат пикселя на исходном изображении, его цвет присваивается соответствующему элементу текстуры.

Алгоритм наглядно представлен ниже (Рисунок 4.16):

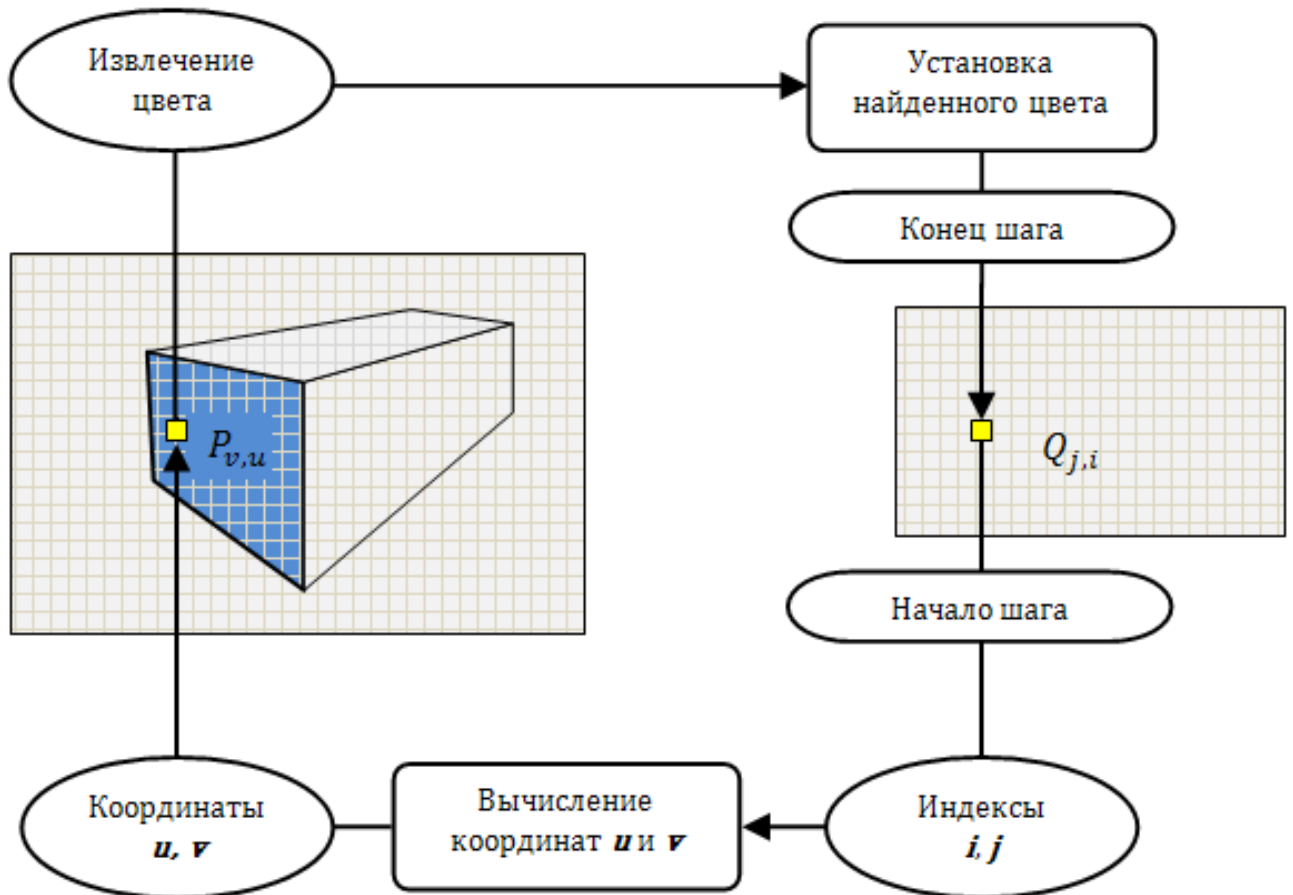


Рисунок 4.16 Алгоритм построения текстуры Q

Текстурирование зданий. Предлагается производить автоматическое текстурирование зданий фотореалистичными изображениями при наличии следующих данных:

- географического положения и направление камеры;
- координат углов стены на фотоснимке для коррекции перспективы;
- географических координат всех объектов сцены.

Современные телефоны и планшеты позволяют производить фотоснимки с записью географических координат, определяемых с помощью систем GPS или ГЛОНАСС.

Нахождение углов стены на фотоснимке довольно сложная и нетривиальная задача, поэтому выделение стены на фотоснимке происходит вручную. Для этого разработано специальное ПО для мобильных устройств (Рисунок 4.17).

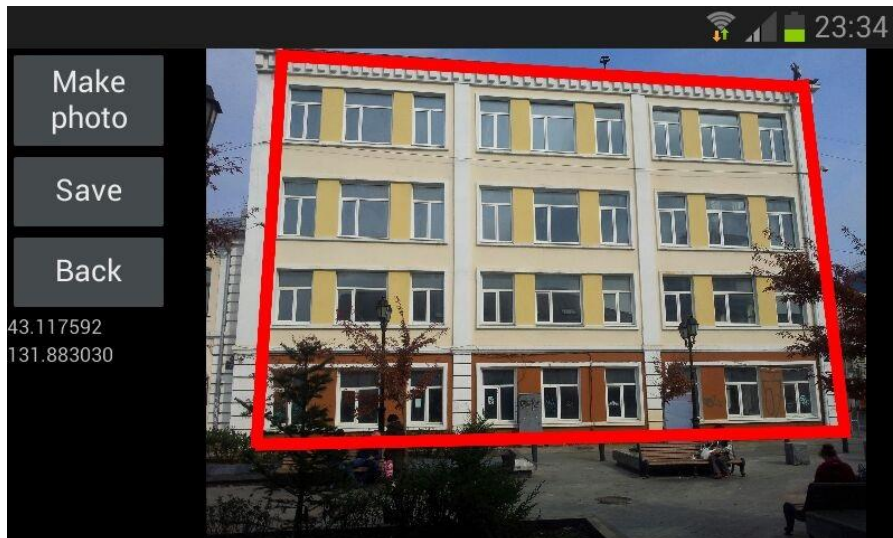


Рисунок 4.17 Мобильное приложение по работе с текстурами зданий

На основании выделенных четырех точек стены корректируется перспектива на фотоснимке (Рисунок 4.18).



Рисунок 4.18 Текстура с исправленной перспективой

Для каждого фотоснимка на модели необходимо определить здание и стену, на которую должна быть нанесена текстура. Каждый снимок содержит данные о географических координатах места и направления съемки. На основании двух географических точек, которые применялись для построения рельефа, определяются координаты места съемки на топоплане с помощью (4.4.2).

Таким образом все фотоснимки будут перенесены в плоскость топоплана. Определение стены здания, к которой принадлежит фотоснимок текстуры, осуществляется с помощью данных о направлении съемки. На топоплане проводится луч по направлению съемки из точки, с которой она производилась. По

каждому зданию определяются стены, с которыми пересекся луч. Из всех этих стен выбирается та, точка пересечения которой ближе остальных к точке съемки (Рисунок 4.19).

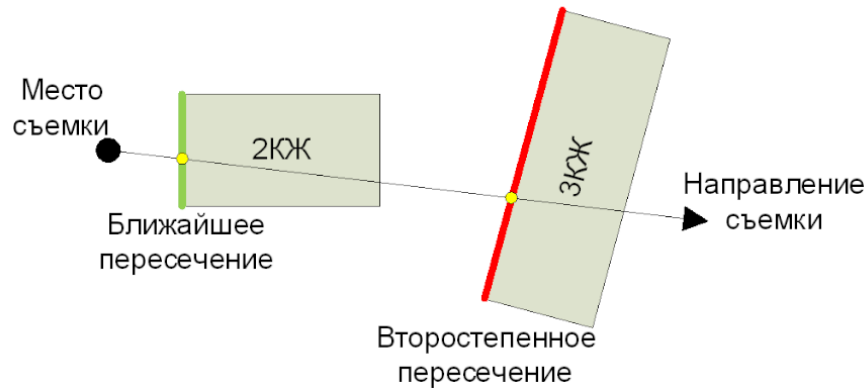


Рисунок 4.19 Определение ближайшей стены

У этой стены известны линейные размеры высоты и ширины, поэтому к ним приводится размер текстуры после коррекции перспективы.

Реалистичность любой трехмерной модели зависит не только от точности воссоздания ее геометрической формы, но и от качества нанесенных на нее текстур. Наилучший способ — это нанесение текстур, полученных из фотоизображений конкретных зданий, как было описано выше. Однако, обработка больших топографических планов даже с несколькими десятками зданий требует серьезных трудозатрат. Поэтому, помимо метода нанесения текстур, полученных из фотоизображений, предлагается метод автоматического текстурирования. Будут использоваться шаблонные текстуры (Рисунок 4.20), которые дают представление о типе здания. На основании распознанной служебной информации каждому зданию будет установлена та или иная текстура, если ранее для неё не была установлена текстура с фотоснимка.



Рисунок 4.20 Стандартные текстуры для жилого здания

Текстура здания должна состояться из композиции текстур базового набора (Рисунок 4.21, Рисунок 4.22, Рисунок 4.23, Рисунок 4.24). По вертикали базовая текстура должна повторяться столько раз, сколько у здания этажей. По горизонтали базовая текстура должна повторяться следующим числом раз (4.14):

$$k = \left\lceil \frac{\text{Высота текстуры}}{\text{Ширина текстуры}} \frac{\text{Длина стены}}{\text{Высота этажа}} \right\rceil \quad (4.14)$$

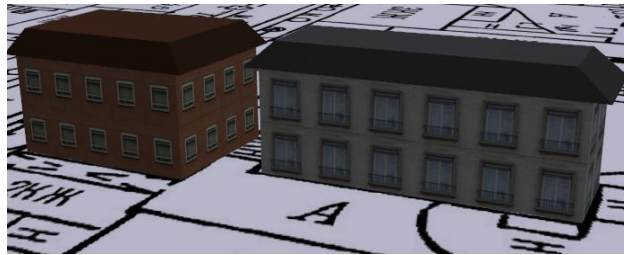


Рисунок 4.21 Здания со стандартными текстурами

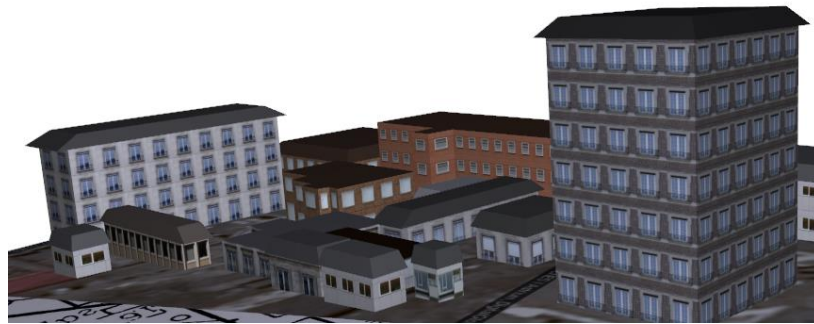


Рисунок 4.22 Здания со стандартными текстурами



Рисунок 4.23 Здания со стандартными текстурами



Рисунок 4.24 Здания со стандартными текстурами

4.4. Программная реализация

На основании предложенного метода реконструкции была разработана программная система (Приложение В. Описание разработанного программного КОМПЛЕКСА), позволяющая производить автоматическую реконструкцию трехмерной сцены на основании топографического плана.

Разработанная система включает десктопное и мобильное приложения. Десктопное приложение разработано на технологии Windows Forms (C# 7.0). Мобильное приложение работает на базе ОС Android 4.1. Общая схема разработанного программного обеспечения представлена в виде следующей схемы (Рисунок 4.25):

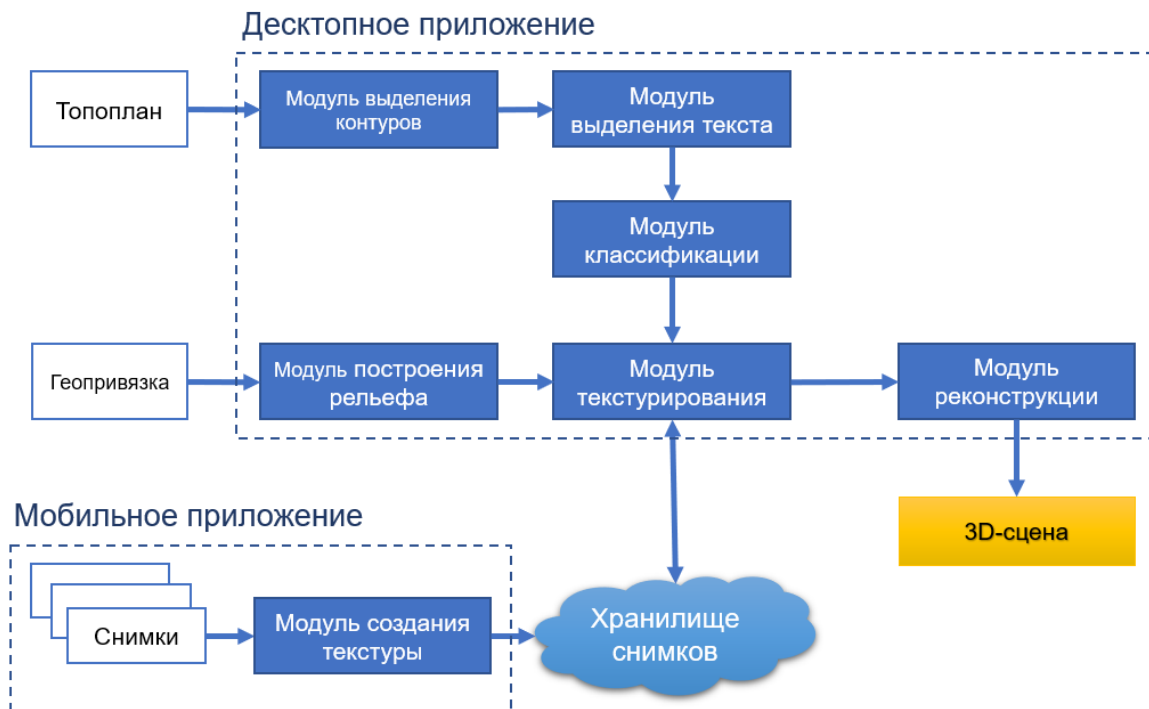


Рисунок 4.25 Общая схема реконструкции

В качестве исходных данных система использует подготовленный для обработки топографический план. Система предоставляет оператору возможность запустить автоматический поиск всех зданий и объектов. Параметры поиска могут корректироваться оператором. После автоматического поиска присутствует возможность скорректировать формы распознанных зданий, добавить или удалить объекты. Всем зданиям автоматически будет присвоена текстура, зависящая от

типа строения. На главном окне оператор может изменить параметры текстуры, типа, количества и высоты этажей здания.

Фотореалистичные текстуры могут быть добавлены как из отдельного файла, так и из специализированного хранилища. Данное хранилище представляет собой набор фотоснимков, у каждого из которых задано определенно составленное имя файла. Формат названия файла выглядит следующим образом: *[Уникальный идентификатор].[Дата съемки].[Формат изображения]*.

Уникальный идентификатор даётся каждому файлу при добавлении в хранилище. Также, в хранилище содержится конфигурационный файл со списком изображений. В нем записана следующая информация:

- Имя файла;
- Географические координаты места съемки;
- Направление съемки;
- Координаты четырехугольника на изображении, выделенного пользователем.

В хранилище информация попадает с помощью мобильного приложения, которое осуществляет съемку. Оператор выделяет область стены здания (четырёхугольник) и отправляет в хранилище. При этом в хранилище будет добавлена сделанная фотография, а в информационный файл добавится информация о снимке. Настройки десктопного приложения позволяют указать путь к хранилищу (адрес FTP-сервера, Google Диск, Яндекс Диск и т.п.).

Десктопное приложения для загрузки снимков обращается к указанному хранилищу, и в конфигурационном файле ищет те снимки, которые по указанным в нем координатам находятся в пространстве топоплана. Эти фотоснимки загружаются из хранилища. При построении сцены на снимках корректируется перспектива на основании размеров стены здания и указанного пользователем четырехугольника при съемке.

Все полученные с топоплана данные используются для построения трехмерной сцены, которая может быть экспортирована в файл трехмерной

графики (.x). Экспортированные модели используются для чтения сторонними визуализаторами, в которых сцена может быть дополнительно обработана. Данные модели могут быть использованы в презентационных задачах.

Разработанная программная система позволяет экспортировать все найденные на сцене контуры в файл формата .SVG. Это позволит использовать контуры на топоплане для других исследовательских и практических задач.



Рисунок 4.26 Главное окно программной системы

Для добавления рельефа и фотореалистичных текстур необходимо воспользоваться соответствующим модулем. Модуль позволяет по двум заданным точкам топоплана построить геолокационное пространство, которое будет связывать координаты топоплана и географические координаты объектов (Рисунок 4.27).

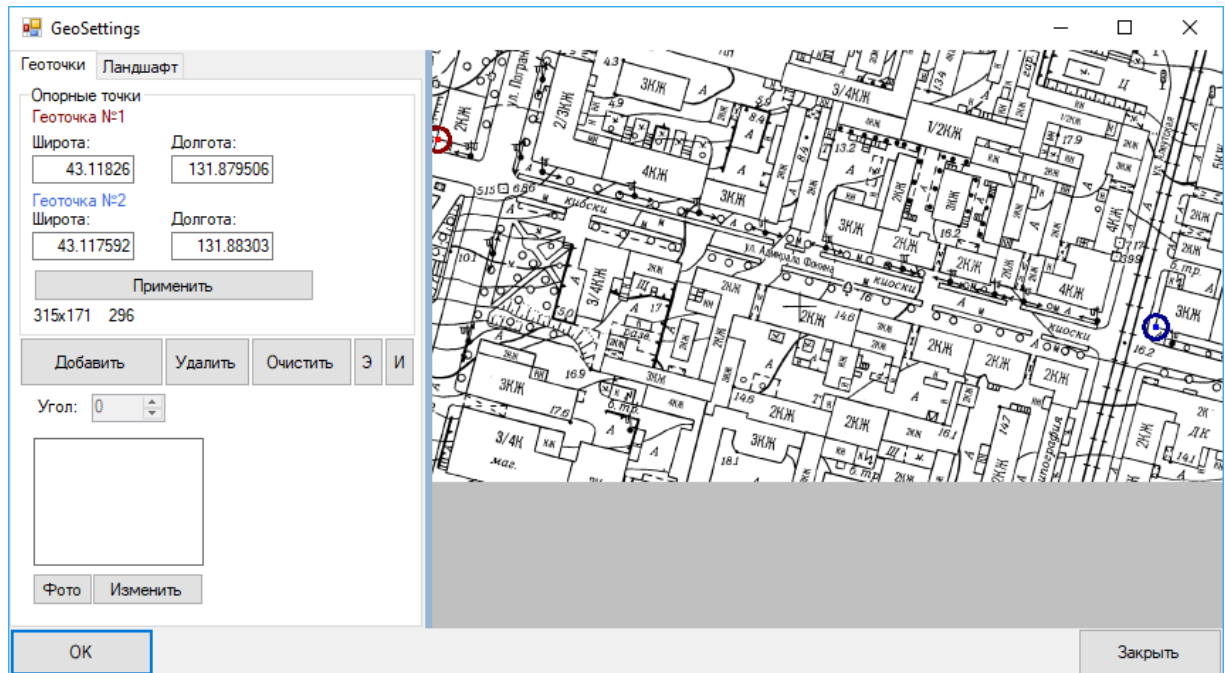


Рисунок 4.27 Модуль установки геопривязки и текстурирования.

Возможности разработанной программной системы:

- 1) Автоматический поиск зданий и объектов на топоплане.
- 2) Ручная правка результатов поиска, добавление и удаление зданий и объектов.
- 3) Построение текстурированного рельефа.
- 4) Текстурирование зданий как шаблонными текстурами, так и фотореалистичными. Использование облачного хранилища фотореалистичных текстур.
- 5) Предварительная визуализация сцены.
- 6) Экспорт всей сцены или только определенных типов объектов.
- 7) Экспорт контуров в формат векторной графики.

Основным достоинством разработанной программной системы является автоматическое распознавание объектов на топографическом плане, что значительно сокращает время реконструкции сцены.

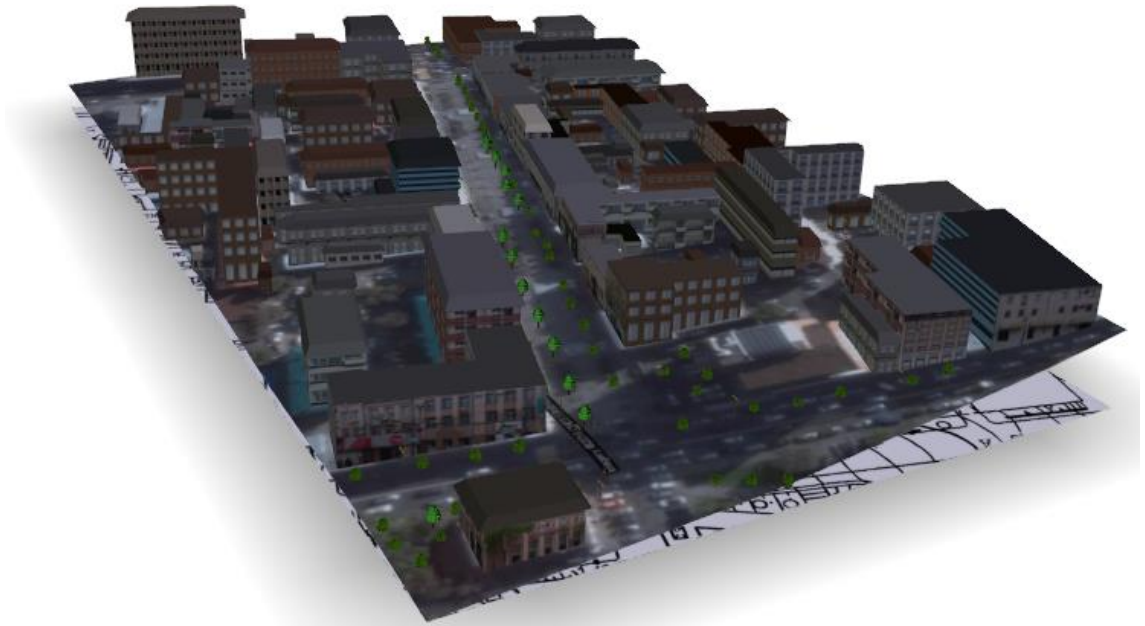


Рисунок 4.29 Сцена «Фокина» с реальными текстурами



Рисунок 4.30 Увеличенный фрагмент

Центральный район города Владивостока (выше площади Борцов за власть Советов) (Рисунок 4.31, Рисунок 4.32, Рисунок 4.33, Рисунок 4.34).

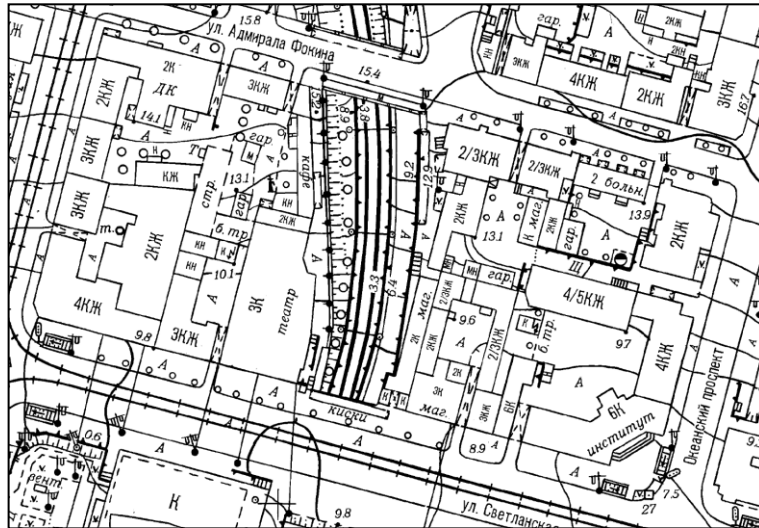


Рисунок 4.31 Сцена "Кварталы". Топоплан сцены



Рисунок 4.32 Сцена "Кварталы". Распознанные контуры и служебная информация

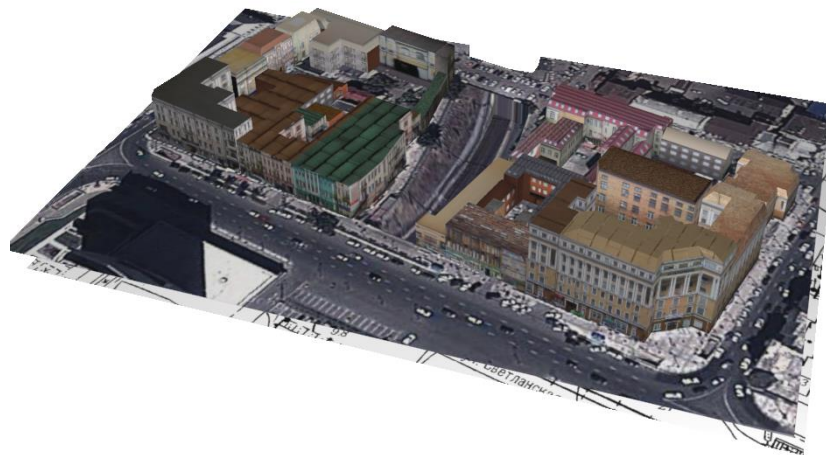


Рисунок 4.33 Сцена "Кварталы" с разных видов

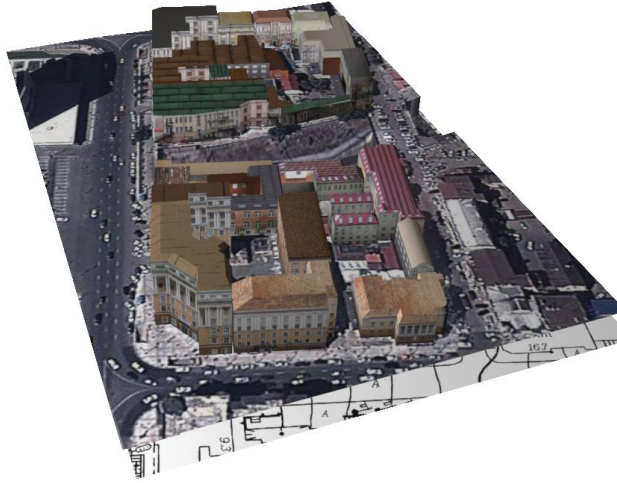


Рисунок 4.34 Сцена "Кварталы" с разных видов

Небольшой участок городской сцены (Рисунок 4.35, Рисунок 4.36).

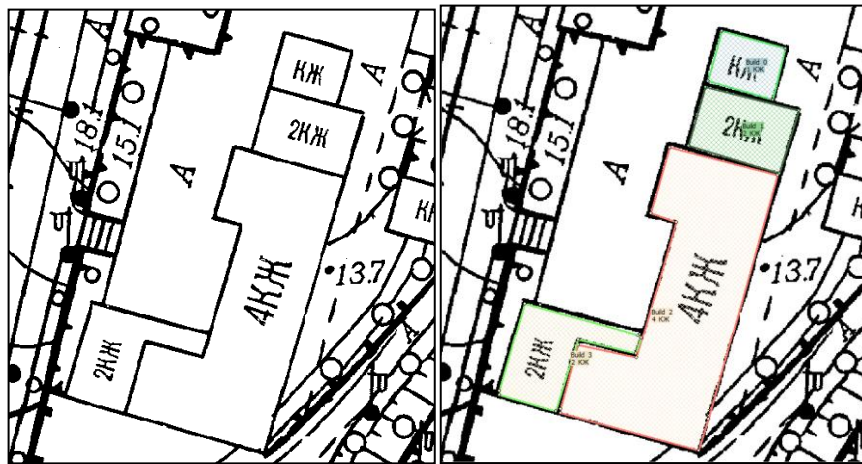


Рисунок 4.35 Топоплан сцены «Алеутская». Исходный (слева) и с распознанной информацией (справа)

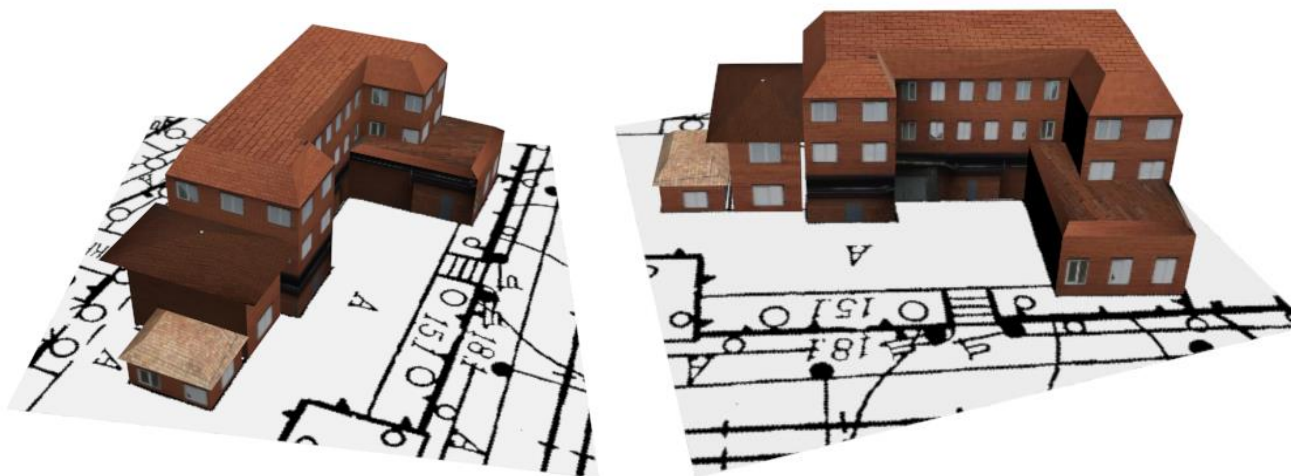


Рисунок 4.36 Топоплан сцены «Алеутская». Трёхмерная реконструкция.

Автоматически реконструированная сцена центральной части г. Владивосток (1×1 км) (Рисунок 4.37):



Рисунок 4.37 Сцена "Центр". а) исходный топоплан; б) без текстур; в) с текстурами и рельефом

Общая информация о реконструированных сценах представлена ниже (Таблица 4.1). Во всех случаях построение трехмерной модели заняло меньше минуты.

Таблица 4.1. Результаты реконструкции сцен

Показатель	Сцена		
	Центр	Фокина	Кварталы
Размер топоплана (пикселей/км)	7874x7874 / 1x1	2480x1356 / 0,7x0,4	814x425 / 0,23x0,12
Количество зданий	539	115	38
Верно распознано зданий (шт, %)	495 (91%)	111 (95%)	36 (94%)
Кол-во объектов (шт)	1415	133	15
Кол-во верно распознанных объектов (шт, %)	1147 (81%)	119 (89%)	13 (86%)
Время выделения контуров (мин)	3,5	2	<1
Время распознавания текста (мин)	2	<1	<1
Время выделения объектов (мин)	1,5	<1	<1
Итого времени (мин)	7	<4	<3

Полученные результаты экспериментов свидетельствуют о том, что предложенный метод дает точные результаты реконструкции. При этом время выполнения реконструкции достаточно небольшое.

Заключение

Разработка методов, алгоритмов и программных систем в области реконструкции сцен городского пространства остается актуальной темой исследования. В диссертационном исследовании была поставлена цель создания метода реконструкции сцен городского пространства на основании топографического плана, включающего все необходимые для этого этапы, в т. ч. реализацию программного комплекса, обеспечивающего автоматизацию всего процесса. Для достижения указанной цели были сформированы задачи, в процессе которых получены следующие практические и теоретические результаты:

1. Предложен и доведен до программной реализации метод автоматической реконструкции городских сцен на основании топоплана местности. Программный комплекс, реализующий данный метод, успешно применяется в деятельности реальной организации для визуализации городского пространства. Особую ценность обеспечивает возможность комплекса обрабатывать протяженные сцены за короткое время.
2. Разработан алгоритм выделения и идентификации объектов на топоплане. Предложенный подход, позволяет иметь высокий процент успешного выделения контуров на бинарном изображении. Данный метод может быть использован независимо от контекста рассматриваемой задачи.
3. Предложен метод моделирования земной поверхности и автоматического текстурирования сцены по набору цифровых изображений и геолокационных данных. Использование данного метода увеличило реалистичность создаваемых трехмерных моделей городских сцен.
4. Проведены реконструкции трехмерных сцен на основании реальных топопланов и фотографий зданий. Полученные готовые трехмерные модели могут широко использоваться на практике в различных областях.

Дальнейшее развитие результатов автор видит в развитии репозитория данных и повышении точности алгоритмов.

Список литературы

1. Akbarzadeh A., Frahm J.-M., Mordohai P., Clipp B., Engels C., Gallup D., Merrell P., Phelps M., Sinha S. N., Talton B., Wang L., Yang Q., Stewénius H., Yang R., Welch G., Towles H., Nistér D., Pollefeys M. Towards Urban 3D Reconstruction from Video // Third International Symposium on 3D Data Processing, Visualization and Transmission. – 14-16 June 2016, Chapel Hill. – Vol. 1. – PP.1-8.
2. Autocad. URL: <https://www.autodesk.ru/products/autocad> (дата обращения 01.12.2020)
3. Autocad. URL: <https://www.autodesk.ru/products/autocad/included-toolsets/autocad-raster-design> (дата обращения 01.12.2020)
4. Autodesk. URL: <http://www.autodesk.com> (дата обращения 01.12.2020)
5. Baillard C. and Maitre H. 3D reconstruction of urban scenes from aerial stereo imagery: a focusing strategy // Computer Vision and Image Understanding. – 1999. – PP. 244-258.
6. Baillard C., Zisserman A. A plane-sweep strategy for the reconstruction of buildings from multiple images // In ISPRS Congress and Exhibition. – 2000. – PP. 105-112.
7. Baillard, C., Schmid, C., Zisserman, A. and Fitzgibbon, A. Automatic line matching and 3D reconstruction of buildings from multiple views // ISPRS Conference on Automatic Extraction of GIS Objects from Digital Imagery. – 1999. – Vol. 32. – №3. – PP. 69-80.
8. Canny J. A computational approach to edge detection // IEEE Transactions on Pattern Analysis and Machine Intelligence. – 1986. – № 8(6). – PP. 679-698.
9. Canny J. An opportunistic global path planner // In: Algorithmica. –1993. – Vol. 10. – № 2-4. – PP. 102-120.
10. Canny J. MultiView: Improving trust in group video conferencing through spatial faithfulness // SIGCHI Conf. on Human Factors in Computing Systems. – 2007. – PP. 1465-1474.

11. Canny J. A computational approach to edge detection // IEEE Transactions on Pattern Analysis and Machine Intelligence. – 1986. – Vol. 8. – PP. 679-698.
12. Chigorin A., Konushin A. A system for large-scale automatic traffic sign recognition and mapping // ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences. – 2013. – Vol. 2. – № 3. – PP. 13-17.
13. Cityengine. URL: <https://www.esri.com/en-us/arcgis/products/esri-cityengine> (дата обращения 01.12.2020)
14. Colombo C., Bimbo A., Pernici F. Metric 3D Reconstruction and Texture Acquisition of Surfaces of Revolution from a Single Uncalibrated View // IEEE Transactions on pattern analysis and machine intelligence. – 2005. – Vol. 27. – №1. – PP. 36-47.
15. Cornelius H., Sara R., Martinec D., Pajdla T., Chum O., Matas J. Towards Complete Free-Form Reconstruction of Complex 3D Scenes from an Unordered Set of Uncalibrated Images // Technical report. – 2004. – PP 1-12.
16. Cremers D., Rousson M., Deriche R. A review of statistical approaches to level set segmentation: integrating color, texture, motion and shape // International journal of computer vision. – 2007. – Vol. 72. – № 2. – PP. 195-215.
17. Cremers D., Tischhäuser F., Weickert J., Schnörr C. Diffusion snakes: Introducing statistical shape knowledge into the Mumford-Shah functional // International journal of computer vision. – 2002. – Vol. 50. – № 3. – PP. 295-313.
18. D.H. Ballard. Generalizing the Hough Transform to Detect Arbitrary Shapes // Pattern Recognition. – 1981. – Vol. 13. – №2. – PP. 11-122.
19. Dinh H.Q., Slabaugh G., Turk G. Reconstructing Surfaces Using Anisotropic Basis Functions // International Conference on Computer Vision (ICCV). – 2001. – PP. 606-613.
20. DirectX. URL: <http://paulbourke.net/dataformats/directx/> (дата обращения 01.12.2020)
21. Easytrace. URL: <http://www.easytrace.com/> (дата обращения 01.12.2020)

22. Endres F., Hess J., Sturm J., Cremers D., Burgard W. 3-D mapping with an RGB-D camera // IEEE Transactions on Robotics. – 2014. – Vol. 30. – № 1. – PP. 177-187.
23. Engel J., Schöps T., Cremers D. LSD-SLAM: Large-scale direct monocular SLAM // European Conference on Computer Vision. – 2014. – PP. 834-849.
24. Faugeras O., Robert L., Laveau S., Csurka G., Zeller C., Gauclin C., Zoghlami I. 3D Reconstruction of Urban Scenes from Image Sequences // Computer Vision and Image Understanding (CVIU). – 1998. – Vol. 3. – № 69. – PP. 292-309.
25. Gonzalez R.C., Woods R.E. Digital Image Processing // Prentice Hall. – 1993. – P. 883.
26. Harville M., Culbertson W. B., Sobel I., Gelb D., Fitzhugh A., Tanguay D. Practical Methods for Geometric and Photometric Correction of Tiled Projector // 2006 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'06). – 2006. – PP. 1-8.
27. Knopp J., Prasad M., Gool L.V. Scene cut: Class-specific object detection and segmentation in 3D scenes // Proceedings of international conference on 3D imaging, modeling, processing, visualization and transmission. – 2011. – PP. 180-187.
28. Konushin A. Kononov V., Konushin V. People Tracking Algorithm for Human Height Mounted Cameras // Proceedings of European Conference of Computer Vision. – 2011. – PP.163-172
29. Konushin A., Barinova O., Konushin V., Yakubenko A., Lee K., Lim H. Fast Automatic Single-View 3-d Reconstruction of Urban Scenes // Proceedings of European Conference of Computer Vision. – 2008. – Vol. 5303. – № 1. – PP. 100-113.
30. Konushin A., Barinova O., Shapovalov R., Sudakov S., Velizhev A. Efficient road mapping via interactive image segmentation. // Object Extraction for 3D City Models, Road Databases and Traffic Monitoring (CMRT). – 2009. – Vol. XXXVIII. – PP. 1-6

31. Konushin A., Vezhnevets V., Ignatenko A. Interactive image-based urban modeling // Proc. of PIA-2007. – 2007. – PP. 63-68.
32. Lee C.Y. An Algorithm for Path Connections and Its Applications // IRE Transactions on Electronic Computers. – 1961. – Vol. 10. – № 2. – PP. 364-365.
33. Lin B. C., You R. J. Tensor-based quality prediction for building model reconstruction from LIDAR data and topographic map // International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. – 2012. Vol. XXXIX-B7. – PP. 437-441.
34. Mapinfo. URL: <http://mapinfo.ru/product/mapinfo-professional> (дата обращения 01.12.2020)
35. Photomodeler. URL: <http://www.photomodeler.com> (дата обращения 01.12.2020)
36. Pollefeys M., Gool L. V., Vergauwen M., Cornelis K., Verbiest F., Tops, J. Image-based 3d recording for archaeological field work // Computer Graphics and Applications. – 2003. – № 23(3). – PP. 20-27.
37. Pollefeys M., Koch R., Vergauwen M., Deknuydt A. A., Gool L. J. V. Three-dimensional Scene Reconstruction from Images // In Proc. –2000. – Vol. 3958. – PP. 215-226.
38. Sato T., Kanbara M., Yokoya N., Takemura H. 3-D modeling of an outdoor scene by multi-baseline stereo using a long sequence of images. // Pattern Recognition. – 2002. – Vol. 3. – PP. 581-584.
39. Schmid C., Zisserman A. Automatic line matching across views // IEEE Conference on Computer Vision and Pattern Recognition. – 1997. – PP. 666-671.
40. Serra J. Image Analysis // Mathematical Morphology. – Academic Press. – 1982. – PP.184-185.
41. Sobel I., Feldman G. A 3x3 Isotropic Gradient Operator for Image Processing. – Pattern Classification and Scene Analysis. – 1973. – PP. 271-272.
42. SVG. URL: <https://svgwg.org/> (дата обращения 01.12.2020)
43. Taillandier F., Deriche R. Reconstruction of 3d linear primitives from multiple views for urban areas modelisation // Proceedings of the ISPRS Commission 3

- Symposium on Photogrammetric Computer Vision (PCV'02). – 2002. – Vol. 34 – PP.10-18.
44. Urban M., Pajdla T., Hlavac V. Consistent projective reconstruction from multiple views // *Confluence of Computer Vision and Computer Graphics*. – 2000. – PP. 49-67.
 45. Vezhnevets V., Konushin A. Automatic building texture completion // *GraphiCon*. – 2007. – PP. 174-177.
 46. Vezhnevets V., Konushin A., Ignatenko A. Interactive image-based urban modelling // *Photogrammetric Image Analysis*. – 2007. – PP. 63-69.
 47. VPmap. URL: <https://www.softelec.com/enu/products/gis/vpmap.htm> (дата обращения 01.12.2020)
 48. Zhang Z., Wu J., Zhang Y., Zhang J. Multi-view 3d city model generation with image sequences // *Vision techniques for digital architectural and archaeological archives*. – 2013. – Vol. XXXIV. – № 5. – PP. 262-267.
 49. Zisserman A. Multiple view geometry in computer vision. – 2000. – PP. 43-44.
 50. Аникина Г.А., Поляков М.Г., Романов Л.Н., Цветков В.Я. О выделении контура изображения с помощью линейных обучаемых моделей // *Известия АН СССР. Техническая кибернетика*. – 1980. – № 6. – С. 36–43.
 51. Афонасенко А.В. Распознавание структурированных символов на основании методов морфологического анализа // *Известия Томского политехнического университета. Инжиниринг георесурсов*. – 2007. – Т. 311. – № 5. – С. 119-123.
 52. Бобков В.А, Роньшин Ю.И., Кудряшов А.П. Сопоставление линий по трем видам пространственной сцены // *Информационные технологии и вычислительные системы*. – 2006. – №2. – С. 71–78.
 53. Бобков В.А., Белов С.Б., Май В.П., Калачева Е.В., Морозов М.А. Векторизация растровых изображений // *Информационные технологии*. – 1998. – № 7. – С. 7-11.

54. Бобков В.А., Борисов Ю.С., Кудряшов А.П. Калибровка изображений городской обстановки // Информационные технологии и вычислительные системы. – 2009. – № 1. – С. 72–83.
55. Бобков В.А., Борисов Ю.С., Кудряшов А.П. Реконструкция и визуализация городской обстановки по изображениям // 13-я Всероссийская конференция «Математические методы распознавания образов». – 2007. – С. 282-285.
56. Бобков В.А., Голенков Е.А., Клещев А.С., Нурминский Е.А. Исследования в области информатики в ИАПУ ДВО РАН // Вестник Дальневосточного отделения Российской академии наук. – 2006. – № 4. – С. 51-63.
57. Бобков В.А., Кудряшов А.П., Мельман С.В., Морозов М.А.. Восстановление траектории движения робота и реконструкция среды по изображениям // Вестник Дальневосточного отделения Российской академии наук. – 2016. – № 4 (188). – С. 60–69.
58. Бобков В.А., Роньшин Ю.И., Кудряшов А.П. Идентификация линий по некалиброванным видам городской обстановки // Информационные технологии и вычислительные системы. – 2007. – №1. – С. 63–71.
59. Визильтер Ю.В., Горбацевич В.С., Вишняков Б.В., Сидякин С.В. Поиск объектов на изображении с использованием морфлетных описаний // Компьютерная оптика. – 2017. – Т. 41. – № 3. – С. 406-411.
60. Визильтер Ю.В., Горбацевич В.С., Рубис А.Ю. и Выголов О.В. Сравнение изображений по форме с использованием диффузной морфологии и диффузной корреляции // Компьютерная оптика. – 2015. – Т. 39. – № 2. – С. 265-274.
61. ГИС «»Панорама. URL: https://gisinfo.ru/products/map12_prof.htm (дата обращения 01.12.2020)
62. Гуненко И.В., Салтыкова Ю.С., Середович В.А., Комиссаров Д.В. Технология построения трехмерных моделей городов на основе топографических планов // Интерэкспо Гео-Сибирь. – 2006. – № 2. – С. 153-157.

63. Дышленко С.Г. Трехмерное моделирование в ГИС // ПНиО. – 2014. – №2 (8). – С. 28-34.
64. Ершова А.А. Применение по AutoCAD для создания топографических планов и дальнейшего управления инженерными данными на всех стадиях проектирования // Интерэкспо Гео-Сибирь. – 2013. – №1. – С. 148-151.
65. Журкин И.Г., Хлебникова Т.А. Технология получения измерительной трехмерной видеосцены по материалам аэрокосмических съемок // Геодезия и картография. – 2009. – № 8. – С.43-48.
66. Зейналов Р.Ш., Конушин А.С., Якубенко А.А. Бесконтактные измерения характеристик движения объекта с использованием инфракрасных маркеров // Новые информационные технологии в автоматизированных системах. – 2012. – №15. – С. 51-58.
67. Ключин Е.Б., Киселев М.И., Михелев Д.Ш., Фельдман В.Д. Инженерная геодезия: Учебник для вузов / Ключин Е.Б., Киселев М.И., Михелев Д.Ш., Фельдман В.Д. – М: Издательский центр «Академия», 10-е издание, – 2004. – 481 С.
68. Кобзева Е.А. Создание топографических планов масштаба 1: 2000 для разработки градостроительной документации средних и малых населенных пунктов // Геоматика. – 2010. – № 3. – С. 76-79.
69. Конушин А.С., Кривовязь Г.Р., Птенцов С.В. Алгоритм плотной стерео реконструкции на основе контрольных точек и разметки плоскостями // Программные продукты и системы. – 2012. – №4. – С. 237–242.
70. Конушин А.С., Чигорин А.А. Система автоматического картографирования знаков дорожного движения // Программные продукты и системы. – 2013. – №3. – С. 288-291.
71. Кудряшов А.П. Идентификация линий по некалиброванным видам городской обстановки // XXX Дальневосточная математическая школа-семинар имени академика Золотова. – 2007. – С. 46–47.

72. Кудряшов А.П. Извлечение и сопоставление точечных особенностей // Электронный журнал "Исследовано в России". – 2007. – С. 1095–1104.
73. Кудряшов А.П. Модификация кросскорреляционного метода // XXIV Дальневосточная математическая школа-семинар имени академика Золотова. – 2003. – С. 80-81.
74. Кудряшов А.П. Реконструкция трехмерных сцен городской обстановки // Информационные технологии. – 2009. – №7. – С. 63-68.
75. Кудряшов А.П. Трехмерная реконструкция сцен городской обстановки по ограниченной, некалиброванной последовательности фотоизображений // XXXI Дальневосточная математическая школа-семинар имени академика Золотова. – 2008. – С.22.
76. Кудряшов А.П., Соловьев И.В. Выделение и распознавание текстовой информации на топографическом плане // Информатика и системы управления. – 2020. – № 3 (65). – С. 26-37.
77. Кудряшов А.П., Соловьёв И.В. Выделение объектов на топографическом плане для реконструкции сцены городского пространства // Программная инженерия. – 2019. – Т. 10. – № 11-12. – С. 464-470.
78. Кудряшов А.П., Соловьёв И.В. Построение рельефа при реконструкции сцены городского пространства с помощью сервисов Google Maps // В сборнике: GraphiCon 2017. Труды 27-й Международной конференции по компьютерной графике и машинному зрению. Пермский государственный национальный исследовательский университет. – 2017. – С. 312-314.
79. Кудряшов А.П., Соловьев И.В. Построение текстурированной трехмерной модели города с помощью набора фотоснимков и топографического плана // Информатика и системы управления. – 2018. – № 2 (56). – С. 24-34.
80. Кудряшов А.П., Соловьёв И.В. Распознавание дополнительных объектов на топографическом плане при реконструкции городских сцен // В сборнике: ГРАФИКОН'2015. Труды Юбилейной 25-й Международной научной конференции. – 2015. – С. 107-111.

81. Кудряшов А.П., Соловьёв И.В. Распознавание контуров зданий на топографическом плане для реконструкции городских сцен // Вестник компьютерных и информационных технологий. – 2015. – № 2 (128). – С. 3-8.
82. Кудряшов А.П., Соловьёв И.В. Реконструкция городской обстановки с учетом рельефа местности с использованием топографического плана и сервисов Google Maps // Информационные технологии. – 2017. – Т. 23. – № 5. – С. 382-387.
83. Кудряшов А.П., Соловьёв И.В. Реконструкция текстурированной модели городского пространства по топографическому плану и фотоснимкам // Программные продукты и системы. – 2019. – № 3. – С. 496-503.
84. Кудряшов А.П., Соловьёв И.В. Реконструкция трехмерной модели городского пространства на основе топографического плана // Вестник Амурского государственного университета. – 2016. – № 73. – С. 58-66.
85. Кудряшов А.П., Соловьёв И.В. Создание единой текстурированной трехмерной модели по набору видов // Программные продукты, системы и алгоритмы. – 2017. – № 1. – С. 3-7.
86. Кудряшов А.П., Соловьёв И.В. Текстурирование трехмерной модели, построенной по объединенному набору видов с зашивкой дыр // Программные продукты и системы. – 2017. – № 2. – С. 286-290.
87. Лопатин М.Л., Балабаев С.Л. Комплекс реконструкции трехмерных моделей объектов // Известия высших учебных заведений. Северо-Кавказский регион. Технические науки. – 2011. – №1. – С. 9-2.
88. Михелев Ю.Д, Лобанов А.А. О точности топографической съемки в масштабе 1: 200 // Известия высших учебных заведений «Геодезия и аэрофотосъемка». – 2005. – №1. – С. 21-28.
89. Оспанов Б.С. Методика оценки интегральной сложности топографического массива // Интерэкспо Гео-Сибирь. – 2013. – №2. – С. 153-156.
90. Пытьев Ю.П., Чуличков А.И. Методы морфологического анализа изображений. – М.: Физматлит. – 2010.

91. Свидетельство о государственной регистрации программы для ЭВМ №2016660490. Распознавание и выделение контуров зданий на топографическом плане / Соловьёв И.В., Кудряшов А.П. – №2016660490. Дата поступления 20.07.2016 г. Зарегистрировано в Реестре программ для ЭВМ 16.09.2016 г.
92. Свидетельство о государственной регистрации программы для ЭВМ №2017660994. Трехмерная реконструкция городской обстановки по топографическому плану / Кудряшов А.П., Соловьёв И.В. – №2017660994. Дата поступления 03.08.2017 г. Зарегистрировано в Реестре программ для ЭВМ 02.10.2017 г.
93. Середович А.В., Дементьева О.А., Горохова Е.И. Трехмерное моделирование участков городских территорий для проектирования инженерных объектов // Интерэкспо Гео-Сибирь. – 2010. – № 3. – С. 64-68.
94. Скрипников В.А., Скрипников А.В. К вопросу о повышении информативности и точности цифровых топографических планов масштаба 1: 5000 // Интерэкспо Гео-Сибирь. – 2006. – №-1. – С. 98-100.
95. Соловьёв И.В. Выделение контуров зданий и распознавание служебных символов для трехмерной реконструкции объектов городской обстановки по топографическому плану // В сборнике: Графикон'2013. Труды конференции. Институт автоматизации и процессов управления ДВО РАН, Дальневосточный федеральный университет. – 2013. – С. 290-293.
96. Тужилкин А.Ю. Распознавание и реконструкция 3D-объектов по спутниковым изображениям на основе сравнения спектров графов // Фундаментальные исследования. – 2015. – № 2 (часть 17) – С. 3727-3732.
97. Условные знаки для топографических планов масштабов 1:5000, 1:2000, 1:1000, 1:500 // Главное управление геодезии и картографии при Совете Министров СССР. – М.: Недра. – 1989. – С. 286
98. Форсайт Д., Понс Ж. Компьютерное зрение. Современный подход.: Пер. с англ. – М.: Издательский дом «Вильямс». – 2004. – С. 928.

99. Хрящёв Д.А. Об одном методе выделения контуров на цифровых изображениях // Вестник Астраханского государственного технического университета. Серия: Управление, вычислительная техника и информатика. – 2010. – № 2. – С. 181-187.
100. Цветков В.Я. Методы и системы обработки и представления видеоинформации // Всероссийский научно-технический информационный центр (ВНТИЦ). – 1991. – 113 С.
101. Чермошенцев А.Ю. Обновление топографических планов масштаба 1 : 5 000 с использованием космических снимков сверхвысокого разрешения // Интерэкспо Гео-Сибирь. – 2012. – № 4. – С. 28-32.
102. Чернов А.В., Чупшев Н.В. Автоматическое распознавание контуров зданий на картографических изображениях // Компьютерная оптика. – 2007. – № 31. – С. 101-103.
103. Шальнов Е.В., Конушин А.С. Использование геометрии сцены для увеличения точности детекторов // Программные продукты и системы. – 2017. – Т. 30. – № 1. – С.106–111.
104. Якубенко А.А., Мизин И.С., Конушин А.С. Поиск регулярных решеток на текстуре фасадов зданий // Программные продукты и системы. – 2010. – № 4. – С. 162–167.

Приложение А. Свидетельство о государственной регистрации программы
для ЭВМ

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2016660490

РАСПОЗНАВАНИЕ И ВЫДЕЛЕНИЕ КОНТУРОВ ЗДАНИЙ
НА ТОПОГРАФИЧЕСКОМ ПЛАНЕ

Правообладатель: *Федеральное государственное бюджетное учреждение науки Институт автоматики и процессов управления Дальневосточного отделения Российской Академии наук (ИАПУ ДВО РАН) (RU)*

Авторы: *Соловьёв Игорь Владимирович (RU),
Кудряшов Алексей Павлович (RU)*

Заявка № 2016617962

Дата поступления 20 июля 2016 г.

Дата государственной регистрации

в Реестре программ для ЭВМ 16 сентября 2016 г.



Руководитель Федеральной службы
по интеллектуальной собственности

Г.П. Ивлиев Г.П. Ивлиев

Приложение Б. Акт о внедрении

УТВЕРЖДАЮ
 генеральный директор
 Лапин В.С.
 «07» февраля 2019 г.

АКТ

о внедрении материалов
 диссертационного исследования в деятельность хозяйственного партнёрства
 специализированной проектной компании "Восток"

Настоящим подтверждаем, что научные результаты диссертационной работы Соловьёва И.В. «Реконструкция трехмерной модели городской обстановки по топографическому плану», представляемой на соискание учёной степени кандидата технических наук, обладают высокой актуальностью и представляют практический интерес. Результаты кандидатской диссертационной работы были применены на практике, а именно:

1. Разработанное программное обеспечение позволило реконструировать трехмерную модель протяженной сцены городской обстановки г. Владивостока. Общая площадь территории составила примерно 6 км². В качестве источника был использован топографический план масштаба 1:2000.
2. Результаты реконструкций применялись в качестве наглядной демонстрации предлагаемых изменений районов г. Владивостока на встречах с руководством города и Приморского края.
3. Предложенная технология построения объемных текстурированных полигональных моделей объектов городской обстановки по топографическому плану и набору фотоснимков сократило время и ресурсы, затрачиваемые на реконструкцию сцен различных масштабов.

Генеральный директор
 Главный геодезист



Лапин В.С.
 Цимбаленко А.Б.

Приложение В. Описание разработанного программного комплекса

Разработанное приложение предназначено для создания трехмерных сцен городской обстановки по топоплану. В качестве исходных данных программа получает изображение топоплана. В главном окне программы (Рисунок В.1) происходит редактирование сцены. На сцену могут быть добавлены любые объекты топоплана. Здания могут быть сформированы вручную.



Рисунок В.1 Главное окно системы

Распознавание. Приложение позволяет автоматически выделить здания на топоплане. Чтобы воспользоваться данной возможностью нужно выбрать «Здания – Поиск...» (Рисунок В.2).

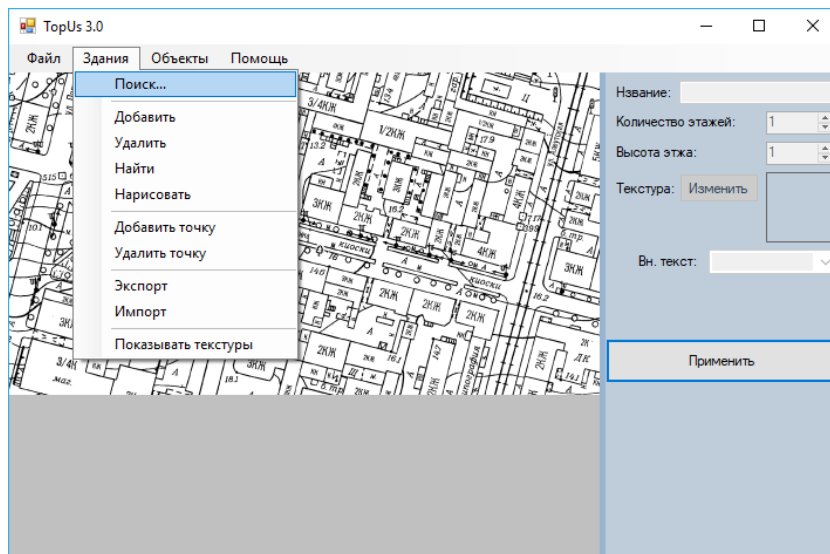


Рисунок В.2 Выбор автоматического распознавания

Будет открыто окно с настройками автоматического поиска (Рисунок В.3).

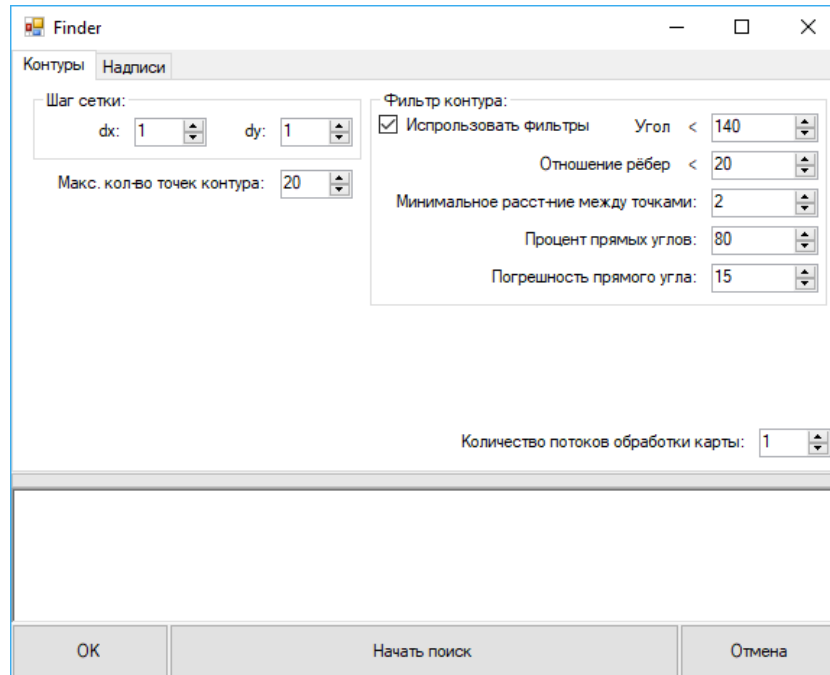


Рисунок В.3 Окно поиска

Настройки поиска:

- *Шаг сетки.* Через сколько пикселей по горизонтали и вертикали необходимо запускать поиск здания.
- *Максимальное количество точек контура.* Контур с количеством точек больше этого количества не будут считаться зданиями.
- *Использовать фильтры.* Если флаг установлен, то контур будет оптимизирован, т.е. количество точек, из которого он состоит, будет уменьшено.
- *Угол.* Если три точки контура будут образовывать угол больше указанного, то серединная точка будет удалена из контура.
- *Отношение ребер.* Если три точки контура будут образовывать такое положение, при котором отношение одного отрезка к другому будет больше указанного числа, то серединная точка будет удалена.
- *Минимальное расстояние между точками.* Точки, расстояние между которыми меньше указанного числа, будут объединены в одну.

- *Процент прямых углов.* Если в контуре количество прямых углов составляет процент менее указанного, то этот контур не будет определяться как здание.
- *Погрешность прямого угла.* Прямым углом в контуре будет считаться угол в диапазоне $90 \pm$ указанное значение в градусах.
- *Количество потоков обработки карты.* Количество потоков, которое используется для подготовки карты к распознаванию.

Для начала поиска нужно нажать «Начать поиск».

После выполнения будет показана информация о времени выполнении поиска (Рисунок В.4):

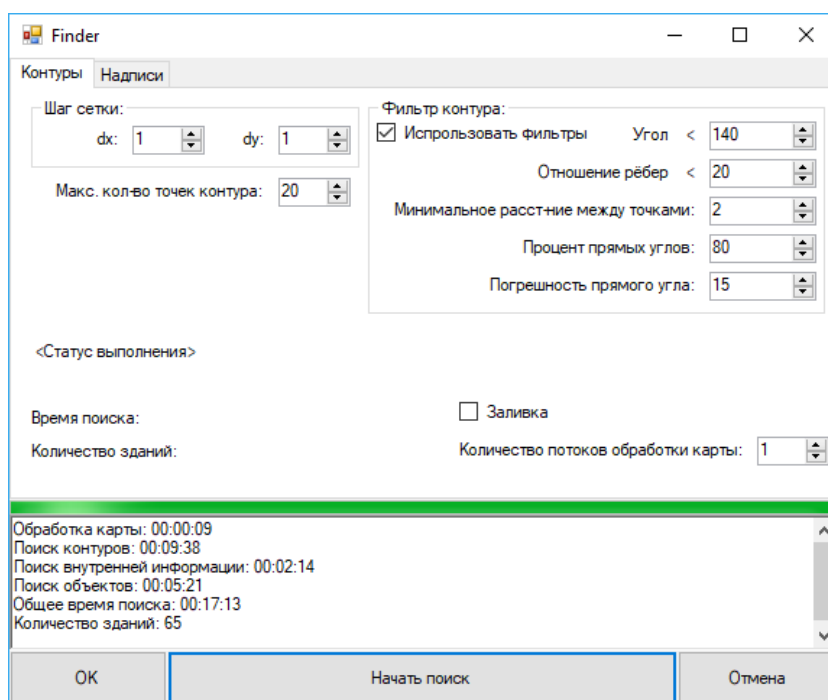


Рисунок В.4 Результат поиска

После нажатия «Ок» будет показан результат распознавания информации в главном окне.

Редактирование. В приложении предусмотрена возможность добавления, удаления и изменения объектов и зданий.

Добавление и удаление зданий осуществляется через меню «Здания» (Рисунок В.5).

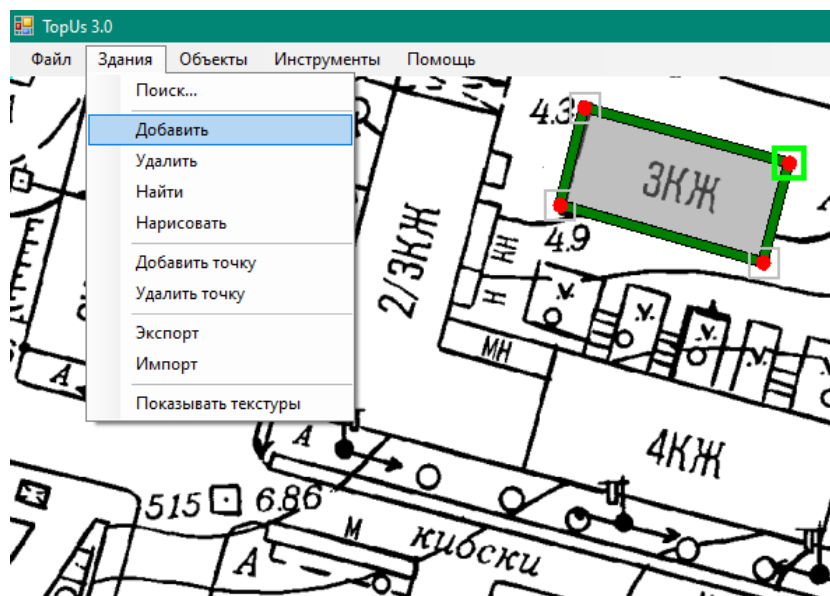


Рисунок В.5 Редактирование сцены

Добавление и удаление объектов осуществляется через меню «Объекты».

Текстурирование. Приложение позволяет накладывать на здания текстуры с фотографий. Для этого нужно в главном меню выбрать «Файл – Геонастройки...». Будет открыто окно геонастроек топоплана (Рисунок В.6):



Рисунок В.6 Модуль текстурирования по геоданным

На вкладке «Геоточки» необходимо указать две опорные геоточки. По ним будут рассчитываться геокоординаты всех объектов.

Для добавления геоточки нужно нажать кнопку «Добавить». В центре карты появится точка. Её нужно передвинуть на место, с которого делался снимок здания и указать направление съемки. Далее нужно указать снимок, который был сделан с этой точки нажав кнопку «Фото».

В окне с фотоснимком нужно указать точки текстуры (Рисунок В.7).

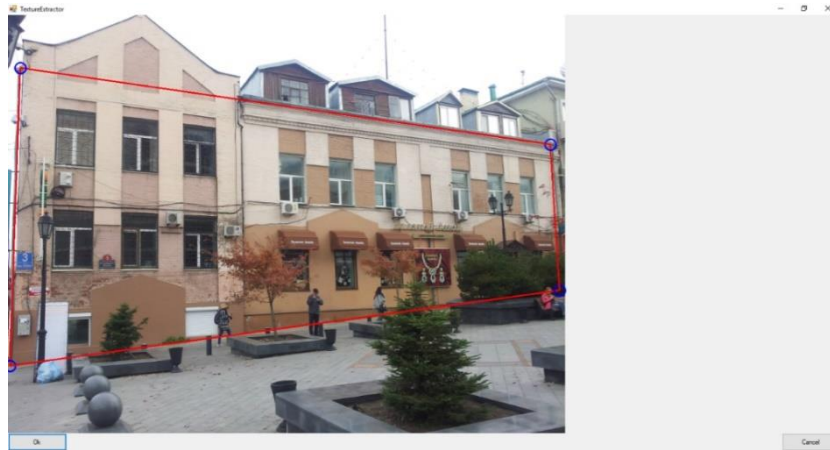


Рисунок В.7 Окно коррекции перспективы

Для экспорта или импорта данных о геоточках и их текстурах можно воспользоваться соответствующей кнопкой.

Построение рельефа. На основании ранее указанных геоточках приложение позволяет построить рельеф сцены на основании обращения к сервису Google Maps Elevation API. Для этого используется вкладка «Ландшафт» (Рисунок В.8).

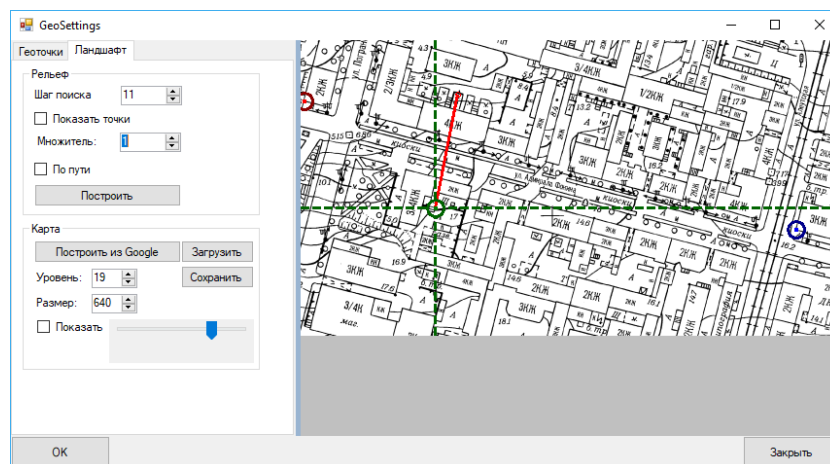


Рисунок В.8 Окно построения рельефа

Рельеф является триангуляционной поверхностью. Количество точек в нем по вертикали и горизонтали одинаково и равно значению в поле «Шаг поиска».

Точки рельефа могут быть построены двумя способами:

По точкам топоплана, геокоординаты которых рассчитываются на основании опорных точек. В этом случае количество запросов к сервису будут зависеть от общего количества точек рельефа.

Между двумя опорными точками проводится прямая линия. И на ней сервис находит высоты. В этом случае количество запросов к сервису будет равно значению, указанному в поле «Шаг поиска». Поле «Множитель» увеличивает ту высоту, которую вернул сервис в указанное число раз для того, что на небольших топопланах был виден перепад высот. Для построения рельефа нужно нажать кнопку «Построить». Для получения текстуры рельефа на основании данных с сервиса Google Maps API нужно нажать кнопку «Построить из Google». В поле «Уровень» нужно указать то, как близко производится съемка со спутника. В поле «Размер» нужно указать размер квадратного изображения, получаемого из сервиса. Текстуру рельефа можно также загрузить и сохранить.

Построение сцены. Программа имеет возможность экспортировать все объекты в файл трехмерной графики «.x». Для этого нужно в главном меню выбрать «Файл – Построить сцену...». Откроется окно построения сцены (Рисунок В.9).

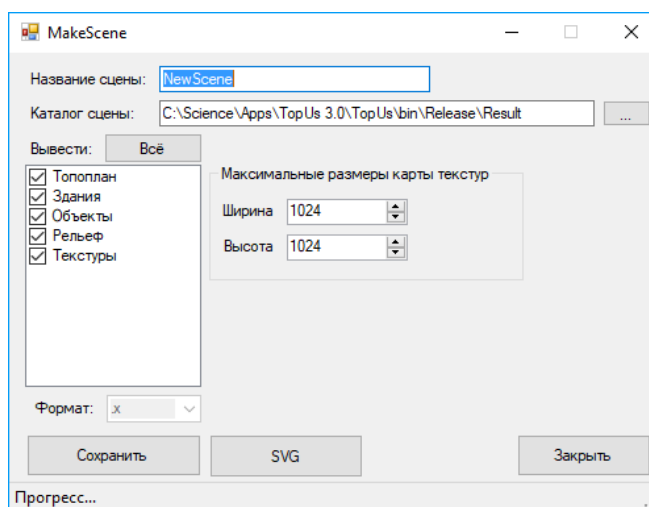


Рисунок В.9 Окно экспорта сцены

В данном окне требуется указать имя сцены и каталог, в который она будет экспортироваться. В каталоге экспорта будет находиться сам X-файл и каталог с текстурами. В окне есть возможность экспортировать объекты выборочно. Для этого в списке слева нужно выбрать то, что должно быть в сцене. Для построения сцены следует нажать кнопку «Сохранить». Для экспорта контуров зданий в SVG-файл необходимо нажать кнопку «SVG»