



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего образования

«Дальневосточный федеральный университет»
(ДВФУ)

Институт математики и компьютерных технологий
Департамент программной инженерии и искусственного интеллекта

ФЕНЬ ЕВГЕНИЯ ВАДИМОВНА

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

бакалаврская работа

вид ВКР

**ПОСТРОЕНИЕ ТРЕХМЕРНОЙ МОДЕЛИ ПОДВОДНОГО
ТРУБОПРОВОДА ПРИ ПОМОЩИ ПОДВОДНОГО АВТОНОМНОГО
АППАРАТА**

по направлению подготовки (специальности) 09.03.04 Программная инженерия
профиль Программная инженерия

Владивосток
2024

Оглавление

Введение.....	3
1 Обзор существующих подходов решения задачи.....	5
1.1 Глоссарий терминов.....	5
1.2 Программные средства для трехмерной реконструкции.....	6
1.3 Методы детектирования и трехмерной реконструкции.....	7
1.4 Краткий вывод.....	20
2 Анализ предметной области и построение модели.....	22
2.1 Анализ множества задач предметной области «Трехмерная реконструкция подводного трубопровода по стереопаре».....	22
2.2 Информационные объекты.....	22
2.3 Задачи решаемые в процессе построения трехмерной модели.....	25
2.4 Задачи обработки, хранения и передачи информации.....	25
2.5 Анализ смысла ситуации предметной области «Трехмерная реконструкция подводного трубопровода по стереопаре».....	26
2.6 Анализ знаний предметной области «Трехмерная реконструкция подводного трубопровода по стереопаре».....	27
2.7 Модель преобразований.....	28
2.8 Краткий вывод.....	41
3 Разработка проекта программной системы.....	41
3.1 Диаграмма вариантов использования.....	43
3.2 Архитектурно-контекстная диаграмма 1-го уровня.....	43
3.3 Диаграмма последовательности состояний.....	46
3.4 Диаграмма перехода из состояния в состояние.....	46
3.5 Диаграмма потоков данных.....	47
3.6 Описание функциональных требований.....	48
3.7 Требования к входным данным.....	49
3.8 Требования к выходным данным.....	50
3.9 План тестирования.....	51
3.10 Краткий вывод.....	62
4 Реализация и тестирование.....	63
4.1 Язык программирования.....	63
4.2 Среда разработки.....	63
4.3 Вспомогательные ПО.....	63
4.4 Тестирование.....	64
4.5 Результаты экспериментов.....	65
Заключение.....	79
Список литературы.....	81

Введение

В современном мире большое место уделяется транспортировке ресурсов, для их перемещения выгоднее всего использовать трубопроводы. Данный вид транспорта сильно сокращает расходы на перевозки и уменьшает время транспортировки добываемых ресурсов. Трубопроводу не требуются дороги, что позволяет проложить его напрямик, даже под водой.

Однако несмотря на свою эффективность даже трубопроводу требуется техническое обслуживание. Нередко в нем появляются дыры и ржавчина. Это может повлечь за собой потерю ресурсов и даже загрязнение окружающей среды. Особенно сильно может пострадать водоем, на дне которого проложен трубопровод. Поэтому очень важно проводить инспекцию подводных трубопроводов.

Поскольку трубопроводы находятся на большой глубине, инспекции опасно проводить с использованием водолазов. Даже используя специальное снаряжение, давление на организм будет очень велико. Это может привести к повреждению легких и барабанных перепонок.

Поэтому для инспекции подводных трубопроводов используют автономный необитаемый подводный аппарат (АНПА), который оснащен программным средством, позволяющим фиксировать нарушения конструкции.

Объектом исследования, являются программные системы, которые могут использовать компьютерное зрение для построения трехмерной модели статической сцены, содержащей объекты, по видеопотоку, который разбивается на изображения.

Предметом исследования, являются алгоритмы по детектированию объектов статической сцены, а также алгоритмы позволяющие построить трехмерную модель объекта статической сцены по ряду изображений.

Задача по трехмерной реконструкции подводного трубопровода с использованием АНПА и стереокамеры **актуальна**, так как такой подход

позволяет точно определить места протечек, либо коррозии и затратить на это минимальное количество времени и денег, поскольку АНПА маневренный и может нести все необходимое оборудование на себе.

Практическая значимость заключается в облегчении АНПА путем уменьшения числа оборудования, поскольку стереокамера способна зафиксировать все необходимые данные, не прибегая к помощи стороннего оборудования. Это позволяет уменьшить нагрузку на процессор АНПА, что позволит расширить спектр задач выполняемых АНПА.

Целью выпускной квалификационной работы является разработка моделей и методов программной системы по созданию трехмерной реконструкции подводного трубопровода при помощи съемки со стереокамеры, находящейся на АНПА.

Задачи выпускной квалификационной работы:

1. Анализ предметной области «Трехмерная реконструкция подводного трубопровода по стереопаре».
2. Разработка модели предметной области «Трехмерная реконструкция подводного трубопровода по стереопаре».
3. Разработка проекта программной системы для трехмерной реконструкции с использованием стереокамеры.
4. Разработка модулей программной системы для трехмерной реконструкции с использованием стереокамеры.

1 Обзор существующих подходов решения задачи

В данной главе будут рассмотрены существующие программные системы и методы для решения задачи трехмерной реконструкции сцены реального мира.

1.1 Глоссарий терминов

АНПА – автономный необитаемый подводный аппарат. Обычно имеет форму торпеды. Перемещается под водой для сбора информации [1].

GSP-точки – особенные точки, которые хорошо различимы на изображении [33].

ТоF-камера – это камера, которая работает на основе сигналов. Она посылает сигнал в заданном направлении и принимает его обратно. По принятому сигналу строится изображение [46].

Стереокамера – это специальная камера с двумя или более объективами, расположенных на определенном расстоянии друг от друга, что позволяет камере имитировать зрение человека [30].

Ректифицированное изображение – это спроецированные два или более изображения на общую плоскость изображения [27].

Стереозрение – это вид зрения, при котором возможно восприятие формы, размера и расстояния [31].

Стереои изображения – это полученные изображения со стереокамеры [29].

Облако точек – это набор вершин в трехмерной системе координат [33].

Карта глубины – это изображение в градациях серого цвета [17].

Тор (Тороид) – это поверхность в форме бублика, полученная вращением образующей окружности вокруг оси, лежащей в плоскости этой окружности и не пересекающей ее [32].

1.2 Программные средства для трехмерной реконструкции

На основании выше сказанного можно выделить компоненты программной системы: создания трехмерной модели, загрузка изображения (потока изображений). Далее рассматриваются примеры программных средств, с реализованными компонентами.

На данный момент уже существуют программные средства позволяющие создать трехмерную модель по изображениям, однако алгоритмы, используемые в программных средствах, в большинстве случаев недоступны для свободного использования. Конечно, есть программные средства с открытым программным кодом, однако их нельзя взять и использовать. Их необходимо сначала доработать.

Существуют такие программные средства, как:

– Meshroom.

Программа, которая написана на Python с открытым исходным кодом и позволяющая по группе фотографий получить 3D модель. В процессе работы она анализирует все входящие изображения и выясняет какие изображения имеет смысл сопоставлять друг с другом. Далее ищутся соответствия на изображениях, избавляется от искажений и создает карту глубины, для последующего создания облака точек. После фильтрации облака точек создается трехмерная модель на основе треугольников. Для более точного отображения модели треугольники также фильтруются и накладываются текстуры.

Программе недостаточно получить только группу изображений, ей также необходимы характеристики камеры для сопоставления фотографий [26].

Достоинства: открытый программный код.

Недостатки: трехмерная модель не точная.

– TripoSR.

Программное средство использует нейросеть для трехмерной реконструкции. Оно получает на вход поток изображений, после чего изображения по одному прогоняются через предварительно обученный кодировщик, который преобразует каждое в векторы, содержащие особенности изображения. Эти векторы используются декодером для создания 3D модели. Данная программа не требует дополнительных входных данных [15].

Достоинства: высокая скорость выполнения.

Недостатки: нейросеть нельзя получить для личного использования.

– 3DPEA.

Программное средство предлагающая по изображению создать трехмерную модель. Достаточно только загрузить изображение и выставить настройки, для создания модели. Модель можно скачать. Также есть возможность посмотреть уже созданные модели [24].

Недостатки: нельзя получить код, модель получается неточная.

Подводя итоги всего выше сказанного, можно сделать вывод, что в каждой программной системе есть недостатки и в большинстве случаев отсутствует возможность использования кода в личных целях, а также получаемая трехмерная модель не точная.

1.3 Методы детектирования и трехмерной реконструкции

В следующих главах рассматриваются методы позволяющие распознать объект на изображении и создать трехмерную модель объекта.

1.3.1 Методы детектирования объектов

В данном разделе представлены методы, которые позволяют идентифицировать объект на реальной сцене.

1.3.1.1 Метод обнаружения подводного объекта с заданными характеристиками при помощи стереокамеры

Данный метод обнаруживает подводные объекты с заданными характеристиками (цвет, размер, геометрическая форма) по поступающим в реальном времени стереоснимкам от подводной системы технического стереозрения. Также метод определяет и уточняет недостающие параметры реального объекта, его координаты и дальность до него в системе отсчёта системы технического стереозрения.

Входным параметром метода является стереопара изображений, полученных со стереокамеры. Главной особенностью является то, что стереокамера наблюдает не всю поверхность объекта, искажен цвет, неправильно воспринимается расстояние до объекта, не всегда видны контуры, не ровная поверхность около объекта. Калибровка камер под водой, применение предобработки изображений, поиск объекта по цвету с учётом искажений, построение 3D облака точек подводной сцены, фильтрация шума позволяют решить поставленную задачу.

Выходными параметрами являются трёхмерные координаты центра объекта, дальность от системы технического стереозрения до объекта, недостающие характеристики объекта, сжатое изображение найденного объекта [6].

1.3.1.2 Метод детектирования с использованием лазера и стереокамеры

Метод визуальной навигации, включает в себя использование лазера и стереокамеры (рисунок 1). Где лазер и стереокамера стоят так, чтобы луч лазера проходил вдоль строки изображения.

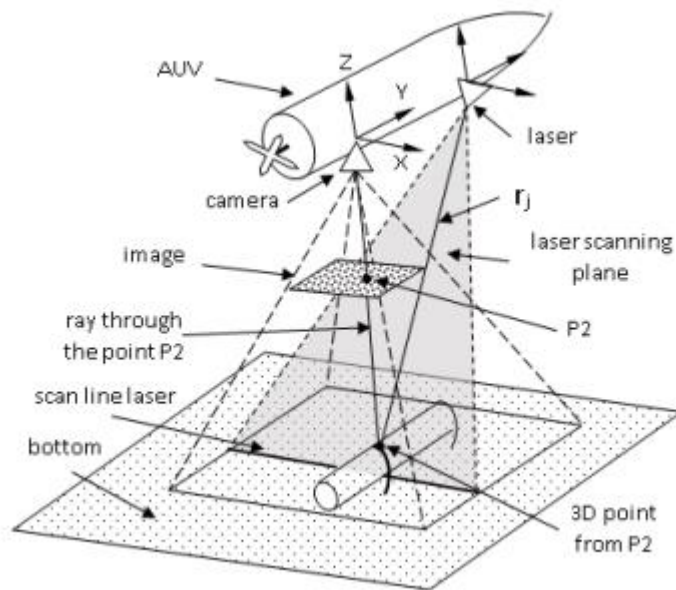


Рисунок 1– Пример сканирования подводного дна, используя линейный лазер и синхронную съемку с камер

Нужно учитывать, что строки изображений параллельны, т.е. два изображения со стереопары выровнены. Такой способ позволяет упростить получение 3D координат точек лазерного профиля за счет известной геометрии взаимного расположения камеры и лазера.

Обработка очередного (текущего) фрагмента траектории включает в себя определение линии лазера на снимке стереопары. Также вычисление трехмерных координат пространственных точек сцены, где применяется триангуляция. После чего анализируется функция профиля с вычислением локальных экстремумов. Далее три предыдущие точки, каждая из которых определена в локальной области видимости одной камеры соответствующей позиции, помещаем в области видимости стереокамеры, применяя предварительно полученные матрицы геометрических преобразований. И по итогу по трем точкам в области видимости стереокамеры строятся два звена ломаной линии, которые определяют текущий угол поворота. Где ищется среди множества трехмерных точек экстремумам в области видимости стереокамеры точку, наилучшим образом отвечающую указанному выше критерию сохранения текущей кривизны [6].

Метод позволяет надёжно управлять движением автономного подводного аппарата вдоль подводного трубопровода

1.3.1.3 Методы отслеживания подводного трубопровода устройствами для обнаружения металлосодержащего объекта

Следующий метод использует автономный подводный необитаемый аппарат (АНПА), который несет на себе устройство для обнаружения и отслеживания. Устройство состоит из двух излучателей электромагнитного поля. Излучатели находятся на носовой и кормовой части АНПА, где одно принимает электромагнитное поле, а второе его излучает. Два электрода повернуты в разные стороны относительно продольной вертикальной осевой плоскости АНПА на равный угол. Электрические поля приемных электродов повернуты в разные стороны относительно поперечной вертикальной осевой плоскости АНПА. Также устройство содержит управляемый аттенюатор, измерительный преобразователь тока, а еще управляемый блок нормализатора сигналов. Соединенные управляемый аттенюатор и измерительный преобразователь тока включены в разрыв связи между выходом генератора переменного напряжения и входом первого коммутатора. Входы блока нормализатора сигналов соединены с приемными электродами, а его выходы через второй коммутатор – с первым входом блока управления и преобразования сигналов [9].

Помимо подводной магнитной съемки существует еще и метод с надводной магнитной съемкой. Такой метод предполагает использование беспилотного летательного аппарата (БПЛА) и магнитной антенны (рисунок 2). Для выполнения магнитометрической съемки пространственного положения трубопровода необходимо перемещать БПЛА над поверхностью воды так, чтобы трубопровод находился прямо под ним. Траектория полета задается GPS координатами, которые предварительно определяются путем криволинейного перемещения магнитной антенны с использованием БПЛА над трубопроводом с шагом в 100 метров по максимальному значению

индукции магнитного поля, получаемому в точке пересечения магнитной антенны с трубопроводом [10]. Однако данный метод неэффективен, если трубопровод находится на большой глубине.

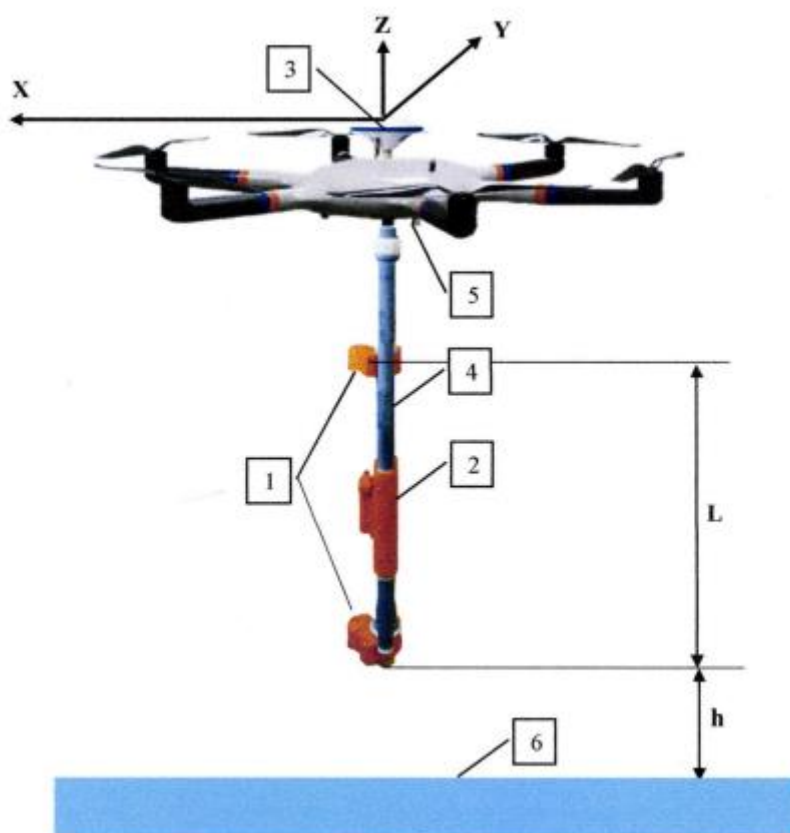


Рисунок 2 – Беспилотный летательный аппарат с магнитной антенной

Помимо магнитной съемки еще можно использовать гидролокатор бокового обзора (ГБО), который прикреплен к АНПА. Его особенностью является излучение и прием зондирующих звуковых посылок.

На итоговом изображении объект выделяется за счет градиента яркости (он вычисляется в центральной точке окна 3x3 пикселя) и на основании того, что АНПА движется прямо без существенного изменения курса, можно рассчитать продольную составляющую вектора градиента [21].

Для определения трубопровода нужно знать информацию о пространственном расположении точек изображения в горизонтальной плоскости. Это можно сделать, если спроецировать гидрологический снимок на горизонтальную плоскость с привязкой точек изображения к географическим координатам. После определяются географических

координат пикселей изображения надо использовать поперечное смещения точек и местоположение АНПА и по данным системы счисления пути, курс аппарата и продольное местоположение антенн ГБО на борту вычисляются координаты. Трубопровод на эхограмме выглядит как прямолинейный объект, в проекции на горизонтальную плоскость.

Для вырисовывания линий трубопровода используется модифицированное преобразование Хафа [2].

1.3.2 Методы трехмерной реконструкции

В данном разделе будут рассматриваться методы, которые описывают поэтапно трехмерную реконструкцию объекта.

1.3.2.1 Метод трехмерной реконструкции объекта на основе стереопары

Метод трехмерной реконструкции с использованием только стереокамеры включает в себя несколько этапов:

1. Поиск соответствующих точек на стереопаре.

Необходимо, с помощью динамического метода, найти оптимальный путь на двухмерной матрице, используя три типа движения (по горизонтали, вертикали и диагонали). За каждый тип движения начисляется определенный штраф. Путь должен пролегать через базовые GCP-точки, которые известны заранее и определены достаточно точно. Также могут быть многозадачные GCP-точки[33].

2. Вычисление трехмерных координат для точек объектов.

Рассматривая этот процесс в общем случае, можно выделить три основных этапа: вычисление фундаментальной матрицы, матриц камер (по фундаментальной матрице) и нахождение для каждой пары сопряженных точек точки в пространстве[33].

3. Построение трехмерной модели сцены.

Для построения трёхмерной модели предлагается использовать гибридный алгоритм построения триангуляции Дилоне. Однако в таком случае нужно учитывать, что строки изображения должны быть параллельны стереобазе, эпиполярные линии совпадают со строками изображения и соответствующие точки лежат на одной строке изображения [33].

Но для использования этого метода стереопара должна быть высокого качества, чтобы не было помех на изображениях.

1.3.2.2 Трёхмерная реконструкция по нескольким изображениям

Данный метод трёхмерной реконструкции также основан на использовании стереопары, однако в нем приведены примеры детекторов и методов для создания трёхмерной модели.

Метод основан на нескольких этапах:

1. Получение стереоизображения сцены
2. Нахождение характерных точек
3. Вычисление трёхмерных координат точек изображения
4. Построение трёхмерной модели по трёхмерным точкам

Как правило, при использовании такого метода, положение камер и фокусные расстояния заранее неизвестны и находятся в ходе триангуляции [18].

На первом этапе получают исходные данные (стереопару).

Далее на втором этапе необходимо найти характерные точки двух изображений. Это точки, каждая из которых соответствует одной и той же точке объекта на двух изображениях. Для нахождения таких точек уже существуют некоторые детекторы: Харриса [40], SIFT [35], SURF [23].

На третьем этапе используются методы позволяющие построить трёхмерные координаты, учитывая и не учитывая калибровку камер.

На последнем этапе можно использовать один из методов трёхмерной реконструкции:

1. Триангуляция Дилоне.

2. Метод по фокусировке и дефокусировке.
3. Метод основанный на изменении яркости.
4. Методы по движению

Для определения соответствующих точек может использоваться метод кросс-корреляции [11] или метод основанный на сопоставлении характерных признаков на двух изображениях. Признаками могут выступать стыки, отрезки или области.

1.3.2.3 Метод трехмерной реконструкции с использованием калибровочной сетки

Метод трехмерной реконструкции дна, где главной особенностью является калибровка камеры путем размещения калибровочной сетки на дне водоема.

Первым делом производится калибровка камер путем размещения калибровочной сетки на дне океана и записи видео с разных ракурсов. Камеры калибруют непосредственно под водой и независимо друг от друга.

Данный метод принимает на вход пару изображений с откалиброванной стереокамеры.

В процессе работы из изображений извлекаются существующие признаки и для каждой стереопары устанавливаются соответствия. Также изображения ректифицируются, что позволяет ограничить поиск соответствий, поскольку калибровки может быть не идеальной. После чего находятся трехмерные координаты каждой точки, для которой было найдено соответствие, вычисляется с использованием матрицы перепроецирования и сохраняются для последующего использования. Для создания итоговой трехмерной модели используется триангуляция Дилоне и фильтрация точек, а также наложение текстур.

На всех изображениях отслеживаются особенности, чтобы восстановить структуру среды. Также учитывается модель движения в пространстве, для оценки положения потенциальных совпадений

особенностей. Это позволяет ускорить поиск соответствий и снизить количество ошибочно найденных совпадений. Благодаря этому модель подводного дна становится более точной [48].

1.3.2.4 Метод трехмерной реконструкции по заданному облаку точек

Следующий метод позволяет по уже заданному облаку точек сформировать трехмерную модель. Облако точек представляет собой массив трехмерных координат точек. С точки зрения трехмерной фигуры, облако точек – это набор вершин, предназначенные для представления внешней поверхности фигуры. Пример визуального представления облака точек можно увидеть на рисунке 3.

Как правило облако точек находится на основе карты глубины. Которая в свою очередь показывает расстояние до объекта относительно камеры. Однако она также содержит погрешности.

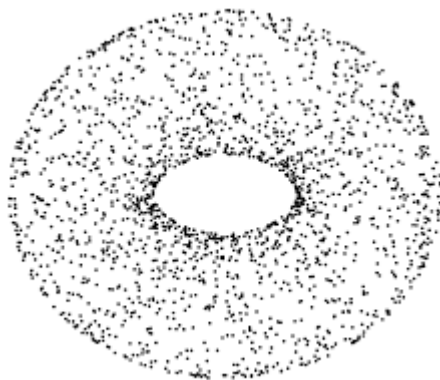


Рисунок 3 – Визуальное представление облака точек в виде тороида

Если говорить о трехмерной реконструкции по облаку точек, то использование неотфильтрованного облака точек может привести к искажению объекта. Поскольку облако точек может содержать точки не принадлежащие объекту или ошибочные точки с большой погрешностью. Поэтому перед реконструкцией необходимо применять фильтрацию, чтобы уменьшить или вовсе избавиться от таких ошибочных точек. На рисунке 6

видно, что при создании модели на неотфильтрованном облаке точек присутствуют артефакты, не принадлежащие объекту [29].

Погрешности появляются при физических ограничениях камер или множественном отражении объекта или освещении или перекрытии объекта.

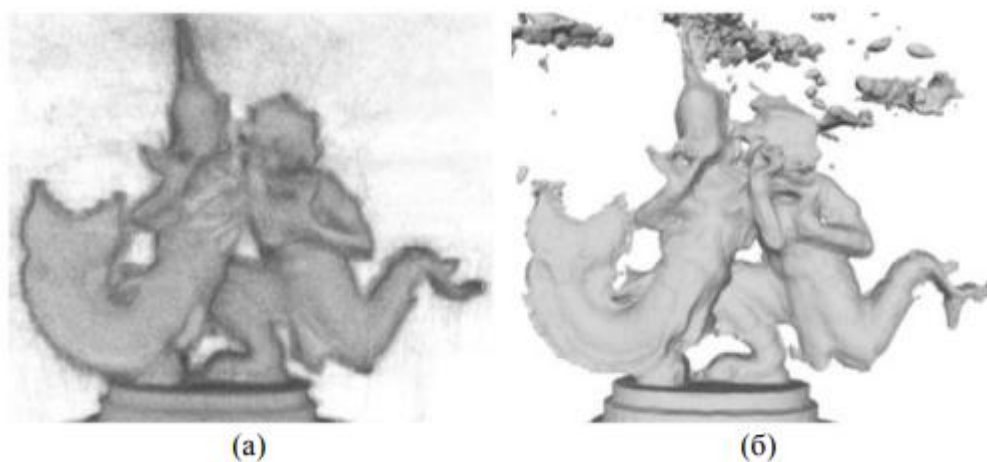


Рисунок 4 – Пример полигональной модели на основе облака точек без применения методов фильтрации: (а) – исходное облако точек; (б) – полигональная модель

1.3.3 Методы, частично решающие задачу трехмерной реконструкции

Далее рассматриваются методы, которые решают только часть задачи трехмерной реконструкции. Такие как сопоставление особенностей изображений, создание карты глубины и выделение контура объекта. Все это подзадачи задачи трехмерной реконструкции.

1.3.3.1 Метод сопоставления особенностей на двух изображениях

Сопоставление особенностей является первым этапом при трехмерной реконструкции. Этап включает в себя поиск особенных точек на стереоизображении.

Особенные точки – это хорошо различимые фрагменты на изображениях. Такие точки имеют особую окрестность (градиент), что позволяет выделить их на фоне остальных точек изображения. С их помощью

можно проанализировать как изображения целиком, так и объекты на них. Также если на изображении присутствуют хорошо отличимые характерные точки, то можно справиться с изменением масштаба, ракурса и перекрытиями сцены или объекта.

Для особенных точек существуют критерии.

- их местоположение на сцене не должно меняться независимо от ракурса и освещения;
- они не должны изменяться от перекрытия;
- их расположение должно быть в малой области;
- их число должно быть в разы меньше общего числа точек;
- каждая особенная точка должна быть уникальна.

Для нахождения особенностей используют детекторы и дескрипторы. Детекторы находят особенности на изображении и обычно они делятся на три категории: детекторы ребер, детекторы углов и детекторы областей. Дескрипторы предназначены для выделения и сравнения признаков особенностей изображения. Однако недостатком является то, что приходится сопоставлять отдельные особенности на двух изображениях, что приводит к ошибкам при нахождении соответствий.

Примеры дескрипторов: SIFT [35], DAISY [39] и BRIEF [37]. Примеры детекторов: преобразования Хафа (детектор ребер) [2], Харрис (детектор углов) [17] и SURF (детектор области) [23].

Поскольку существуют не только простые особенности (к примеру углы), но также и сложные, которые состоят из целых регионов изображения, универсального способа нахождения особенностей не существует. Поэтому для каждого типа особенностей был создан свой детектор.

1.3.4 Методы построения карты глубины

После сопоставления особенностей на стереопаре нужно получить карту глубины (карту высоты). Она представляет из себя техническое изображение, в котором, для каждого пикселя вместо цвета хранится информация в виде расстояния до камеры (Рисунок 5). Карта высоты, это по факту разновидность карты глубины. Она применяется в случае, если нужно создать рельеф местности для различных информационных систем.

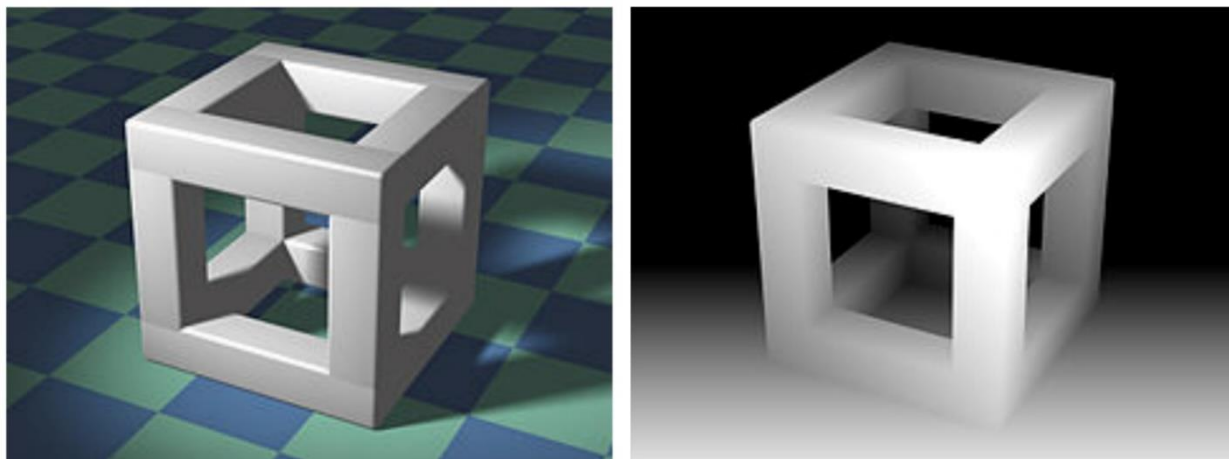


Рисунок 5 – Карта глубины на примере куба

Обычно карта глубины – это изображение в градациях серого цвета. Основным параметром пикселя является его яркость. Яркость определяется значением цветности пикселя от 0 (Полностью черный) до 1 (Полностью белый), но также возможно и от 0 (Полностью черный) до 255 (Полностью белый) [3-5, 7, 8, 14, 19]

Для построения карты глубины существует несколько разных методов. К примеру, метод, основанный на использовании специальной камеры, установленной на АНПА. Для этого может использоваться ToF-камера или специальные структурированные световые камеры. ToF-камера работает по принципу измерения значения задержки света, с которой свет возвращается в каждую точку [41, 42, 46]. Для более эффективного использования ToF-камер можно взять несколько камер с разным временем отраженной световой индикации и, зная сдвиг по времени относительно каждой камеры и снятой яркости вспышки, возможно рассчитать сдвиг и, соответственно, расстояние

до объекта. Нужно также учитывать, что чем больше камер задействовано в применении, тем выше точность метода.

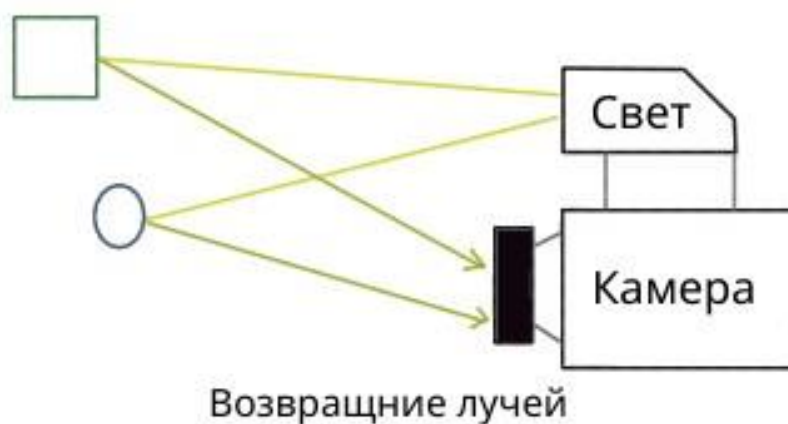


Рисунок 6 – Принцип ToF-камеры

Другой же метод основан на использовании нейронных сетей. Для такой задачи может подойти сверточная нейронная сеть или капсульная.

Сверточные нейронные сети работают на основе стереоизображения и карты смещения. Сначала они обучаются на тестовых данных, а после нейронная сеть уже способна сама предопределять карту глубин по единственному изображению и объектам, расположенным на нём объектам. Однако стоит взять во внимание, что сверточные нейронные сети способны регистрировать только наличие какого-либо объекта на картинке, не кодируя его ориентацию и положение [16].

В отличие от сверточных нейронных сетей капсульные способны регистрировать наличие объектов на изображении, а также их положение в пространстве. Самое главное отличие, это наличие капсул или групп нейронов, которые и позволяют идентифицировать объект. Информация поступает в капсулы виде векторов, которые содержат информацию об ориентации и положении объектов на изображении, а затем передается в капсулы более высокого уровня. Капсулы более высокого уровня принимают информацию из нескольких капсул низкого уровня, а затем выдают прогноз. Капсулы одного уровня не связаны между собой. Капсулы создаются при

помощи разделения выводных данных из сверточного слоя, затем трехмерные векторы разделяются на капсулы таким образом, чтобы в каждой капсуле была информация о каждом пикселе. Состояние нейронов капсульной нейронной сети внутри изображения фиксирует свойство области или объекта внутри изображения: его положение и ориентацию. В отличие от сверточной нейронной сети, капсульная показывает более точные результаты [17].

1.3.5 Метод выделения контура объекта на изображении

Выделение контура изображения позволяет уменьшить количество данных, сохраняя лишь важную информацию о форме объекта в сцене. Контур изображения выделяется за счет перепада яркости в полутоновом изображении. Также это является поиском объекта на изображении. Это позволяет сфокусироваться только на важных объектах сцены.

Контурное изображение представляет из себя набор связанных кривых, которые передают форму объекта.

Существует несколько детекторов, которые способны распознать границы объектов на изображении: «Оператор Кэнни» или Canny Detector[38], также еще есть метод кластеризации.

Бывают моменты, когда необходимо лишь выявить объект на изображении без последующей его классификации. Для этого достаточно найти границы объекта. Эта задача легко решается, если объект находится на однотипном фоне. Для каждого пикселя можно посчитать разницу его цвета с ближайшими пикселями, также можно задать расстояние, в пределах которого будут производиться вычисления. Существует много работ, которые рассматривают поиск контура на изображении [12, 22, 34].

1.4 Краткий вывод

В ходе обзора существующих подходов решения задачи трехмерной реконструкции сцены реального мира, были рассмотрены методы,

необходимые для трехмерной реконструкции, а также были выделены их плюсы и минусы.

2 Анализ предметной области и построение модели

В данной главе выделяются информационные объекты предметной области, строится математическая модель предметной области.

2.1 Анализ множества задач предметной области «Трехмерная реконструкция подводного трубопровода по стереопаре»

Основной задачей рассматриваемой предметной области является детектирование и трехмерная реконструкция подводного трубопровода.

Объектом задачи в предметной области является трехмерная модель подводного трубопровода. Результатом решения задачи предметной области является построение трехмерной модели подводного трубопровода по ряду изображений.

2.2 Информационные объекты

В данном разделе выделяются информационные объекты, которые участвуют при построении модели.

В ряде алгоритмов по трехмерной реконструкции идет работа с отдельными точками изображения.

1. Стереокамера.

Это тип фотоаппарата с двумя объективами, создающими стереоизображение. Расстояние между объективами настраивается в зависимости от расстояния съемки АНПА до объекта. Камеры повернуты не параллельно, благодаря этому возникает фокусное расстояние.

2. Стереоизображение.

Это два изображения сделанных на одну стереокамеру, с небольшим смещением. Каждое изображение по отдельности – это массив пикселей (точек) размером $H \times W$, где значение каждого пикселя определяется яркостью. Где H – это высота изображения, а W – ширина изображения. Минимальный возможный размер изображения 640×640 .

3. Подводный трубопровод.

Металлосодержащий протяженный объект, имеющий цилиндрическую форму и находящийся под водой. На изображении трубопровод выглядит как прямоугольник и имеет характеристики W , L , где W – ширина, L – длина. Минимальный возможный размер трубопровода, на изображении, для его распознавания: $W = 2\text{см}$, $L = 2\text{см}$. Максимальные значения: $W = 90\%$ от размера изображения, $L = \text{размер изображения}$.

Точки трехмерной модели имеют следующие признаки:

1. Координаты для каждой точки.

Для каждой точки модели существует трехмерная система координат, представляющая из себя три вектора указывающие направления осей данной системы координат. Начало системы координат находится в точке пересечения трех векторов. Каждая точка трехмерной системы координат (x , y , z) представляется вещественным числовым значением.

2. Информация о форме модели

Множество значений координат точек, связанных между собой, которые в совокупности представляют контур трехмерной модели объекта.

3. Цвет

Цвет точки задается палитрой RGB. Значение точки имеет три числа, каждое число задается в диапазоне от 0 до 255.

Точки на изображении имеют следующие признаки:

1. Координаты для каждой точки на изображении.

Для каждого изображения существует двухмерная система координат, представляющая из себя два вектора указывающие направление осей данной системы координат. Начало системы координат находится в точке пересечения двух векторов и совпадает с началом системы координат камеры. Каждая точка из декартовой системы координат (x , y) представляется числовым значением.

2. Информация о форме составляющего сцены.

Множество значений координат точек, связанных между собой, которые в совокупности представляют контур объекта, изображенного на снимке.

3. Информация о яркости точки

Яркость точки определяется путем перевода изображения в черно-белый формат. В таком случае точка может принимать значение в диапазоне $[0, 255]$, где 0 – это черный цвет, а 255 – белый.

Множество терминов для описания знаний образуют следующие термины: «признаки изображений», «возможные значения признаков изображений», «признаки точки изображения», «возможные значения признаков точки изображения», «признаки объекта» «признаки трехмерной точки» «возможные значения признака трехмерной точки» «возможные значения признаков объекта».

«Признаки точки изображения» является понятием, соответствующим разреженному множеству признаков. Термин «признаки двухмерной точки» обозначает конечное множество названий признаков точек изображения.

«Признаки изображений» является понятием, соответствующим разреженному множеству признаков. Термин «признаки изображений» обозначает конечное множество названий признаков точек изображения.

«Возможные значения признаков изображений» является понятием, соответствующим конечным отображениям. Областью определения отображения является множество изображений. Областью значений отображения является множество конечных подмножеств множества скалярных или размерных значений.

«Возможные значения признаков точки изображения» является понятием, соответствующим конечным отображениям. Областью определения отображения является множество названий признаков точки. Областью значений отображения является множество конечных подмножеств множества скалярных значений.

«Признаки трехмерной точки» является понятием, соответствующим разреженному множеству признаков. Термин «признаки трехмерной точки» обозначает конечное множество названий признаков трехмерных точек.

«Возможные значения признаков трехмерной точки» является понятием, соответствующим конечным отображениям. Областью определения отображения является множество названий признаков трехмерной точки. Областью значений отображения является множество конечных подмножеств множества скалярных значений.

«Признаки объекта» является понятием, соответствующим разреженному множеству признаков. Термин «признаки объектов» обозначает конечное множество названий признаков объектов изображения.

«Возможные значения признаков объекта» является понятием, соответствующим конечным отображениям. Областью определения отображения является множество названий признаков объектов. Областью значений отображения является множество конечных подмножеств множества скалярных или размерных значений.

2.3 Задачи решаемые в процессе построения трехмерной модели

- Построение карты глубины для стереоизображения.
- Нахождение соответствующих точек на стереоизображении.
- Распознавание трубопровода на стереоизображении.
- Расчет трехмерных координат для точек изображения.
- Построение облака точек.
- Создание трехмерной модели подводного трубопровода.

2.4 Задачи обработки, хранения и передачи информации

Обработка информации: обработка данные о точках стереоизображения.

Хранение данных: хранение информации о загруженных изображениях, хранение информации о трехмерных моделях.

2.5 Анализ смысла ситуации предметной области «Трехмерная реконструкция подводного трубопровода по стереопаре»

Ситуация – это случай создания трехмерной модели подводного трубопровода с использованием стереокамеры.

В процессе создания трехмерной модели подводного трубопровода с использованием стереокамеры анализируется информация о точках стереоизображения.

Терминами для описания ситуации являются «описание изображения», «значение признаков точки изображения», «значение признаков объекта», «значения признаков трехмерной точки».

«Описание изображения» является понятием, соответствующим множеству значений признаков. Термин «Описание изображения» обозначает конечное множество названий признаков рассматриваемого изображения.

«Значение признаков точки изображения» является понятием, соответствующим конечным отображениям значений признаков. Область определения данного отображения – множество, состоящее из названий признаков точки изображения. Область значений – множество, состоящее из возможных значений этих признаков.

«Значение признаков объекта» является понятием, соответствующим конечным отображениям значений признаков. Область определения данного отображения – множество, состоящее из названий признаков объекта. Область значений – множество, состоящее из возможных значений этих признаков.

«Значение признаков трехмерной точки» является понятием, соответствующим конечным отображениям значений признаков. Область определения данного отображения – множество, состоящее из названий

признаков трехмерной точки. Область значений – множество, состоящее из возможных значений этих признаков.

Ограничение целостности ситуации:

1. Входное стереоизображение должно содержать объект.
2. Входное стереоизображение должно быть сделано на одну стереокамеру в один момент времени.
3. Для любого признакового описания объекта, значения этого признакового описания принадлежит множеству значений этого признакового описания объекта.
4. Значение параметров объекта не принадлежит пустому множеству.
5. Для любого признакового описания трехмерной точки, значения этого признакового описания принадлежит множеству значений этого признакового описания трехмерной точки.
6. Значение признакового описания трехмерной точки не принадлежит пустому множеству.
7. Для любого признакового описания точки изображения, значения этого признакового описания принадлежит множеству значений этого признакового описания точки изображения.
8. Значение признакового описания точки изображения не принадлежит пустому множеству.

2.6 Анализ знаний предметной области «Трехмерная реконструкция подводного трубопровода по стереопаре»

Ограничение целостности знаний:

1. Для любой точки изображения, множество признаков, принадлежащих признаковому описанию этой точки изображения, является непустым множеством;
2. Для любой точки изображения, для каждого признака характерного для данной точки изображения, множество значений этого признака содержит все элементы;

3. Для любой трехмерной точки, для каждого признака характерного для данной трехмерной точки, множество значений этого признака является непустым множеством;

4. Для любой трехмерной точки, для каждого признака характерного для данной трехмерной точки, множество значений этого признака содержит все элементы;

5. Для любого объекта, для каждого признака характерного для данного объекта, множество значений этого признака является непустым множеством;

6. Для любого объекта, для каждого признака характерного для данного объекта, множество значений этого признака содержит все элементы.

Связь между двумя системами понятий задают следующие утверждения: матрица преобразований

Связь трехмерных координат изображения определяется с помощью карты глубины. Эта связь устанавливается при помощи математических функций преобразования координат двухмерных точек в трехмерные координат и настройки параметров изображения.

Связь трехмерных координат модели определяется с помощью триангуляции.

2.7 Модель преобразований

В данной главе описываются подробнее этапы создания трехмерной модели по стереопаре с математической точки зрения.

2.7.1 Ректификация

При использовании стереокамер часто говорят о ректификации, которая из себя представляет перенос двух плоскостей изображений в одну плоскость таким образом, чтобы все эпиполярные линии были параллельны

оси абсцисс и соответствующие эпполярные линии на обоих изображениях имели одинаковые ординаты.

Создавая один стереоснимок, можно сказать, что создается два отдельных изображения, которые различаются небольшим отклонением. Диспаратет, это расстояние между соответственными пикселями левого и правого изображений. Для его вычисления необходимо точное значение расстояния между матрицами обеих камер. Оно вычисляется по формуле:

$$d = |x_r - x_l| \quad (1)$$

где x_l - положение выбранного пикселя фиксированной строки u на левом изображении, x_r - положение соответственного пикселя этой строки на правом изображении, а d – диспаратет изображения

Пусть имеются две камеры (рисунок 7), где C — центр первой камеры, C' — центр второй камеры, а также точка пространства X проецируется в x на плоскость изображения левой камеры и в x' на плоскость изображения правой камеры. Прообразом точки x на изображении левой камеры является луч xX . Этот луч проецируется на плоскость второй камеры в прямую l' , называемую эпполярной линией. Образ точки X на плоскости изображения второй камеры обязательно лежит на эпполярной линии l' .

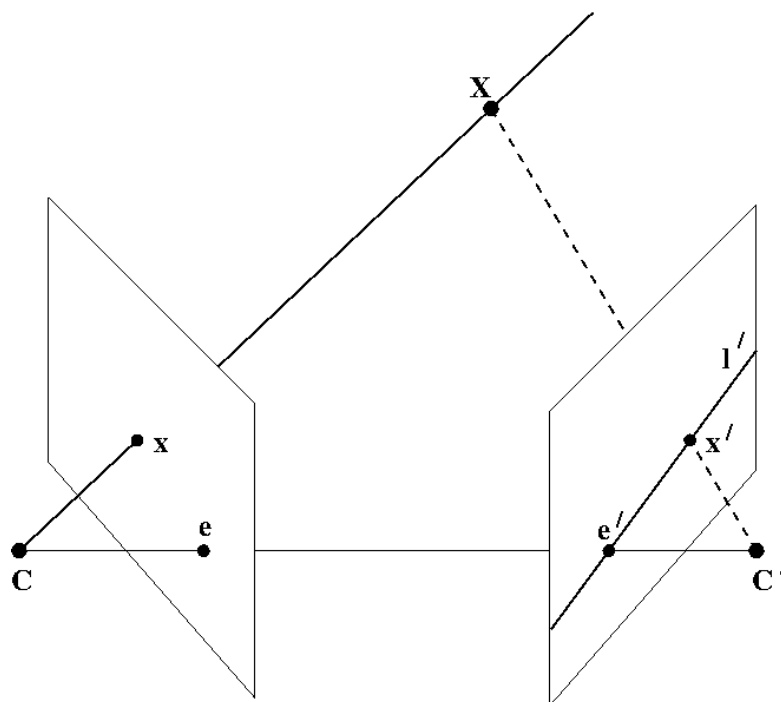


Рисунок 7— Вычисление положения точек при помощи эпиполярной геометрии в пространстве по двум изображениям

Таким образом, каждой точке x на изображении левой камеры соответствует эпилярная линия l' на изображении правой камеры. При этом пара для x на изображении правой камеры может лежать только на соответствующей эпилярной линии. Аналогично, каждой точке x' на правом изображении соответствует эпилярная линия l на левом.

Эпилярную геометрию используют для поиска стереопар, и для проверки того, что пара точек может быть стереопарой (т.е. проекцией некоторой точки пространства).

Эпилярная геометрия имеет очень простую запись в координатах. Пусть имеется пара откалиброванных камер, и пусть x — однородные координаты точки на изображении одной камеры, а x' — на изображении второй. Существует такая матрица F размера 3×3 , что пара точек x, x' является стереопарой тогда и только тогда, когда:

$$x'^T F x = 0 \quad (2)$$

Матрица F называется фундаментальной матрицей (fundamental matrix). Ее ранг равен 2, она определена с точностью до ненулевого множителя и

зависит только от матриц исходных камер P и P' . В случае, когда матрицы камер имеют вид $P = K[I|0]$, $P' = K'[R|t]$ фундаментальная матрица может быть вычислена по формуле:

$$F = K'^{-1T} R K^T [K R^T t] x \quad (3)$$

где для вектора e обозначение $[e]X$ вычисляется как

$$[e]x = \begin{bmatrix} 0 & -e_z & e_y \\ e_z & 0 & -e_x \\ -e_y & e_x & 0 \end{bmatrix} \quad (4)$$

С помощью фундаментальной матрицы вычисляются уравнения эпиполярных линий. Для точки x , вектор, задающий эпиполярную линию, будет иметь вид $l' = F x$, а уравнение самой эпиполярной линии: $l'^T x' = 0$. Аналогично для точки x' , вектор, задающий эпиполярную линию, будет иметь вид $l = F^T x'$.

Помимо фундаментальной матрицы, существует еще такое понятие, как существенная матрица (essential matrix): $E = K'^T F K$. В случае, когда матрицы внутренних параметров будут единичными существенная матрица будет совпадать с фундаментальной. По существенной матрице можно восстановить положение и поворот второй камеры относительно первой, поэтому она используется в задачах, в которых нужно определить движение камеры [20].

2.7.2 Карта глубины

Значения глубины обратно пропорциональны величине смещения пикселей. Зависимость между диспаратетом и глубиной можно выразить следующим способом:

$$\frac{T - d}{Z - f} = \frac{T}{Z} \rightarrow Z = \frac{fT}{d} \quad (5)$$

Где d – это диспаратет, f – фокусное расстояние стереокамеры, T – истинное расстояние до объекта, Z – расстояние до объекта в пикселях или единицах измерения, используемых для глубины.

Фокусное расстояние определяет границу, дальше которой не генерируется разницы высот. Это происходит потому, что при размытии становится невозможно определить соответствующий пиксель для другого изображения.

В этом уравнении d и Z связаны следующим образом:

1. $\frac{T-d}{Z-f}$ – дробь выражающая отношение, где $T-d$ – это глубина изображения, скорректированная на диспаратет.

2. $\frac{T}{Z}$ – дробь, представляющая истинное расстояние T к глубине Z .

Из-за обратной зависимости глубины и смещения, разрешающая способность систем стереозрения, которые работают на основе данного метода, лучше на близких расстояниях, и хуже на далеких. Главное достоинство стереокамер заключается в том, что солнечный свет им не только не мешает, но и наоборот, делает их результаты точнее.

2.7.3 Контур объекта на изображении

Для нахождения контура объекта на изображении используется обработка изображения с помощью маски. Маска представляет собой некую квадратную матрицу, обычно 3×3 , соответствующую указанной группе пикселей исходного изображения. Элементы матрицы принято называть коэффициентами. Также нахождение контура еще называют фильтрацией.

Процесс основан на перемещении маски от точки к точке. В каждой точке (x,y) отклик фильтра вычисляется с использованием предварительно заданных связей. В случае линейного поиска отклик фильтра задается суммой произведения коэффициентов на соответствующие значения пикселей в области, покрытой маской фильтра.

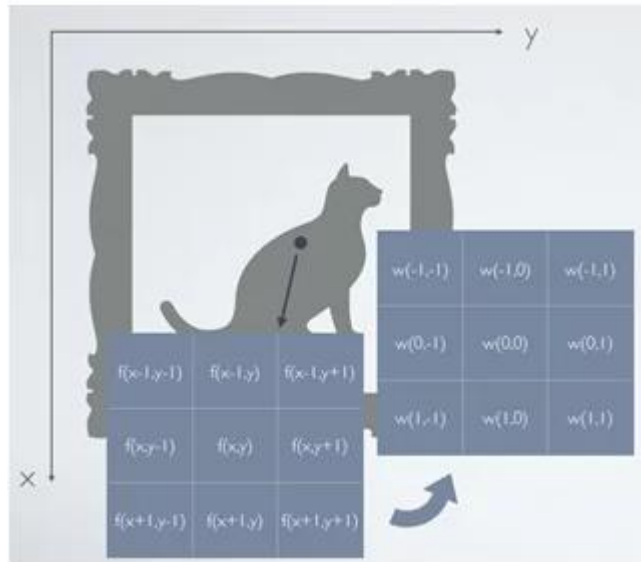


Рисунок 8 – Пример перемещения маски по точкам

Для точки (x,y) значение отклика фильтра R будет вычисляться так: каждый элемент окрестности точки (x,y) последовательно умножается на соответствующий элемент маски фильтра. Полученные значения суммируются.

$$R = w(-1,-1)f(x-1,y-1) + \dots + w(1,1)f(x+1,y+1) \quad (6)$$

Коэффициенты фильтра вычисляются на основе дифференцирования амплитуды сигнала. Так как производная – это отношение приращения функции к приращению аргумента, в случае изображения приращение аргумента будет всегда равно единице, т.е. одному пикселю, то это эквивалентно разности значений яркости соседних пикселей:

$$\frac{\partial f(x,y)}{\partial x} \sim \Delta f_x(x,y) = f(x,y) - f(x-1,y) \quad (7)$$

$$\frac{\partial f(x,y)}{\partial y} \sim \Delta f_y(x,y) = f(x,y) - f(x,y-1) \quad (8)$$

Далее после вычисления второй производной можно вывести модель вектора градиента:

$$|\nabla f| = \sqrt{G_x^2 + G_y^2} \quad (9)$$

Эта величина равна значению максимальной скорости изменения функции f в точке (x,y) .

Направление вектора градиента также является важной характеристикой. Обозначим $\alpha(x, y)$ угол между направлением вектора ∇f в точке (x, y) и осью x .

$$\alpha(x, y) = \arctg\left(\frac{G_y}{G_x}\right) \quad (10)$$

Отсюда легко найти направление контура в точке (x, y) , которое перпендикулярно направлению вектора градиента в точке. А вычислить градиент изображения можно, вычислив величины частных производных для каждой точки.

Оператор Робертса является одним из самых простых способов нахождения частных производных.

$$G_x = (z_9 - z_5) \quad (11)$$

$$G_y = (z_8 - z_6) \quad (12)$$

-1	0	0	-1
0	1	1	0

Рисунок 9 – Маски оператора

Эти производные могут быть реализованы путем обработки всего изображения с помощью оператора (рисунок 9), используя процедуру фильтрации, описанную ранее

2.7.4 Облако точек

Триангуляция точек – это решение такой задачи, целью которой является - определить трехмерные координаты точки по координатам ее проекций.

Пусть имеются две откалиброванные камеры с матрицами P_1 и P_2 . x_1 и x_2 — однородные координаты проекций некоторой точки пространства X . Тогда можно составить следующую систему уравнений:

$$\begin{cases} x_1 = P_1 X \\ x_2 = P_2 X \end{cases} \quad (13)$$

На практике для решения этой системы применяется следующий подход. Векторно умножают первое уравнение на x_1 , второе на x_{21} , избавляются от линейно зависимых уравнений и приводят систему к виду $A X = 0$, где A имеет размер 4×4 . Далее можно либо исходить из того, что вектор X является однородными координатами точки, положить его последнюю компоненту равной 1 и решать полученную систему из 3х уравнений с тремя неизвестными. Альтернативный способ — взять любое ненулевое решение системы $A X = 0$, например вычисленное, как сингулярный вектор, отвечающий наименьшему сингулярному числу матрицы A .

2.7.5 Фильтрация

Фильтрация точек позволяет устранить ложные, появившиеся в результате зашумленности точки в облаке, тем самым значительно улучшить качество получаемой в результате реконструкции модели (рисунок).



Рисунок 10 – Построение полигональной модели из облака точек после процесса фильтрации: (а) – облако точек, полученное в результате применения фильтрации к исходному облаку точек; (б) – полигональная модель, построенная из облака точек

Точка, относительно удаленная от кластера, построенного из k ближайших соседних с ней точек, с большой вероятностью будет ложная. Пусть для точки p множество k ближайших соседних точек — N_p . Для приближения кластера можно построить сферу минимального радиуса S ,

закрывающую в себя все точки множества N_p . Сравнивая расстояние d от точки p до центра сферы с радиусом сферы r , получим оценку правдоподобия того, что точка ложная (рисунок 11).

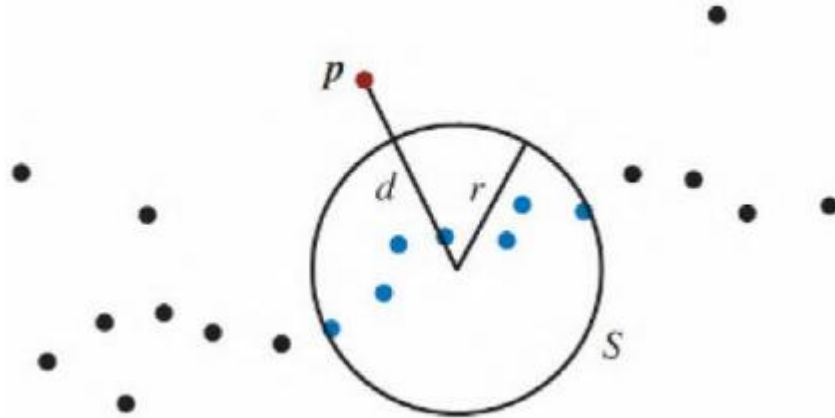


Рисунок 11 – Проекция точки X на плоскости изображений

Можно определить следующий критерий:

$$\chi(p) = \frac{d}{d + 2r/\sqrt{k}} \quad (14)$$

Увеличение радиуса сферы с увеличением числа соседних точек в (1) компенсируется за счет деления на \sqrt{k} .

Фильтрация осуществляется следующим образом. Сначала подсчитывается $\chi(p_i)$ для каждой точки p_i . Затем для всего множества $\{\chi(p_i)\}$ подсчитывается среднее значение μ и среднеквадратическое отклонение σ . Отбрасываются все точки p_i , для которых $\chi(p_i) > 3\sigma$. При реализации фильтрации можно использовать структуру октодерева для быстрого поиска k -ближайших точек. Если учитывается небольшое количество соседних точек, то можно также использовать структуру kdдерева, однако для $k \geq 10$ kd-дерево работает менее эффективно. Фильтр, описанный выше, позволяет удалить точки, находящиеся далеко от поверхности объекта. Однако он не может устранить высокочастотный шум, когда точки находятся очень близко к поверхности. Высокочастотный шум можно устранить с помощью нелокальной двусторонней фильтрации (NonLocal bilateral filtering), которая применяется для устранения шума на

изображениях. Для пикселя p и множества соседних пикселей $N(p)$ фильтрация определяется следующим образом:

$$I(p) = \frac{\sum_{u \in N(p)} W_c(p, u) W_s(p, u) I(u)}{\sum_{u \in N(p)} W_c(p, u) W_s(p, u)} \quad (15)$$

где $W_c(p, u)$ – мера близости между пикселями p и u , $W_s(p, u)$ – мера схожести между пикселями p и u .

В случае облака точек: $W_c(p, u)$ – расстояние между точками p и u , $W_s(p, u)$ – расстояние Хаусдорфа между множеством соседних с p точек $N(p)$ и множеством соседних с u точек $N(u)$.

Вычисление нормалей. Вычисление нормалей для точек позволяет определить внутреннюю и наружную стороны поверхности объекта, благодаря чему можно исключить точки, попавшие внутрь объекта. Один из распространённых методов оценки нормалей – алгоритм, предложенный Н. Норре. Этот метод строит приближение касательной плоскости к множеству соседних с p точек. Нормалью точки p считается нормаль к построенной касательной плоскости. Касательная плоскость определяется как плоскость, проходящая через центральную точку множества k -ближайших соседних точек N_p и имеющая наименьшее квадратическое отклонение от всех точек N_p . Для определения нормали строится матрица ковариации, размерностью 3×3

$$C = \sum_{i=1}^k (p_i - \bar{p})^T (p_i - \bar{p}) \quad (16)$$

Формула 16 позволяет вычислить сумму квадратов отклонений от среднего значения, где $\bar{p} = \frac{1}{k} \sum_{i=1}^k p_i$ – центральная точка множества N_p .

Нормаль u_i определяется как собственный вектор матрицы C , соответствующий наименьшему собственному числу матрицы C .

Вычисленные в результате работы алгоритма нормали для каждой точки не ориентированы. Так как положение камеры известно, то можно

определить и направление вектора от каждой точки к камере. Поэтому выбрать направление нормали можно следующим способом:

$$n_i = \begin{cases} u_i, & \text{если } u_i \cdot c > 0; \\ -u_i, & \text{если } u_i \cdot c \leq 0. \end{cases} \quad (17)$$

Однако качество работы данного метода значительно уменьшается, если модель содержит много острых углов. Для получения нормали используется локальное приближение поверхности касательной плоскостью. Вблизи острых углов такое приближение будет неточным, так как в множество соседних точек будут добавляться точки с обеих сторон угла, и полученная касательная плоскость будет находиться в промежуточном положении между гранями объекта (рисунок).

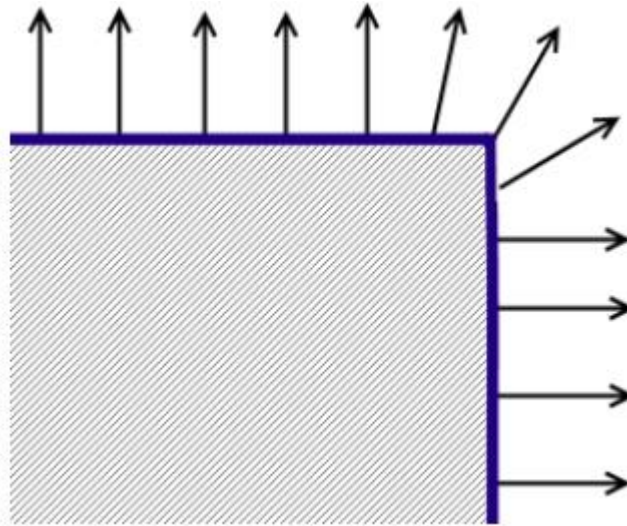


Рисунок 12 – Ошибка при вычислении нормалей вблизи острых углов

Для устранения ошибки при вычислении нормалей могут использоваться улучшения алгоритма. Например, при построении касательной плоскости можно учитывать веса соседних точек, которые зависят от удаленности точки от центра множества N_p (формула 18).

$$C = \sum_{i=1}^k w_i (p_i - \bar{p})^T (p_i - \bar{p}) \quad (18)$$

Формула 18 позволяет вычислить сумму взвешенных квадратов отклонений от среднего значения, где $w_i = e^{-|\bar{p}-p_i|}$.

Тогда точки, находящиеся дальше от центра, будут оказывать меньшее влияние на направление нормали. Также для устранения таких ошибок хорошо подходит алгоритм реорганизации соседей (Neighborhood reorganization) [25].

2.7.6 Создание трехмерной модели

Для создания трехмерной модели можно использовать гибридный рекурсивно-инкрементный алгоритм построения триангуляция Дилоне. Этот алгоритм позволяет избежать недостатков рекурсивного и инкрементного алгоритмов.

В начале на множестве S строится выпуклая оболочка $CH(S)$, каждое из ребер которой делается ориентированным с учетом обхода по часовой стрелке. Далее определяются медианы по абсциссе X_M и ординате Y_M . Проводятся горизонтальные и вертикальные прямые $X_M = \text{const}$ и $Y_M = \text{const}$. Ищутся точки пересечения этих прямых с соответствующими ребрами выпуклой оболочки. Сравнив расстояния по вертикали и горизонтали и учитывая предположение о случайности набора точек, за базовую берется прямая, для которой расстояние между точками пересечения меньше. Такой подход позволит быстрее разделить множество на два подмножества. Далее схема будет рассматриваться на примере базовой прямой (рисунок 12).

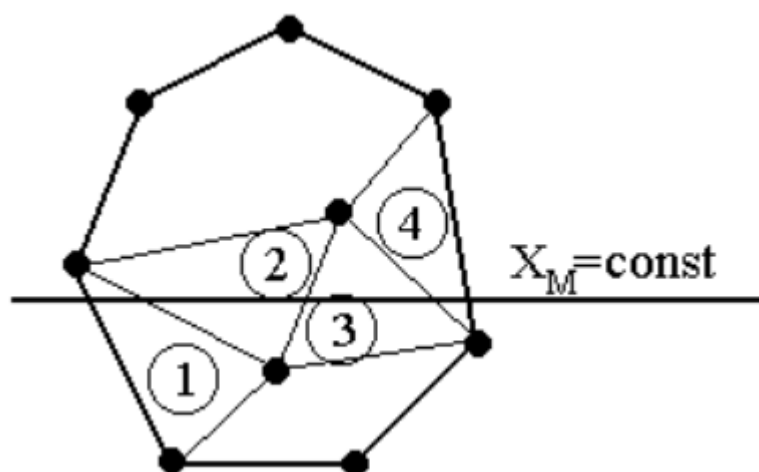


Рисунок 13 – Последовательность выделения разделяющей цепи выбранных треугольников

Теперь первым делом все непересекаемые ребра оболочки, расположенные выше X_M , заносятся в список ребер верхней оболочки, а точки как внутренние, так и граничные, расположенные выше X_M заносятся в верхний список точек; непересекаемые и расположенные ниже X_M ребра оболочки заносятся в список ребер нижней оболочки, а все точки расположенные ниже X_M помещаются в нижний список точек. Пусть первое исследуемое ребро будет левое ребро выпуклой оболочки, которое пересекает базовая прямая X_M . Используя инкрементный алгоритм находится подходящая точка, при этом определяется треугольник 1. Если прямая X_M пересекает условную верхнюю сторону, как показано на рисунке 12, то она используется для продолжения построения цепи разделяющих треугольников. Нижнее непересекаемое прямой X_M ребро добавляется в список нижней оболочки. Аналогично для треугольника 2 условно нижнее ребро используется для последующего определения треугольника 3, а верхнее непересекаемое прямой X_M ребро добавляется в список верхней оболочки. Точки, участвующие в цепи разделяющих треугольников, исключаются из соответствующих верхнего и нижнего списков точек. Построение разделяющей цепи треугольников заканчивается, когда второе ребро пересеканого треугольника совпадает с правым ребром оболочки,

пересекаемой прямой ХМ. На рисунке 12 это произошло для треугольника 4. В результате первого прохода построена цепь разделяющих треугольников Делоне 1, 2, 3, 4 и подготовлены данные для следующих проходов алгоритма. Рекурсивным окончанием разделения точек служит либо отсутствие оставшихся точек в условно верхнем или нижнем списке точек, либо наличие 3 оставшихся точек в таком списке, что образует последний треугольник [28, 33].

2.8 Краткий вывод

В ходе выполнения анализа предметной области были рассмотрены задачи этой предметной области, проведен анализ знаний, а также построена математическая модель.

3 Разработка проекта программной системы

В данной главе рассматривается проектирование программной системы для трехмерной реконструкции по стереопаре. В процессе проектирования также проводится анализ требований, для получения информации о специфике задачи и уточнения рамок проекта.

Разрабатываемая программная система будет относиться к прикладному программному обеспечению на основании классификатора представленного в приказе Министерства цифрового развития, связи и массовых коммуникаций российской федерации от 22 сентября 2020 г. № 486 об утверждении классификатора программ для электронных вычислительных машин и баз данных.

Прикладное программное обеспечение предназначенный для решения практических задач и разработки.

Программный модуль – это логически заверченный фрагмент кода, который решает конкретную задачу и компилируется, и тестируется независимо. Такой модель может применяться в других программах как

основной модуль или расширение. Разбиение программы на модули позволяет решить общую задачу, которая разбивается на более мелкие. Это в свою очередь увеличивает скорость работы и читабельность кода. Модели могут объединяться в пакеты, а пакеты в библиотеки. Поскольку модули независимы от основной программы их доработка не затрагивает всю программу и другие модули.

Модульность также позволяет распределить нагрузку между серверами и увеличить скорость выполнения программы.

3.1 Диаграмма вариантов использования

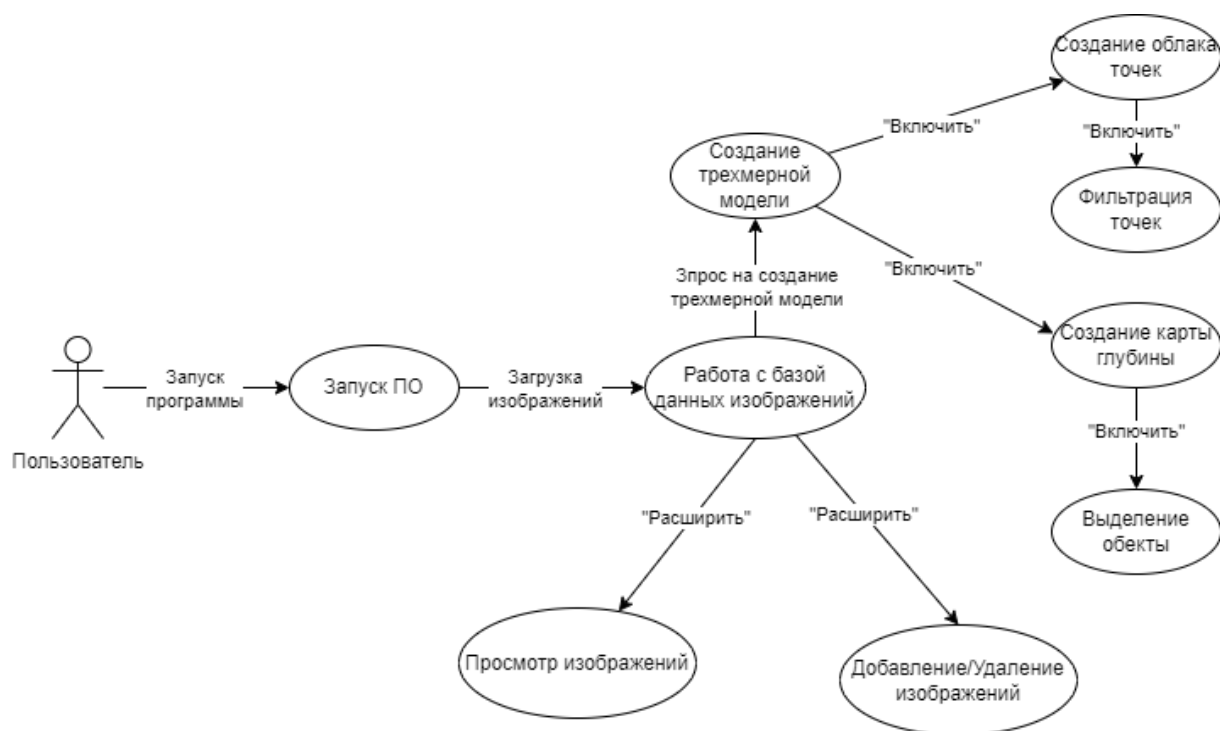


Рисунок 14 – Диаграмма вариантов использования

На рисунке 14 представлена диаграмма вариантов использования. В ней находятся основные варианты использования: работа с базой данных изображений и создание трехмерной модели, а также возможно посмотреть изображение или добавить(удалить) его в базу данных.

3.2 Архитектурно-контекстная диаграмма 1-го уровня

На рисунке 15 представлена архитектурно-контекстная диаграмма 1-го уровня системы по трехмерной реконструкции подводного трубопровода.

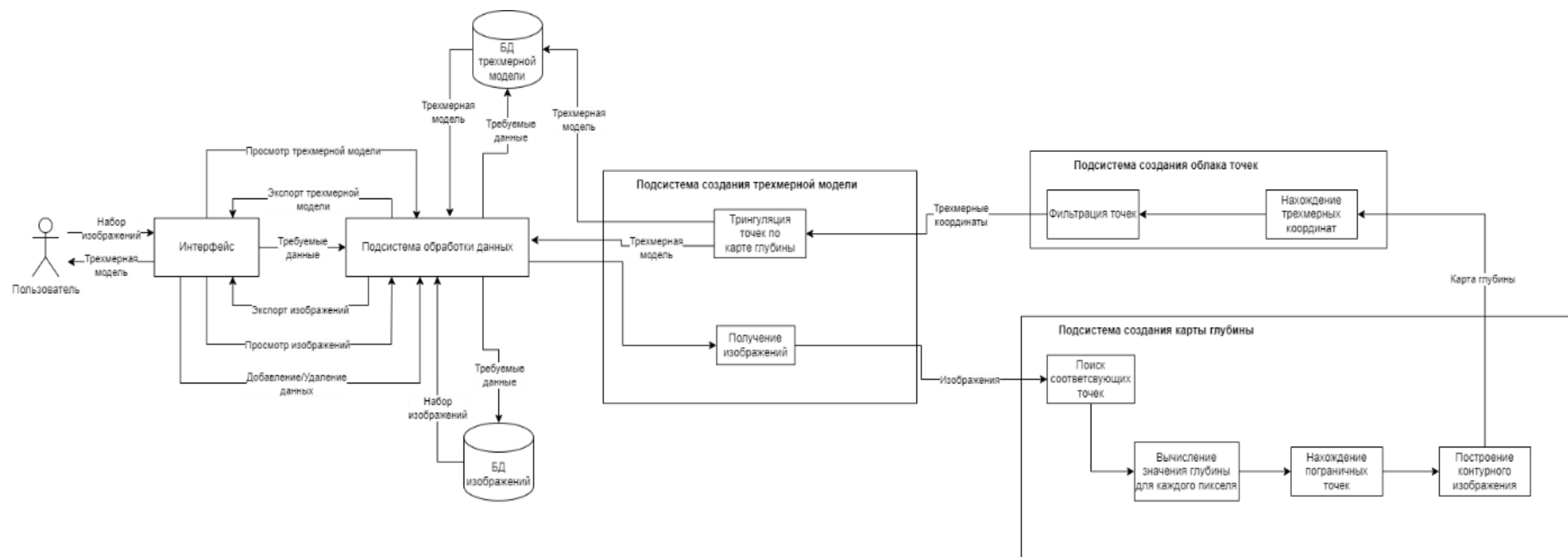


Рисунок 15 – Архитектурно-контекстная диаграмма 1-го уровня

На диаграмме выше видно, что в системе есть пять подсистем: подсистема «Интерфейс», подсистема обработки данных, подсистема создания трехмерной модели, подсистема создания облака точек, подсистема создания карты глубины. Пользователь через интерфейс загружает стереоизображение в подсистему создания трехмерной модели. Далее она передает эти стереоизображения в подсистему создания карты глубины. После чего создание карты глубины и контур изображения. На карте глубины, после нахождения контура, выделяется объект. Затем строятся трёхмерные координаты точек объекта по карте глубины. И уже после фильтрации трехмерных точек, строится трехмерная модель.

Архитектурно-контекстная диаграмма 1-го уровня включает в себя следующие компоненты:

1. Пользовательский интерфейс.
2. Подсистема обработки данных.
3. База данных изображений.
4. База данных трехмерных моделей.
5. Подсистема создания трехмерной модели.
 - 5.1. Получение изображений.
 - 5.2. Триангуляция точек по карте глубины.
6. Подсистема создания карты глубины.
 - 6.1. Поиск соответствующих точек.
 - 6.2. Нахождение пограничных точек.
 - 6.3. Вычисление значения глубины для каждого пикселя.
 - 6.4. Построение контура изображения.
7. Подсистема создания облака точек
 - 7.1. Нахождение трехмерных координат.
 - 7.2. Фильтрация точек.

3.3 Диаграмма последовательности состояний

Диаграмма последовательности состояний позволяет описать взаимодействие между объектами в системе в виде последовательности сообщений, действий и операций, отображая порядок выполнения действий и обмена информацией между объектами.

Ниже представлена диаграмма последовательностей состояний для системы трехмерной реконструкции (рисунок 16).

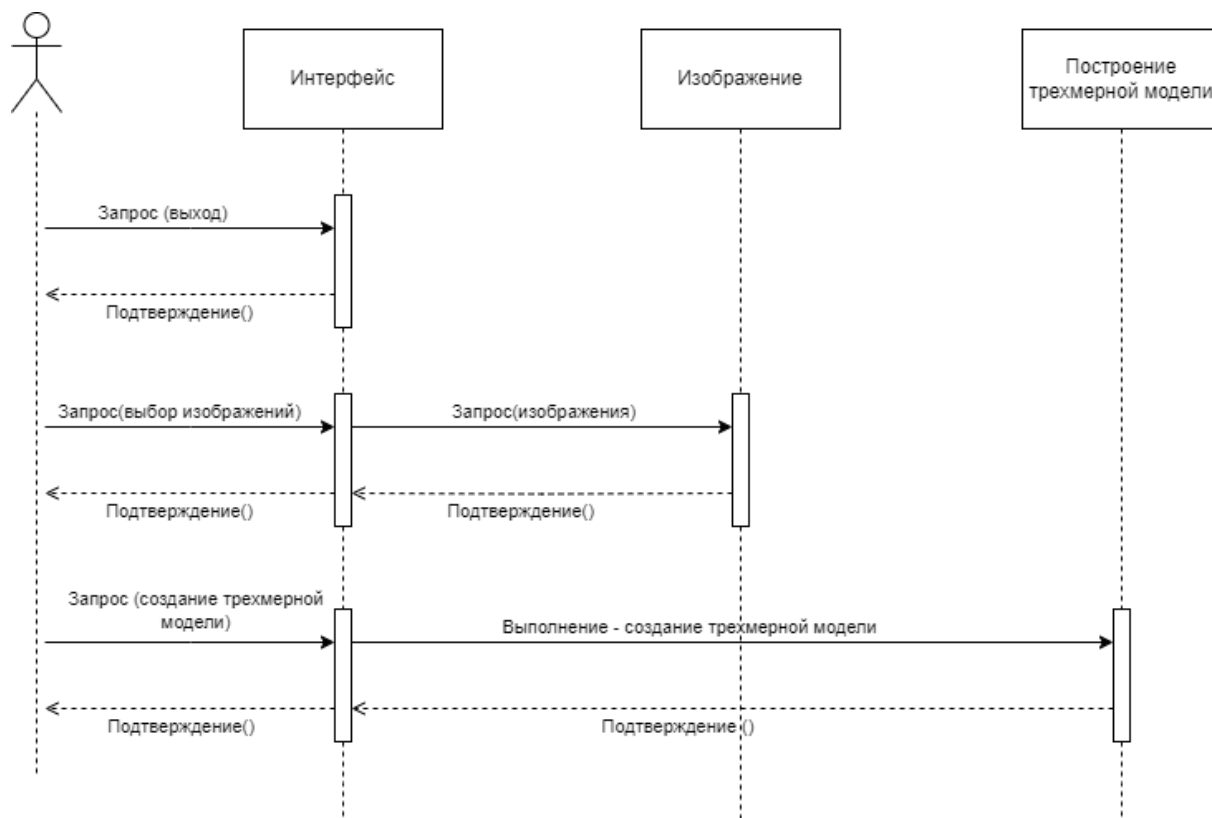


Рисунок 16 – Диаграмма последовательности состояний

3.4 Диаграмма перехода из состояния в состояние

Диаграмма перехода из состояния в состояние графически показывает, как происходит процесс изменения состояний системы в зависимости от определенных событий. Диаграмма помогает моделировать динамические аспекты системы.

На рисунке 17 представлена диаграмма перехода из состояния в состояние системы по трехмерной реконструкции подводного трубопровода.

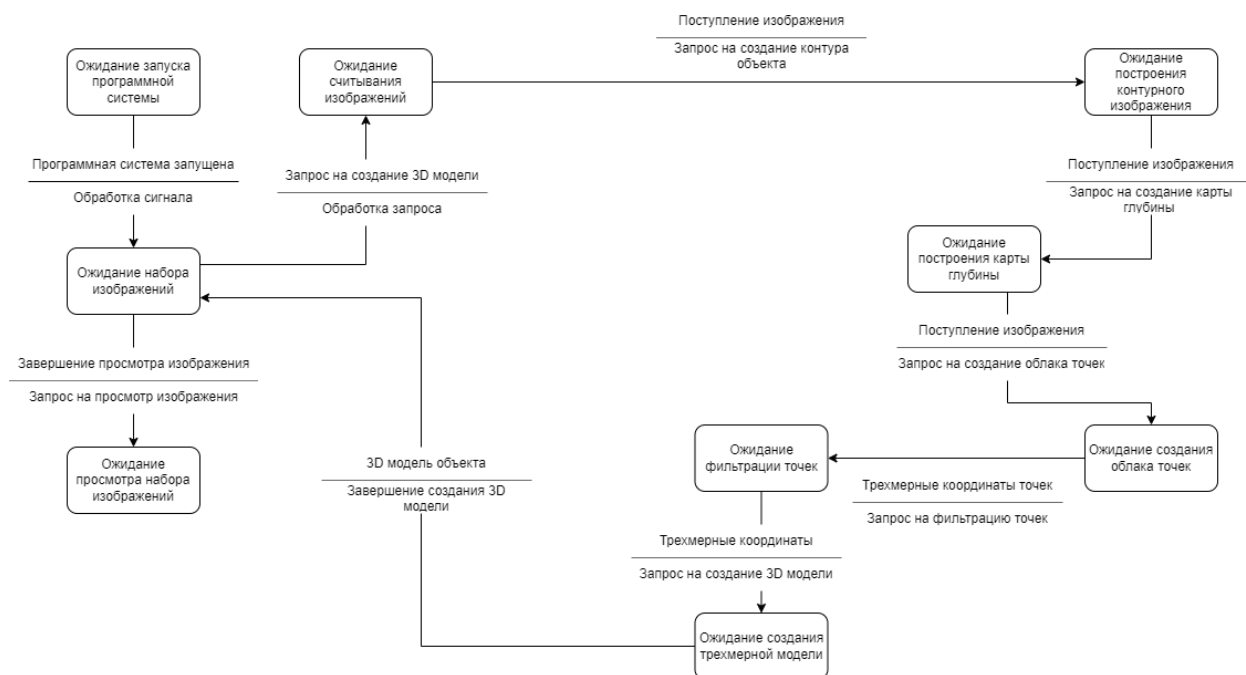


Рисунок 17 – Диаграмма перехода из состояния в состояние

На данной диаграмме можно увидеть изменения состояний программной системы от запуска до экспортирования созданной трехмерной модели.

3.5 Диаграмма потоков данных

Диаграмма потоков данных (рисунок 18) графически показывает поток данных в информационной системе. Данная диаграмма позволяет описать входящие и выходящие потоковые данные и хранилища этих данных.

В данном случае на диаграмме можно увидеть, как протекают данные при загрузке изображений пользователем, при добавлении/удалении изображений, а также при формировании трехмерной модели.

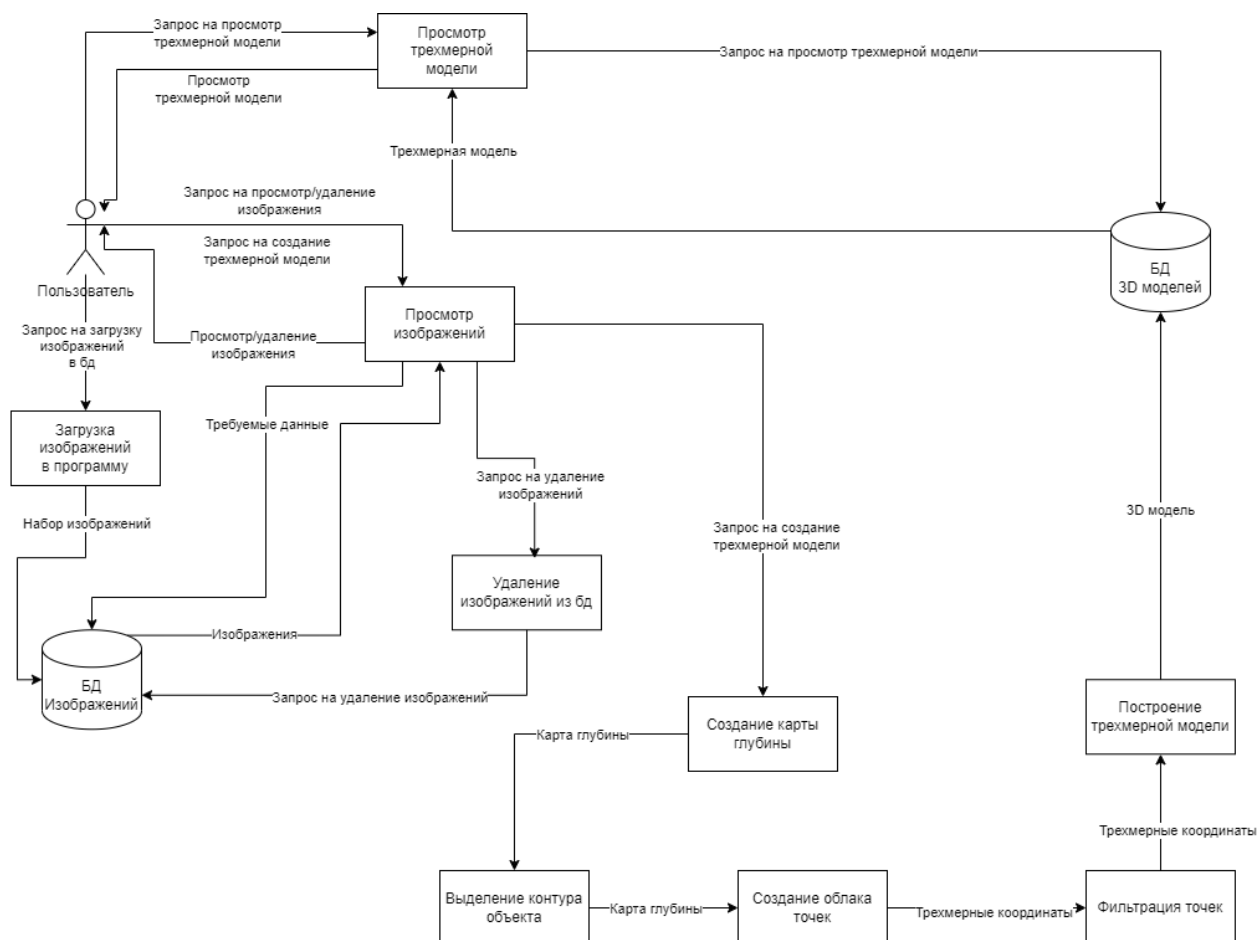


Рисунок 18 – Диаграмма потоков данных

3.6 Описание функциональных требований

Система для трехмерной реконструкции должна соответствовать набору функциональных требований, которые выделяются в процессе ее проектирования.

В рамках системы должна быть реализована возможность создания трехмерной реконструкции по стереоизображению, которое подается на вход. Внутри процесса создания трехмерной реконструкции должны быть реализованы функции:

- создания карты глубины;
- выделение объекта на изображении;
- создание облака точек;
- фильтрация облака точек;
- создание объекта по облаку точек.

Функция создания карты глубины должна на основании стереоизображения создавать матрицу значений глубины для каждой точки ректифицированного изображения. Ректификация происходит в момент создания карты глубины. Функция выделения объекта должна на основании матрицы карты глубины находить объект и отсеивать все точки, которые не принадлежат объекту. Функция создания облака точек должна на основании матрицы карты глубины и фокусного расстояния стереокамеры создавать список координат трехмерных точек объекта. Фокусное расстояние задается перед работой процесса. Функция фильтрации облака точек должна отсеивать некое количество точек, которые образуют слишком большое скопления в определенном радиусе одной точки. Функция создания трехмерной модели должна на основании списка трехмерных координат создавать модель объекта.

3.7 Требования к входным данным

Входные данные системы трехмерной реконструкции подводного трубопровода по стереопаре является набор файлов .JPG или .PNG. Каждый файл содержит изображение, загруженное пользователем.

Требования к входному файлу изображения:

- минимальное допустимое разрешение: 640*640;
- максимальное допустимое разрешение: 2000*2000;
- максимальный размер файла: 50 Мбайт;
- минимальный размер файла 1Мбайт;
- наименование файла должно начинаться с маленькой или заглавной буквы латинского алфавита;
- в наименовании могут присутствовать только буквы латинского алфавита или цифры;
- размер наименования не должен превышать 15 символов;
- формат файла должен быть .JPG или .PNG.

Входными данными подсистемы создания карты глубины является входные изображения.

Требования к входным изображениям:

- каждая пара изображений должна быть стереоизображением одной сцены;
- изображения содержат хоть один объект сцены.

Входными данными подсистемы создания облака точек является матрица значений глубины и фокусное расстояние стереокамер.

Требования к входной матрице и фокусному расстоянию:

- фокусное расстояние определяется вещественным числом;
- матрица глубины должна содержать значения глубины для каждой точки, принадлежащей сцене;
- матрица глубины должна быть не пустой;
- значения в матрице должны быть вещественные числа.

Входными данными подсистемы создания трехмерной модели является список трехмерных координат точек (облако точек) сцены.

Требования к входному списку:

- список точек должен быть отфильтрован;
- список точек содержать точки, принадлежащие сцене;
- у каждой точки должны быть координаты x , y , z ;
- каждая координата должна быть описана вещественным числом;

3.8 Требования к выходным данным

Выходными данными системы трехмерной реконструкции подводного трубопровода по стереопаре является файл формата .OBJ. В файле содержится трехмерный объект, полученный после обработки стереоизображений. В рамках работы подсистем, на выходе, получается основной файл. Требования к файлу прописаны ниже.

Выходными данными подсистемы создания карты глубины является матрица значений глубины для каждого пикселя ректифицированного изображения.

Требования к выходной матрице:

- значения в матрице должны быть вещественными;
- матрица должна содержать значения для каждой точки ректифицированного изображения;

Выходными данными подсистемы создания облака точек является список точек и их трехмерные координаты.

Требования к выходной матрице:

- список должен содержать трехмерные координаты точек: x , y , z ;
- каждая координата должна быть описана вещественным числом;

Выходными данными подсистемы создания трехмерного объекта является файл формата .OBJ, системы трехмерной реконструкции подводного трубопровода по стереопаре.

Требования к выходному файлу:

- максимальный размер файла: 100 Мбайт;
- наименование начинается с латинской буквы;
- в наименовании допустимо использовать цифры и буквы латинского алфавита;
- запрещено в наименовании файла использовать спец символы;
- минимальный размер наименования 5 символов, максимальный – 30 символов;
- формат файла .OBJ.

3.9 План тестирования

Каждое программное обеспечение (ПО) проходит процесс тестирования, который позволяет проверить ПО на наличие ошибок при его

работе. Процесс включает в себя проектирование тестов, выполнение тестирования и анализ полученных результатов.

Тестирование для проверки программного средства на наличие ошибок приведено при помощи деления входных данных на классы эквивалентности.

В таблице 1 представлено деление входных данных на классы эквивалентности.

Таблица 1 – Классы эквивалентности

Входные данные	Номер правильного класса эквивалентности	Правильный класс эквивалентности	Номер неправильного класса эквивалентности	Неправильный класс эквивалентности
Изображение формата {JPG и .PNG} с минимальным разрешением 640*640 b максимальным разрешением 2000*2000, максимальным размером 50 Мбайт, минимальный размер 1 Мбайт. Наименование файла – последовательность символов, первый символ принадлежит множеству {A, B, ..., Z, a, b, ..., z}, a последующие {A... Z, a... z, 0, 1, ..., 9}	(35)	Добавлено изображение формата {JPG и .PNG} с разрешением в пределах от 640*640 до 2000*2000, размера файла от 1 Мбайта до 50 МБайт Наименование файла = {a ₁ b ₁ b ₂ ... b _k }, где a ₁ {A...Z}, b _k {A... Z, a... z, 0, 1, ..., 9}	(1)	Добавлено изображение, где у файла изображения разрешение меньше 640*640
			(2)	Добавлено изображение, где файл меньше 1Мбайта
			(3)	a ₁ {a...z }
			(4)	Добавлено изображение, где у файла изображения наименование содержит не только латинские буквы или цифры
			(5)	Добавлено изображение, где у файла формат не .png или .jpg

Продолжение таблицы 1

Входные данные	Номер правильного класса эквивалентности	Правильный класс эквивалентности	Номер неправильного класса эквивалентности	Неправильный класс эквивалентности
			(6)	Добавлено изображение, где у файла изображения размер больше 50 Мбайт
			(7)	Добавлено изображение, где у файла изображения разрешение больше 2000*2000
Удаление изображения	(36)	Удалено изображение с указанным наименованием из набора изображений	(8)	Удалено изображение из набора изображений, изображения с указанным наименованием в наборе нет
			(9)	Удаление изображения, набор изображений пуст
Составление карты глубины	(37)	Создана карта глубины, в которой содержатся только вещественные символы и содержаться точки ректифицированного изображения на основании стереоизображения,	(10)	Создана карта глубины, стереоизображение содержит разные сцены
			(11)	Создана карта глубины, изображения не содержат объекты на сцене

Продолжение таблицы 1

Входные данные	Номер правильного класса эквивалентности	Правильный класс эквивалентности	Номер неправильного класса эквивалентности	Неправильный класс эквивалентности
		содержащего объекты на сцене, стереопара принадлежит одной сцене.	(12)	Создана карта глубины, где содержаться не только вещественные числа
			(13)	Создана карта глубины, где потеряны точки ректифицированного изображения
Создание трехмерной модели которая сохраняется в файл форматом { .OBJ } с максимальным размером 100Мбайт по отфильтрованному облаку точек, в облаке точек все точки принадлежат сцене, каждая точка имеет три координаты и каждая координата это вещественное число	(38) Создано трехмерная модель которая сохраняется в файл форматом { .OBJ } с размером от 1 Мбайта до 100Мбайт по отфильтрованному облаку точек, в облаке точек все точки принадлежат сцене, каждая точка имеет три координаты и каждая координата это вещественное число	Наименование файла = { a_1 b_1 b_2 ... b_k }, где $a_1 \{A, B, \dots, Z, a, b, \dots, z\}$, $b_k \{A \dots Z, a \dots z, 0, 1, \dots, 9\}$, $k \leq 30$, $k \geq 5$	(14)	Создана трехмерная модель, облако точек пусто
			(15)	Создана трехмерная модель, неотфильтровано облако точек
			(16)	Создана трехмерная модель, отфильтрованное облако точек не содержит одну из координат
			(17)	Создана трехмерная модель, координаты отфильтрованного облака точек содержат не только вещественные символы

Продолжение таблицы 1

Входные данные	Номер правильного класса эквивалентности	Правильный класс эквивалентности	Номер неправильного класса эквивалентности	Неправильный класс эквивалентности
Наименование файла – последовательность символов, первый символ принадлежит множеству {A, B, ..., Z, a, b, ..., z}, }, а последующие {A... Z, a... z, 0, 1, ..., 9}			(18)	Создана трехмерная модель, отфильтрованное облако точек содержит точки не принадлежащие сцене
			(19)	Создана трехмерная модель, где наименование выходного файла менее 5 символов
			(20)	Создана трехмерная модель, где наименование выходного файла более 30 символов
			(21)	Создана трехмерная модель, где формат файла не .OBJ
			(22)	Создана трехмерная модель, где в наименовании присутствуют не только латинские буквы и цифры

Продолжение таблицы 1

Входные данные	Номер правильного класса эквивалентности	Правильный класс эквивалентности	Номер неправильного класса эквивалентности	Неправильный класс эквивалентности
Создание облака точек	(39)	Создано облако точек, в котором для каждой точки присутствует три координаты и каждая координата это вещественное число, где карта глубины содержит значения всех точек, значения вещественные числа и фокусное расстояние вещественное число	(23)	Создана трехмерная модель, где размер выходного файла более 100МБ
			(24)	$a_1\{a, \dots, я, A, \dots, Я\}$
			(25)	Создано облака точек, где фокусное расстояние не вещественное число
			(26)	Создано облако точек, где карта глубины содержит значения не для всех точек
			(27)	Создано облако точек, где карта глубины содержит не вещественные числа
			(28)	Создано облако точек, где карта глубины пустой массив
			(29)	Создано облако точек, где в выходном облаке точек не для каждой точки есть три координаты

Окончание таблицы 1

Входные данные	Номер правильного класса эквивалентности	Правильный класс эквивалентности	Номер неправильного класса эквивалентности	Неправильный класс эквивалентности
			(30)	Создано облако точек, где в выходном облаке точек не все координаты вещественные числа
Фильтрация облака точек	(40)	Облако точек отфильтровано, облако точек содержит точки сцены, координаты являются вещественными числами и для каждой точки есть значения	(31)	Облако точек отфильтровано, облако точек пусто
			(32)	Облако точек отфильтровано, облако точек содержит не только точки сцены
			(33)	Облако точек отфильтровано, не все точки имеют координаты
			(34)	Облако точек отфильтровано, координаты содержат не только вещественные числа

Таблица 2 – Тест 1

№	Входные данные	
1	Изображение «Img1» формата .JPG, размер файла 10 Мбайт добавить в набор изображений	
	Выходные данные	Покрытые классы эквивалентности
	Изображение добавлено в набор изображений	35

Таблица 3 – Тест 2

№	Входные данные	
2	Изображение «1_Im» формата .RAW, размер файла 100 Мбайт, разрешением 2500*2300	
	Выходные данные	Покрытые классы эквивалентности
	-	3, 6, 7

Таблица 4 – Тест 3

№	Входные данные	
3	Удалено изображение «Img1» формата .JPG из пустого списка	
	Выходные данные	Покрытые классы эквивалентности
	-	8, 9

Таблица 5 – Тест 4

№	Входные данные	
4	Изображение «Img2» формата .JPG, размер файла 12 МБайт добавлено и удалено из набора изображений	
	Выходные данные	Покрытые классы эквивалентности
	Изображение	35, 36

Таблица 6 – Тест 5

№	Входные данные	
5	Создание карты глубины по стереопаре с одной сцены с объектом	
	Выходные данные	Покрытые классы эквивалентности
	Список значений глубины для каждого пикселя	37

Таблица 7 – Тест 6

№	Входные данные	
6	Создание облака точек по карте глубины, которая содержит значение для каждой точки ректифицированного изображения, и каждое значение вещественное число. И фокусное расстояние вещественное число	
	Выходные данные	Покрытые классы эквивалентности
	Список трехмерных точек	39

Таблица 8 – Тест 7

№	Входные данные	
7	Фильтрация облака точек, где все точки принадлежат сцене, координаты являются вещественными числами, и каждая точка имеет три координаты	
	Выходные данные	Покрытые классы эквивалентности
	Отфильтрованное облако точек	40

Таблица 9 – Тест 8

№	Входные данные	
8	Создание трехмерной модели по отфильтрованному облаку точек, где каждая точка принадлежит сцене и имеет три координаты и каждая координата это вещественное число	
	Выходные данные	Покрытые классы эквивалентности
	Трехмерная модель	38

Таблица 10 – Тест 9

№	Входные данные	
9	Создание карты глубины по двум изображениям, где одно из изображений не принадлежит сцене и оба изображения не содержат объекты	
	Выходные данные	Покрытые классы эквивалентности
	-	10, 11

Таблица 11 – Тест 10

№	Входные данные	
10	Создание облака точек по пустой карте глубины	
	Выходные данные	Покрытые классы эквивалентности
	-	26, 28

Таблица 12 – Тест 11

№	Входные данные	
11	Создание облака точек по карте глубины, где значения целые числа и фокусное расстояние целое число	
	Выходные данные	Покрытые классы эквивалентности
	-	25, 27

Таблица 13 – Тест 12

№	Входные данные	
12	Фильтрация пустого облака точек	
	Выходные данные	Покрытые классы эквивалентности
	-	31

Таблица 14 – Тест 13

№	Входные данные	
13	Фильтрация облака точек содержащего точки не только сцены, а также есть точки у которых значения координат не вещественные числа и отсутствует одна из координат	
	Выходные данные	Покрытые классы эквивалентности
	-	32, 33, 34

Таблица 15 – Тест 14

№	Входные данные	
14	Создание трехмерной модели на пустом облаке точек	
	Выходные данные	Покрытые классы эквивалентности
	-	14

Таблица 16 – Тест 15

№	Входные данные	
15	Создание облака точек на неотфильтрованном облаке точек, где облако точек не содержит одну из координат, значение координат не вещественные числа и есть точки, которые не принадлежат сцене	
	Выходные данные	Покрытые классы эквивалентности
	-	15, 16, 17, 18

Таблица 17 – Тест 16

№	Входные данные	
16	Создание карты глубины по стереопаре с одной сцены с объектом	
	Выходные данные	Покрытые классы эквивалентности
	Список значений глубины, где не у всех точек значение вещественное число и отсутствуют точки ректифицированного изображения	12, 13

Таблица 18 – Тест 17

№	Входные данные	
17	Создание облака точек по карте глубины, которая содержит значение для каждой точки ректифицированного изображения, и каждое значение вещественное число. И фокусное расстояние вещественное число	
	Выходные данные	Покрытые классы эквивалентности
	Список трехмерных точек, где не у всех значение координат вещественное число и не у всех точек три координаты	29, 30

Таблица 19 – Тест 18

№	Входные данные	
18	Создание трехмерной модели по отфильтрованному облаку точек, где каждая точка принадлежит сцене и имеет три координаты и каждая координата это вещественное число	
	Выходные данные	Покрытые классы эквивалентности
	Трехмерная модель, где файл формата не .OBJ, наименование файла менее 5 знаков и начинается не с латинской буквы, размер файла более 100Мб	19, 21, 23, 24

Таблица 20 – Тест 19

№	Входные данные	
19	Создание трехмерной модели по отфильтрованному облаку точек, где каждая точка принадлежит сцене и имеет три координаты и каждая координата это вещественное число	
	Выходные данные	Покрытые классы эквивалентности
	Трехмерная модель, где формат файл формата не .OBJ, наименование файла более 30 знаков и начинается не с латинской буквы и присутствуют спец символы, размер файла более 100Мб	20, 21, 22, 23, 24

Таблица 21 – Тест 20

№	Входные данные	
20	Изображение «1_Im» формата .RAW, размер файла 0,5 Мбайт, разрешением 300*200	
	Выходные данные	Покрытые классы эквивалентности
	-	1, 2, 3, 4, 5

3.10 Краткий вывод

В ходе разработки проекта программной системы были описаны подсистемы трехмерной реконструкции подводного трубопровода по стереопаре, составлены требования к подсистемам и составлен план тестирования.

4 Реализация и тестирование

В этой главе будет рассматриваться процесс реализации и тестирования программного средства по трехмерной реконструкции подводного трубопровода.

4.1 Язык программирования

Для реализации алгоритма трехмерной реконструкции подводного трубопровода был выбран Python.

Python – это высокоуровневый объектно-ориентированный язык программирования с динамической типизацией. Синтаксис языка является простым в освоении. Он также поддерживает модульность и пакетирование программы, что повышает читабельность кода и снижает трудозатраты на написание программ. [36]

4.2 Среда разработки

Для реализации функционала трехмерной реконструкции использовалась среда разработки PyCharm. Это среда разработки, которая позволяет писать на языке разработки Python. PyCharm позволяет писать код на любой платформе: Windows, Linux и Mac OS, а также отслеживать версии с помощью системы управления версиями Git. Также PyCharm позволяет интегрировать различные фреймворки и библиотеки, которые позволяют работать с машинным зрением.

4.3 Вспомогательные ПО

Для реализации алгоритма было использованы вспомогательные средства, такие как OpenCV, Open3d

OpenCV – это библиотека для работы с алгоритмами компьютерного зрения, обработки изображения и видео [44]. Она позволяет распознать на изображении лица, откалибровать камеру, а также может распознать файл

изображения и выдать из него массив пикселей со значением цвета для каждого пикселя.

Open3d – это библиотека для работы с 3D данными [45]. Она позволяет создать облако точек, отфильтровать его, вывести визуально облако точек, а также можно создать модель объекта по облаку точек.

4.4 Тестирование

Было проведено тестирование базового функционала (добавление, удаление, создание трехмерной модели, создание карты глубины, создание облака точек, фильтрация облака точек) в соответствии с классами эквивалентности, представленными в разделе «План тестирования».

Тестирование программного средства проводилось на ноутбуке. Характеристики приведены в таблице 22

Таблица 22 – Характеристики

Наименование	Характеристика
Модель	Acer
Процессор	Intel(R) Core(TM) i5-5200U
ОЗУ	6,00 ГБ
Видеокарта	NVIDIA GeForce 840M
Тактовая частота	2.20 GHz

Для первичного тестирования были заданы следующие ограничения:

- Формат изображения JPG.
- Размер изображения до 6 Мб.

Тест №1

Ситуация: в программном средстве нужно было создать карту глубины.

Результат: создана новая карта глубины.

Тест №2

Ситуация: в программном средстве необходимо отфильтровать облако точек.

Результат: облако точек отфильтровано

Тест №3

Ситуация: необходимо создать облако точек на основе существующей карты глубины.

Результат: создан новый набор трехмерных точек.

Тест №4

Ситуация: необходимо создать трехмерную модель на основе отфильтрованного облака точек.

Результат: создана новая трехмерная модель.

4.5 Результаты экспериментов

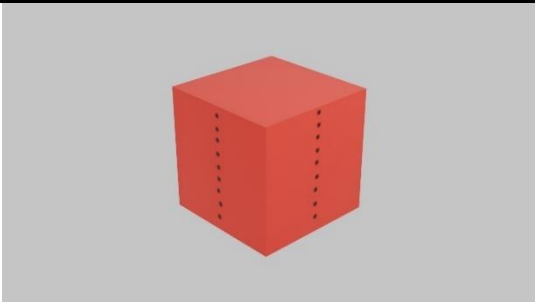
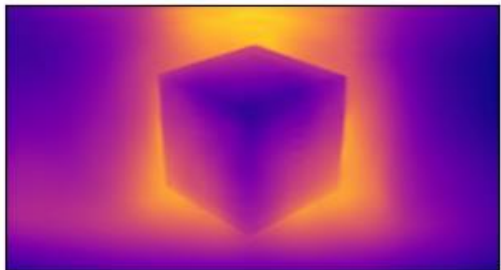
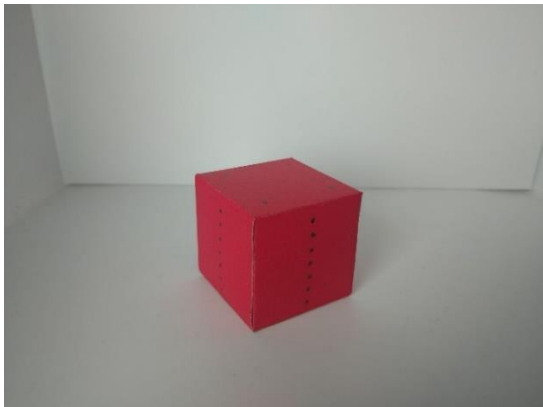
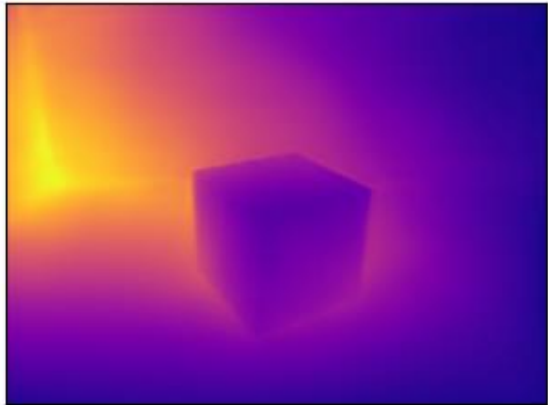
Для проведения экспериментов над основным функционалом разработанных программных модулей программного средства по 3D реконструкции были разработаны две модельных сцены. Первая сцена была сделана с применением специализированного программного обеспечения для 3D моделирования (Blender) и содержит объект, размещённый в идеальных условиях (однотонный фон, контрастирующий с объектом). Сам объект имеет простые формы, яркий контрастный однотонный цвет с выделением ключевых отметок на гранях. Такая сцена далее будет обозначена как сцена с идеальными условиями. Вторая сцена была выполнена с применением физических компонентов, по визуалу приближена к первой сцене, фоновые условия осложнены рядом цветовых тонов и тенями. Данная сцена будет обозначена как сцена со стандартными условиями.

4.5.1 Эксперименты с картой глубины

В рамках работы над программным средством для трехмерной реконструкции были проведены эксперименты для алгоритма создания карты глубины.

Для создания карты глубины на вход подается изображение, а на выходе получается карта глубины входного изображения. Результат представлен в таблице 23.

Таблица 23 – Эксперимент по созданию карты глубины при идеальных и стандартных условиях

№	Изображение	Карта глубины
1		
2		

Эксперимент 1 был проведен с использованием изображения сцены с идеальными условиями. Эксперимент 2 был проведен с использованием изображения сцены со стандартными условиями.

Стоит обратить внимание, что на карте глубины яркие места, это тени, а затемнённые – это освещение.

Между результатами данных экспериментов можно заметить разницу в освещенности объекта и фона. В первом случае, поскольку на фоне нет теней, объект, четко выражен на карте глубины, за счет контраста объекта и фона. Во втором случае объект менее выражен на карте глубины, за счет угла освещения и контрастности. Можно заметить, что у объекта правая и верхняя грань слабо освещены, а также фон имеет в той же области тень. За счет

этого контраст фона и объекта уменьшается, и он хуже выделяется на карте глубины.

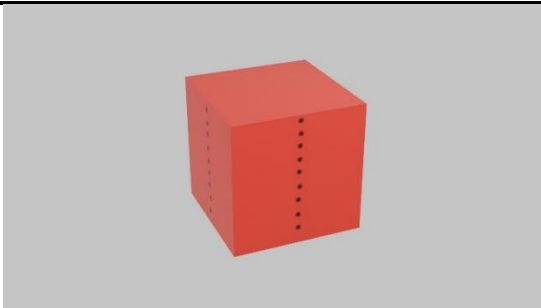
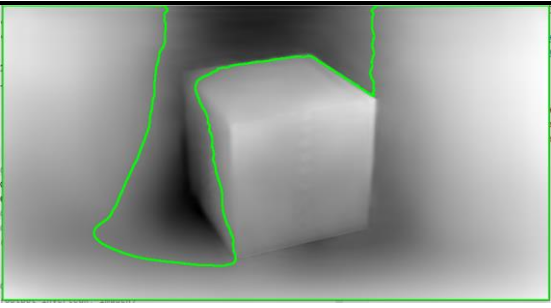
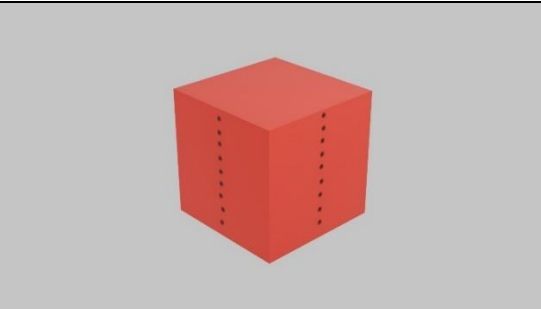
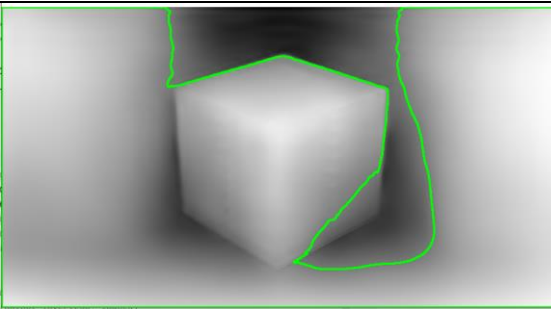
4.5.2 Эксперименты с выделением объекта

Для проверки работы алгоритма выделения объекта на изображении проведены эксперименты на изображениях при идеальных и стандартных условиях.

Алгоритм выделения объекта на изображении основан на поиске градиентных и контрастных изменений цвета. В самом начале для проверки выделения объекта на изображении были взяты изображения с идеальной сценой.

Результаты экспериментов можно посмотреть в таблице 24

Таблица 24 – Эксперимент по выделению объекты на изображении при идеальных условиях


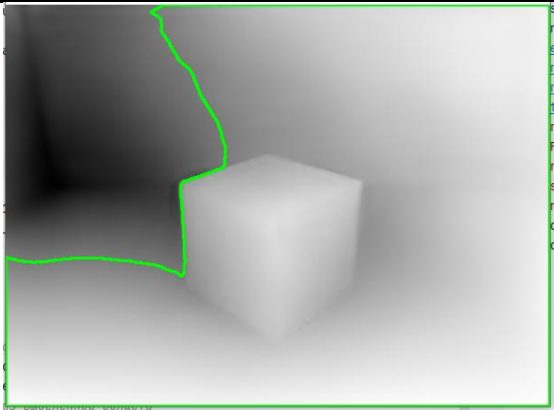
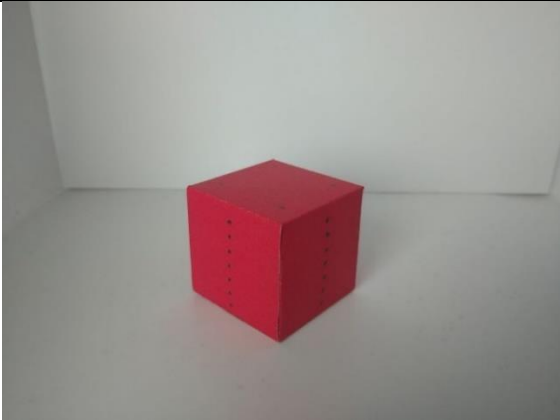
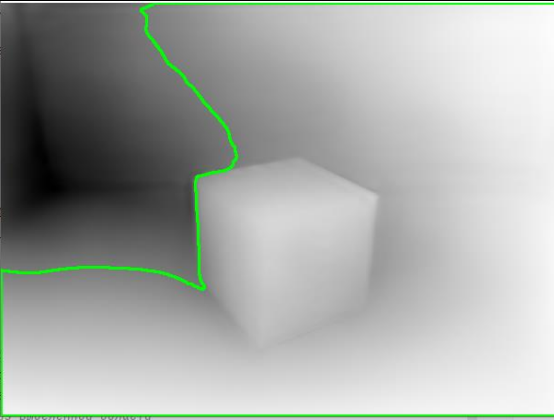
№	Изображение	Выделение
1		
2		

В обоих экспериментах таблицы 24 видно, что контур объекта выделяется за счет резкого перехода света тени. Однако из-за неравномерного освещения объекта контур выделяется неточно. Видно, что в местах, где объект и фон имеют слабый контраст, контур выделяется

некорректно, поскольку там плавный переход света и тени. В эксперименте 2 таблицы 24 контур в левой верхней части объекта идет некорректно из-за слабого контраста. По той же причине в эксперименте 1 таблиц 24 в верхней правой части контур выделяется некорректно.

Далее рассматривается работа алгоритма над изображениями со стандартными условиями.

Результаты экспериментов можно посмотреть в таблице 25.
Таблица 25 – Эксперимент по выделению объекты на изображении при стандартных условиях

№	Изображение	Выделение
1		
2		

По сравнению с предыдущим экспериментом (таблица 24) видно, что в эксперименте 1 и 2 таблицы 25 контур объекта выделяется гораздо хуже, поскольку контраст между объектом и фоном слабый. Это происходит из-за освещения сцены, а не только объекта.

Можно сделать вывод, что алгоритм эффективен, только если объект и фон имеют сильную разницу в контрасте, только в таком случае контур объекта выделиться точно.

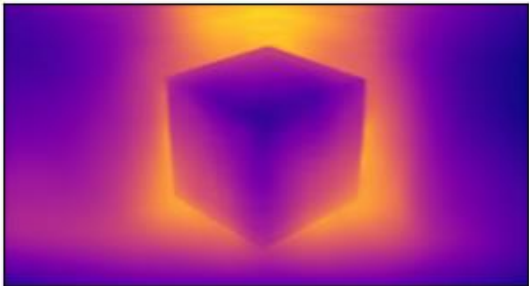
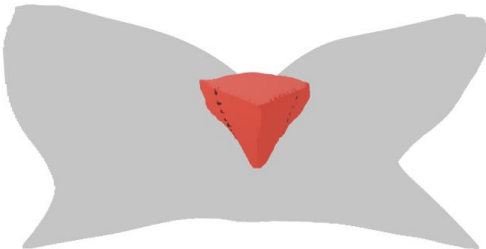
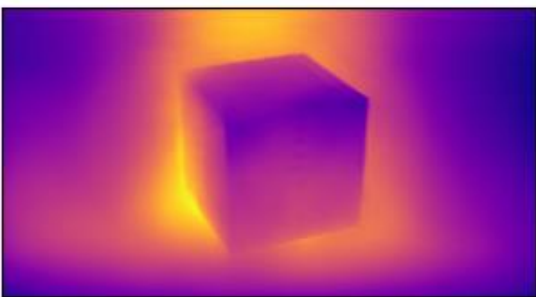
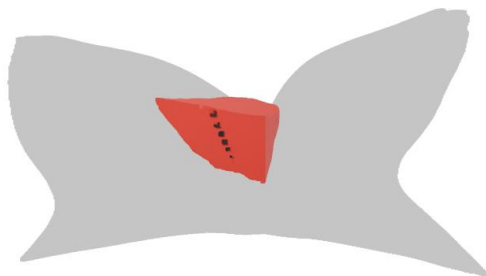
4.5.3 Эксперименты с облаком точек

Для проверки работы алгоритма по созданию облака точек проведены эксперименты с применением карт глубин для подготовленных сцен. Рассматриваются два случая, когда за основу создания карты глубины берется изображение с идеальными и стандартными условиями.

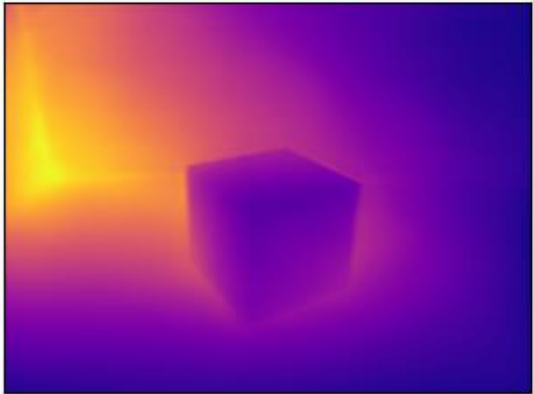
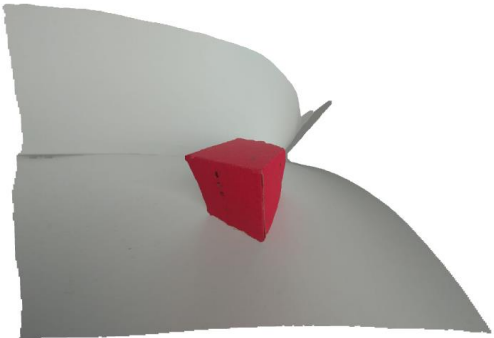
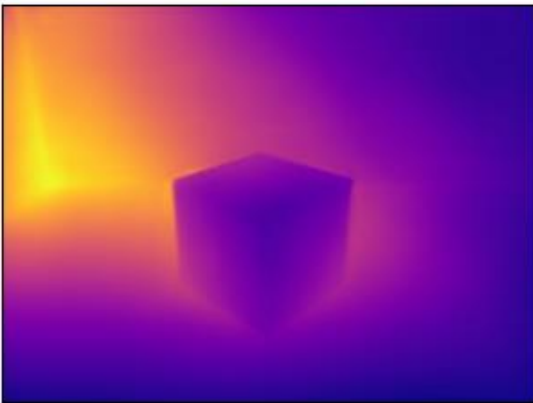
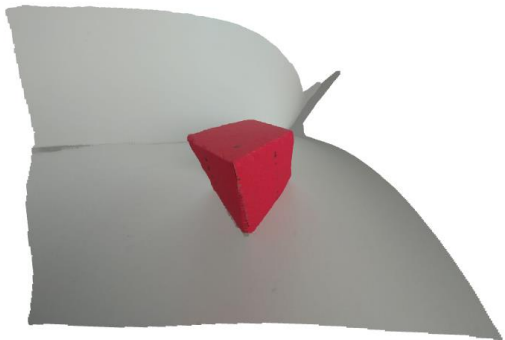
На вход подается карта глубины, по которой строится облако точек.

Результаты экспериментов можно увидеть в таблице 26.

Таблица 26 – Эксперимент по созданию облака точек при идеальных и стандартных условиях

№	Карта глубины	Облако точек
	Идеальные условия	
1		
2		

Окончание таблицы 26

№	Карта глубины	Облако точек
Стандартные условия		
3		
4		

На основании карты глубины создается облако точек. Карта глубины содержит значение глубины каждой точки изображения. Облако точек строится на основании этих значений.

В первом и втором эксперименте объект сильно искажен. Это происходит из-за отсутствия заднего фона. При сравнении экспериментов 1 и 3 можно заметить, что благодаря фону точки объекта эксперимента 3 в пространстве определились более корректно, чем в эксперименте 1. В случае со светлыми участками, карты глубины, не получается детально распознать глубину, в связи с этим появляется искажение в облаке точек.

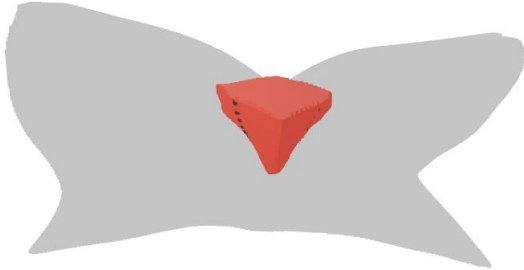
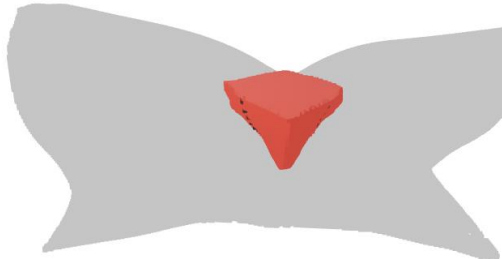
4.5.4 Эксперименты с фильтрацией облака точек

Для проверки работы алгоритма фильтрации облака точек проведены эксперименты на разных облаках точек. В экспериментах используется фильтрация методом группировки точек. Рассматриваются два случая, когда за основу берется изображение с идеальными и стандартными условиями.

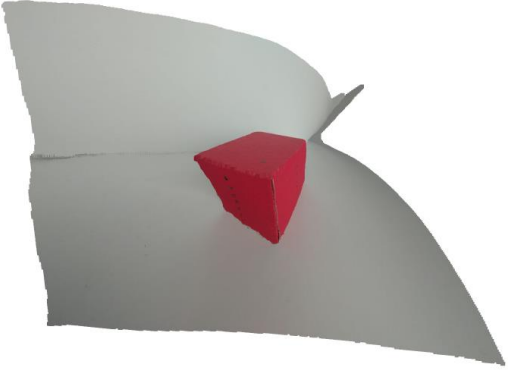
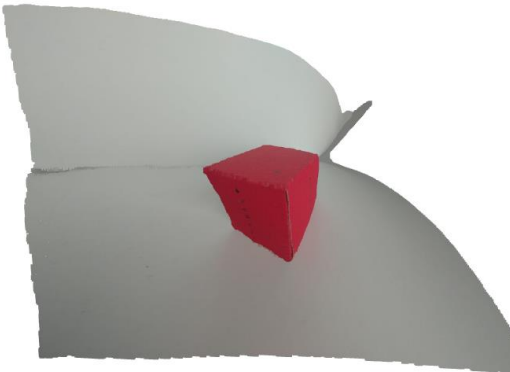
Алгоритму на вход подается облако точек. Для фильтрации точек применяется статистический метод.

Результаты экспериментов можно посмотреть в таблице 27.

Таблица 27 – Эксперимент по фильтрации облака точек при стандартных и идеальных условиях

№	Облако точек	Отфильтрованное облако точек
	Идеальные условия	
1		
	Количество точек до фильтрации: 372713	
	Количество точек после фильтрации: 367621	

Окончание таблицы 27

№	Облако точек	Отфильтрованное облако точек
	Стандартные условия	
2		
	Количество точек до фильтрации: 272366	
	Количество точек после фильтрации: 269604	

Можно заметить, что отсеялось не так много точек, но если сравнивать эти два эксперимента (из таблицы 28), то можно заметить, что в первом случае отсеялось больше точек, поскольку некоторые точки сгруппировались неверно. Однако это число слишком мало, чтобы его можно было визуальное как-то увидеть, но при детальном рассмотрении можно заметить, что края облака точек стали более неровными. Это можно заметить на изображении 19.

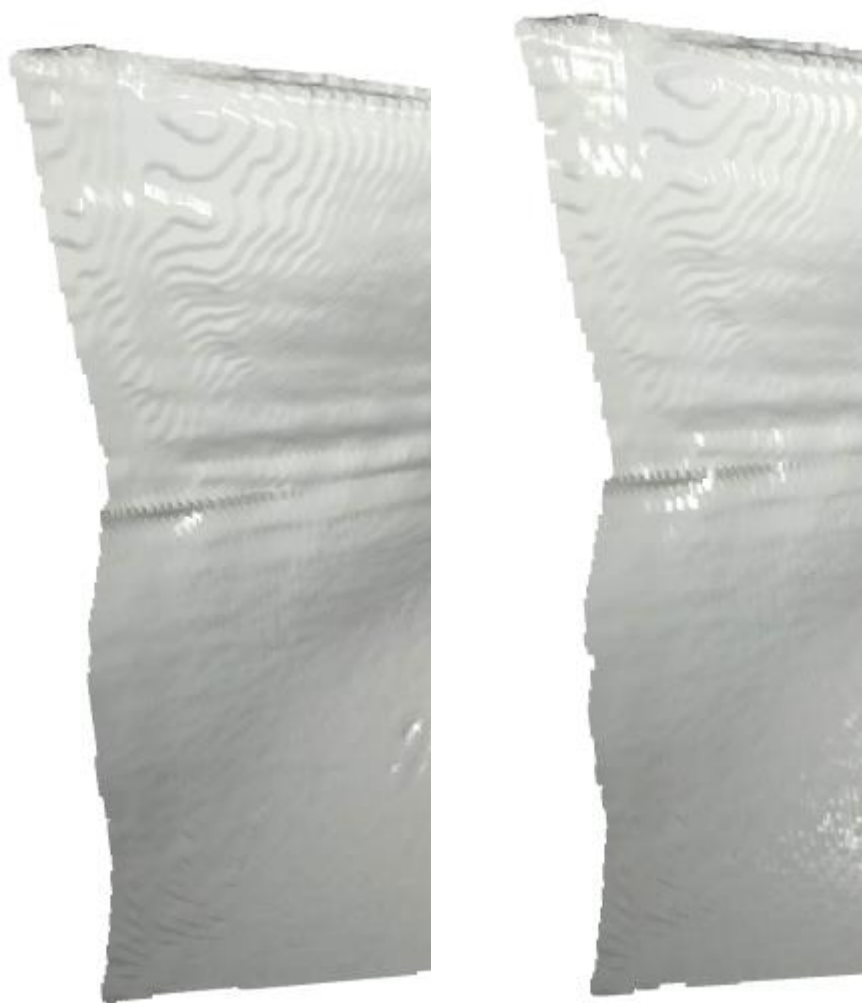


Рисунок 19 – Часть облака точек до фильтрации и после

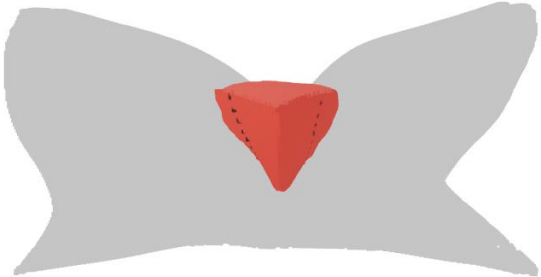
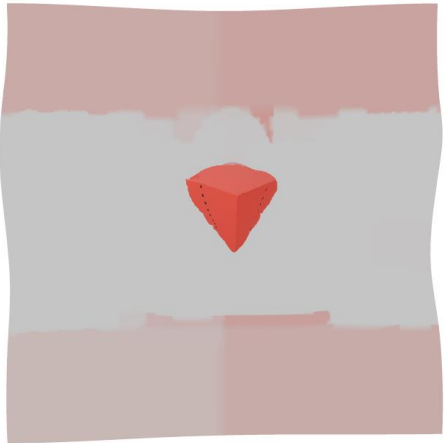
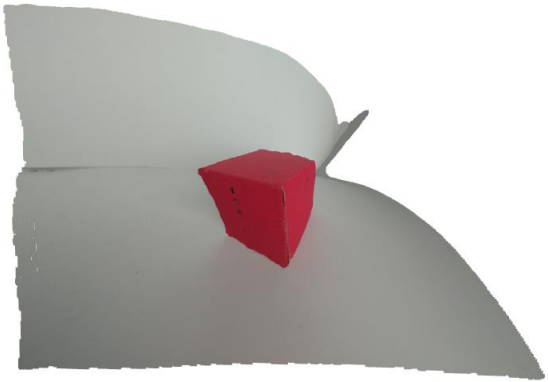
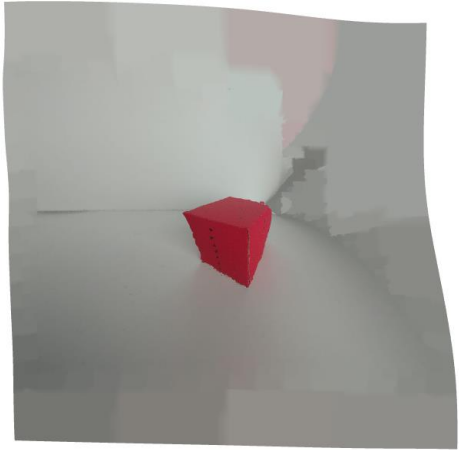
4.5.5 Эксперимент с трехмерной моделью

Для проверки работы алгоритма создания трехмерной модели проведены эксперименты на основании облака точек, которое было создано при разных условиях. Рассматриваются два случая, когда за основу берутся облака точек, построенные, в первом случае, на основании карты глубины с идеальными условиями, во втором случае стандартными условиями.

Алгоритму принимает облако точек, а на выходе получается трехмерная модель, на основании входного облака точек.

Результаты экспериментов можно посмотреть в таблице 28.

Таблица 28 – Эксперимент по созданию трехмерной модели при стандартных и идеальных условиях

№	Входное облако точек	Трехмерная модель
Идеальные условия		
1		
Время выполнения: 42.652218 секунды		
Стандартные условия		
2		
Время выполнения: 31.677923 секунд		

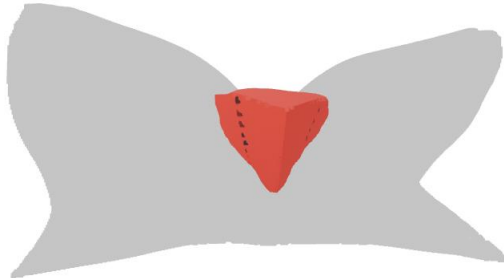
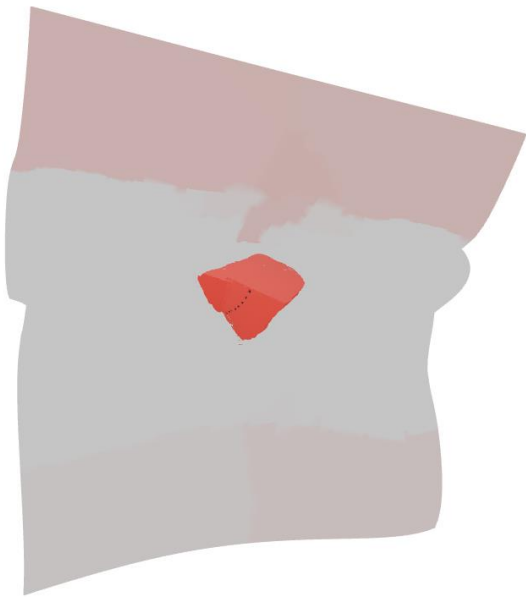
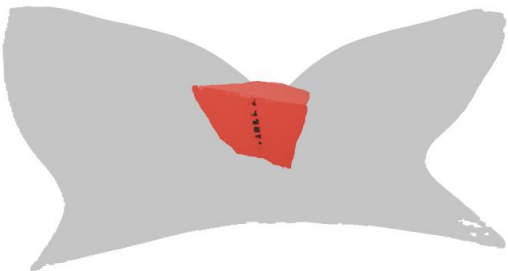

В двух экспериментах создались трехмерные модели сцен, на основании облаков точек. Можно заметить, что во втором эксперименте объект выглядит более естественно, по сравнению с первым экспериментом, поскольку облако точек сформировалось более точным, за счет теней на фоне объекта. Так же при построении трехмерной модели сцена увеличилась, поскольку меш растянулся относительно формы облака точек.

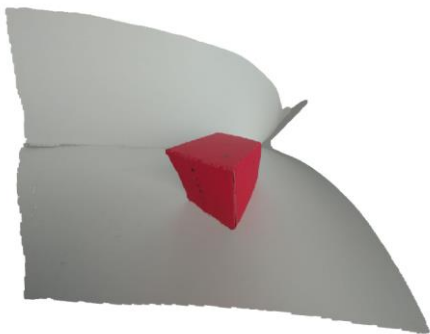
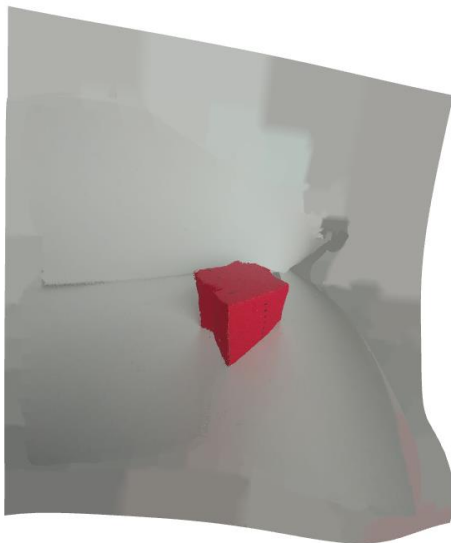
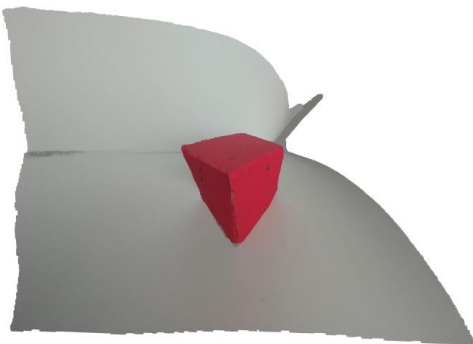
4.5.6 Эксперименты создания трехмерной модели по нескольким изображениям

Далее рассматриваются эксперименты с созданием трехмерной модели на основании нескольких изображений. Рассматриваются два случая, когда за основу создания облака точек берется изображение с идеальными условиями и стандартными условиями.

Результаты экспериментов можно посмотреть в таблице 29.

Таблица 29 – Эксперимент по созданию трехмерной модели на нескольких облаках точек при стандартных и идеальных условиях

№	Облако точек	Трехмерная модель
1	Идеальные условия	
		
		
	Время выполнения: 76.410757 секунд	

№	Облако точек	Трехмерная модель
2	Стандартные условия	
		
		
	Время выполнения: 59.997220 секунд	

При создании трехмерной модели сцены по нескольким изображениям получается грубая модель. Поскольку два облака точек смещаются на определенный угол, относительно центра сцены, который вычисляется за счет ключевых точек, и совмещаются без отсеивания повторяющихся точек и точек сцены.

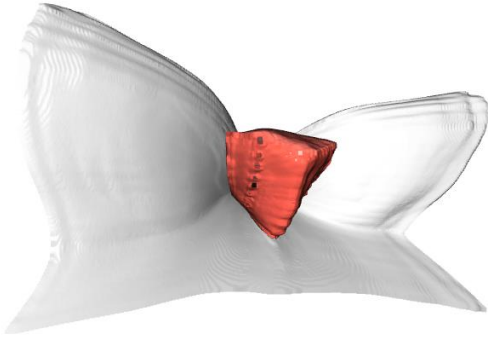

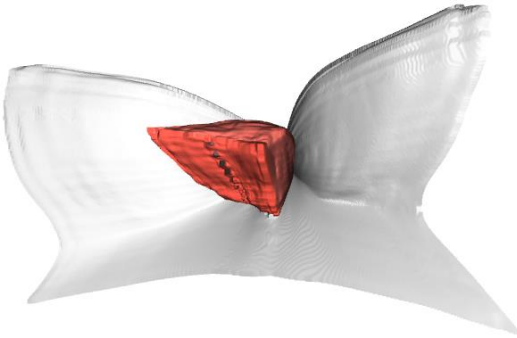
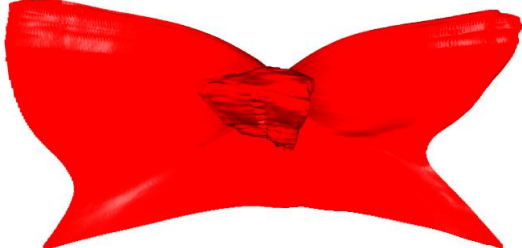
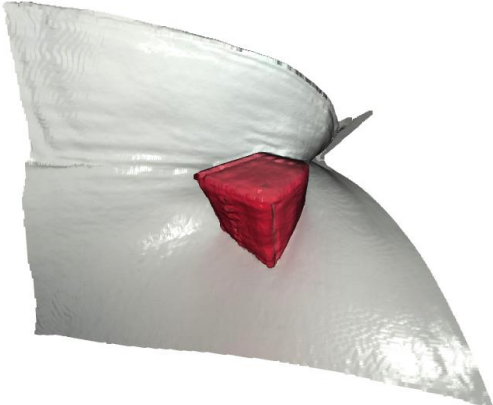
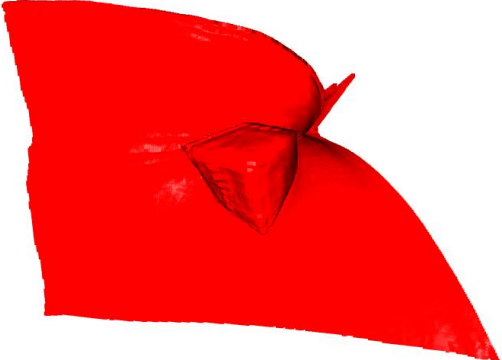
4.5.7 Дополнительный эксперимент сегментации облака точек

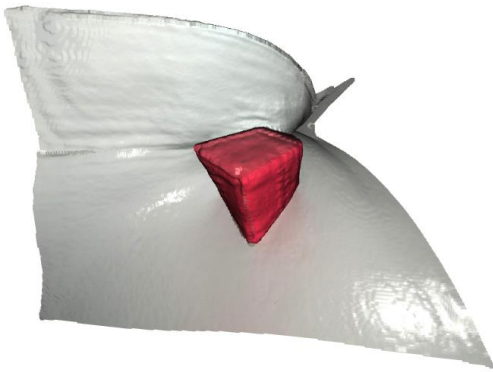
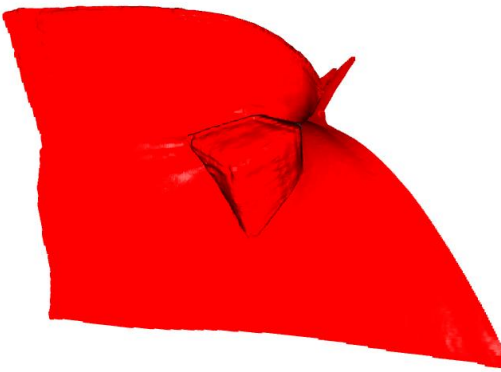
Рассматривается также дополнительный эксперимент по сегментации облака точек. Который проводится при идеальных и стандартных условиях.

При анализе результатов эксперимента необходимо учитывать, что цвет выделения объекта и плоскости должны различаться. Объект должен выделяться зеленым, а плоскость – красным.

Результаты экспериментов можно посмотреть в таблице 30.

Таблица 30 – Эксперимент по фильтрации облака точек при идеальных и стандартных условиях

№	Облако точек	Выделенная область
Идеальные условия		
1		
2		
Стандартные условия		
3		

№	Облако точек	Выделенная область
4		

Можно заметить, что алгоритм не смог определить, где находится объект, а где плоскость из-за грубой формы облака точек. Облако точек не содержит точек, которые лежали бы на одной координате. По этой причине алгоритм не может правильно выделить плоскость.

В данном случае он определил все облако точек как плоскость.

4.5.8 Краткие выводы

В рамках данной главы были проведены эксперименты для реализации и проверки алгоритма трехмерной реконструкции по стереоизображению. Были рассмотрены случаи создания карты глубины, облака точек и трехмерной модели. Дополнительно проведены эксперименты выделения объекта на изображении и облаке точек.

Заключение

В заключение выпускной квалификационной работы можно подвести итоги:

Основная цель выпускной квалификационной работы разработка моделей и методов программной системы по трехмерной реконструкции по стереоизображениям была выполнена.

Также в ходе работы над выпускной квалификационной работой были выполнены следующие задачи:

1. Анализ предметной области «Трехмерная реконструкция подводного трубопровода по стереопаре»;
2. Разработка модели предметной области «Трехмерная реконструкция подводного трубопровода по стереопаре».
3. Разработка проекта программной системы для трехмерной реконструкции с использованием стереокамеры.
4. Разработка модулей программной системы для трехмерной реконструкции с использованием стереокамеры.

В рамках работы был проведен анализ предметной области «Трехмерная реконструкция подводного трубопровода по стереопаре», разработаны программные модули, такие как:

1. подсистема «Создания карты глубины»;
2. подсистема «Создания облака точек»;
3. подсистема «Создания трехмерной модели».

Подводя итоги проведенной работы, можно сделать вывод, что существующие на данный момент программные системы не предназначены для повсеместного проведения инспекции подводных трубопроводов с использованием стереокамеры, установленной на АНПА, поскольку у большинства из них закрытый исходный код, что затрудняет отлаживание работы алгоритмов. Также в рамках выпускной квалификационной работы была предпринята попытка разработки программной системы, позволяющей

реконструировать 3D модель реальных объектов на основании стереопары. Результаты экспериментов показали зависимость распознавания объектов на снимках от ряда критериев, таких как: цветовой градиент, контрастность, пороговые значения света и тени. В рамках дальнейшей разработки возможна работа над увеличением детальности построения реконструированной 3D модели, а также работа над уточнением критериев по базовой идентификации объектов на снимках.

Список литературы

1. Автономный необитаемый подводный аппарат. — Текст: электронный // Рувики: [сайт]. — URL: https://ru.ruwiki.ru/wiki/Автономный_необитаемый_подводный_аппарат (дата обращения: 28.05.2023)
2. Алгоритм Хафа для обнаружения произвольных на изображениях. — Текст: электронный // Хабр: [сайт]. — URL: <https://habr.com/ru/articles/102948/> (дата обращения: 28.05.2023)
3. Асеев, Д. А. Современные технологии построения карт глубины / Д. А. Асеев, А. А. Бочаров, В. В. Чаплыгин // Медико-экологические информационные технологии - 2020 : сборник научных статей по материалам XXIII Международной научно-технической конференции: в 2 ч., Курск, 20–22 мая 2020 года. Том Часть 2. – Курск: Юго-Западный государственный университет, 2020. – С. 177-184.
4. Бобков, В. А. Построение карт глубин на множественной последовательности видов / В. А. Бобков, Ю. И. Роньшин, В. Ю. Машенцев // Информационные технологии. – 2012. – № 6. – С. 39-44.
5. Бобков В. А. Реализация алгоритма построения карт глубин на графических процессорах / В. А. Бобков, Ю. И. Роньшин // Информационные технологии. – 2011. – № 8. – С. 17-22.
6. Бобков В.А., Морозов М.А., Инзарцев А.В., Елисеенко Г.Д. Метод визуальной навигации для организации инспекции подводного трубопровода автономным роботом. – Морские интеллектуальные технологии. — Текст: электронный // Elibrary:[сайт]. — URL: https://www.elibrary.ru/download/elibrary_39950767_49389779.pdf – Дата публикации: 2019 г
7. Бобырь М. В. Метод расчета карты глубин на основе мягких операторов / М. В. Бобырь, А. Е. Архипов, Н. А. Милостная // Системы и средства информатики. – 2019. – Т. 29, № 2. – С. 71-84.

8. Восстановление карты глубины на основе комбинированной обработки многоканального изображения / В. В. Воронин, А. В. Фисунов, В. И. Марчук, И. С. Свирин, С. П. Петров // Современные проблемы науки и образования. – 2014. – № 6. – С. 210.

9. Герасимов В.А., Павин А.М., Телешев А.В. Устройство для обнаружения и отслеживания металлосодержащего протяженного подводного объекта с борта автономного необитаемого подводного аппарата. – Федеральное государственное бюджетное учреждение науки Институт проблем морских технологий Дальневосточного отделения Российской академии наук (ИПМТ ДВО РАН), г. Владивосток. — Текст: электронный // Elibrary:[сайт]. — URL: https://www.elibrary.ru/download/elibrary_38159556_29091164.PDF — Дата публикации: 2018 г

10. Григорашвили Ю.Е., Иваненко В.В., Стицей Ю.В. Способ определения пространственного положения трубопровода на участке подводного перехода. – Общество с ограниченной ответственностью "Научно-производственное предприятие "Техносфера-МЛ", Общество с ограниченной ответственностью "Технические Идеи Новых Технологий", г. Москва — Текст: электронный // Elibrary:[сайт]. — URL: https://www.elibrary.ru/download/elibrary_50007000_69501626.PDF — Дата публикации: 26.12.2022 г

11. Дилла Д.С., Пустовалов Е.В., Федорец А.Н., Плотников В.С. Программный комплекс для фильтрации изображений методом кросс-корреляции с составным ядром (кросскорр-состав). – Федеральное государственное автономное образовательное учреждение высшего образования «Дальневосточный федеральный университет», г. Владивосток. — Текст: электронный // Elibrary:[сайт]. — URL: https://www.elibrary.ru/download/elibrary_67984237_68427712.PDF — Дата публикации: 2024 г

12. Жарких А. А. Анализ и синтез контурных изображений на основе вейвлет-преобразования Хаара / А. А. Жарких, А. С. Юрко // Математические методы распознавания образов. – 2005. – Т. 12, № 1. – С. 318-321.

13. Жиляков П.В., Фатеев С.И., Крамарь О.А. Программный модуль обнаружения подводных объектов с заданными характеристиками при помощи подводной системы технического стереозрения. – Федеральное государственное автономное образовательное учреждение высшего образования «Севастопольский государственный университет», г. Севастополь — Текст: электронный // Elibrary:[сайт]. — URL: https://www.elibrary.ru/download/elibrary_49776353_45915353.PDF — Дата публикации: 13.10.2022 г

14. Зачесов А. А. Построение карты глубины и многоакурсного видео по стереопаре / А. А. Зачесов, М. Б. Колинченко, Д. С. Ватолин // Мир техники кино. – 2012. – Т. 6, № 1(23). – С. 19-21.

15. Изучение технических основ TriposSR. — Текст: электронный // TriposRAI: [сайт]. — URL: <https://triposrai.com/posts/exploring-the-technical-underpinnings-of-triposr/> (дата обращения: 28.05.2024)

16. Как сверточные нейронные сети имитируют работу мозга. — Текст: электронный // Сверточные нейронные сети: [сайт]. — URL: <https://practicum.yandex.ru/blog/svertochnye-neyronnye-seti/> (дата обращения: 28.05.2023)

17. Карта глубины. — Текст: электронный //ИТМО: [сайт]. — URL: https://neerc.ifmo.ru/wiki/index.php?title=Карта_глубины (дата обращения: 28.05.2023)

18. Кирпичников А.П. Трёхмерная реконструкция сцены по нескольким изображениям / А. П. Кирпичников, И. И. Шамсутдинов, М. П. Шлеймович // Вестник казанского технологического университета. – 2014. – Т. 17, № 11. – С. 229-232.

19. Котюжанский Л. А. Вычисление карты глубины стереоизображения на графическом процессоре в реальном времени //Фундаментальные исследования. – 2012. – Т. 2. – №. 6.

20. Метод трехмерной реконструкции сцены в относительных координатах по двум изображениям с неоткалиброванных видеокамер / К. М. Шепилова, А. В. Сотников, А. В. Шипатов, Ю. В. Савченко // Известия высших учебных заведений. Электроника. – 2020. – Т. 25, № 3. – С. 265-276.

21. Модельное решение задачи автоматической инспекции подводных трубопроводов с помощью гидролокатора бокового обзора / А. В. Багницкий, А. В. Инзарцев, А. М. Павин, С. В. Мельман, М. А. Морозов // Подводные исследования и робототехника. – 2011. – № 1(11). – С. 17-23.

22. Об эффективности алгоритма субполосного выделения контуров на изображении / Е. Г. Жиляков, А. А. Черноморец, В. А. Голощапова, А. Н. Заливин // Научные ведомости Белгородского государственного университета. Серия: Экономика. Информатика. – 2013. – № 15(158). – С. 128-133.

23. Пименова М. Б. Распознавание характерных объектов на местности с использованием метода SURF / М. Б. Пименова // Политехнический молодежный журнал. – 2019. – № 10(39). – С. 3.

24. Преобразование 2D JPG файла изображения в 3D OBJ модель. — Текст: электронный //3DPEA: [сайт]. — URL: <https://www.3dpea.com/ru/convert/JPG-to-OBJ> (дата обращения: 28.05.2023)

25. Редько А. В. Применение методов фильтрации в задаче построения поверхности по облаку точек / А. В. Редько, С. А. Глебов // Электронный журнал: наука, техника и образование. – 2017. – № СВ2(13). – С. 92-97.

26. Реконструкция 3D-объекта по фото в программе Meshroom. — Текст: электронный // Mapper720: [сайт]. — URL: <https://mapper720.ru/tutorials/meshroom.html> (дата обращения: 28.05.2024)

27. Ректификация изображений. — Текст: электронный //Дзен: [сайт]. — URL: <https://dzen.ru/a/Y9LWMhOPVxu9NmH8> (дата обращения: 28.05.2023)
28. Синицын С. И. Гибридный рекурсивноинкрементный алгоритм построения триангуляции Делоне // Материалы конф. «Graphicon». Н. Новгород, 2006. С. 175–177.
29. Стереοизображение. — Текст: электронный //Рувики: [сайт]. — URL: <https://ru.ruwiki.ru/wiki/Стереοизображение> (дата обращения: 28.05.2023)
30. Стереоскопический фотоаппарат. — Текст: электронный //Рувики: [сайт]. — URL: https://ru.ruwiki.ru/wiki/Стереоскопический_фотоаппарат (дата обращения: 28.05.2023)
31. Стереоскопическое зрение. — Текст: электронный //Рувики: [сайт]. — URL: https://ru.ruwiki.ru/wiki/Стереоскопическое_зрение (дата обращения: 28.05.2023)
32. Топ (поверхность). — Текст: электронный //Рувики: [сайт]. — URL: [https://ru.ruwiki.ru/wiki/Топ_\(поверхность\)](https://ru.ruwiki.ru/wiki/Топ_(поверхность)) (дата обращения: 28.05.2023)
33. Тупицын И. В. Реконструкция трехмерной модели объекта на основе стереопары при решении задач 3D-моделирования / И. В. Тупицын // Вестник Сибирского государственного аэрокосмического университета им. академика М.Ф. Решетнева. – 2011. – № 3(36). – С. 88-92.
34. Фахми Ш. С. Выделение контуров морских объектов на основе пирамидально-рекурсивного метода представления изображений / Ш. С. Фахми, Н. В. Шаталова, М. С. Крюкова // Морские интеллектуальные технологии. – 2019. – № 2-2(44). – С. 129-136.
35. Федосеев А. А. Детектирование изображения алгоритмом особых точек SIFT / А. А. Федосеев, Е. Э. Гаязова // Ученые заметки ТОГУ. – 2020. – Т. 11, № 1. – С. 224-228.

36. Что такое Python? Краткое содержание. — Текст: электронный // Python: [сайт]. — URL: <https://www.python.org/doc/essays/blurp/> (дата обращения: 28.05.2024)
37. Brief: Binary Robust Independent Elementary Features. — Текст: электронный // EPFL: [сайт]. — URL: <https://www.epfl.ch/labs/cvlab/research/descriptors-and-keypoints/research-detect-brief/> (дата обращения: 28.05.2023)
38. Canny J. A computational approach to edge detection // IEEE Transactions on pattern analysis and machine intelligence. — 1986. — №. 6. — С. 679-698.
39. Daisy: a fast local descriptor for dense matching. — Текст: электронный // EPFL: [сайт]. — URL: <https://www.epfl.ch/labs/cvlab/software/descriptors-and-keypoints/daisy/> (дата обращения: 28.05.2023)
40. Harris C. A combined corner and edge detector / С. Harris, М. Stephens // Исследовательский центр Плесси, Рок-Мэнор, Великобритания. — 1988.
41. Hussmann S., Liepert T. Robot vision system based on a 3D-ToF camera // 2007 IEEE Instrumentation & Measurement Technology Conference IMTC 2007. — IEEE, 2007. — С. 1-5.
42. Lindner M., Kolb A. Calibration of the intensity-related distance error of the PMD ToF-camera // Intelligent Robots and Computer Vision XXV: Algorithms, Techniques, and Active Vision. — SPIE, 2007. — Т. 6764. — С. 338-345.
43. Massa F., Aubry M., Marlet R. Convolutional Neural Networks for joint object detection and pose estimation: A comparative study. // Computer Vision and Pattern Recognition(CVPR). — 2015. — № 4. — С. 231–239.
44. OpenCV. — Текст: электронный // OpenCV: [сайт]. — URL: <https://opencv.org/> (дата обращения: 28.05.2023)
45. Open3D. — Текст: электронный // Open3D: [сайт]. — URL: <https://pyri.org/project/open3d/> (дата обращения: 28.05.2023)

46. Rapp H. Experimental and theoretical investigation of correlating TOF-camera systems / H. Rapp, M. Frank, F. A. Hamprecht, B. Jähne // A Theoretical and Experimental Investigation of the Systematic Errors and Statistical Uncertainties of Time-of-Flight Cameras. – 2007.

47. Shaaban Z. An Intelligent License Plate Recognition System. // International Journal of Computer Science and Network Security (IJCSNS). 2011. Vol. 11. No. 7. P. 55–61.

48. Underwater Pipeline Inspection Using Guided Waves. — Текст: электронный // ResearchGate: [сайт]. — URL: https://www.researchgate.net/publication/245364714_Underwater_Pipeline_Inspection_Using_Guided_Waves (дата обращения: 28.05.2024)