Федеральное агентство связи

Ордена Трудового Красного Знамени федеральное государственное бюджетное образовательное учреждение высшего образования

«МОСКОВСКИЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ СВЯЗИ И ИНФОРМАТИКИ»

Разрешаю

допустить к защите

Зав. кафедрой

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_\_\_г.

**БАКАЛАВРСКАЯ РАБОТА**

НА ТЕМУ

\_\_\_\_\_Построение 3D модели объекта по набору двумерных изображений\_\_\_\_\_\_

Студент \_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_Нгуен Тхе Ань\_\_\_\_\_\_\_

Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Яшина Марина Викторовна

Рецензент \_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Москва 2019 г.

Лист для объёмного задания

**ОТЗЫВ РУКОВОДИТЕЛЯ**

на выпускную квалификационную работу студента \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Нгуен Тхе Ань\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

на тему \_\_\_\_\_Построение 3D модели объекта по набору двумерных изображений\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**РЕЦЕНЗИЯ**

на выпускную квалификационную работу студента \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Нгуен Тхе Ань\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

на тему \_\_\_\_\_Построение 3D модели объекта по набору двумерных изображений\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**АННОТАЦИЯ**

Тема бакалаврской работы – Построение 3D модели объекта по набору двумерных изображений

Ключевыми словами бакалаврской работы являются:

Объектом исследования является …..

Предмет исследования – …….

Бакалаврская работа изложена на .. листах, включает .. рисунков, .. литературный источник, . приложение.

Во введении

В первой главе

Во второй главе

В третьей главе

Заключение

Содержание

[**Введение** 8](#_Toc10688187)

[**Глава 1. Анализ предметной области** 10](#_Toc10688188)

[**1.1.** **Понятие восстановления модели.** 10](#_Toc10688189)

[**1.2.** **Классификация методов.** 13](#_Toc10688190)

[**1.3.** **Принцип построения карты глубины.** 21](#_Toc10688191)

[**1.3.1.** **Обобщенный алгоритм построения карты глубины** 21](#_Toc10688192)

[**1.3.2.** **Ректификация** 23](#_Toc10688193)

[**1.3.3.** **Поиск сопряженных точек** 24](#_Toc10688194)

[**1.3.4.** **Формирование карты глубины** 27](#_Toc10688195)

[**1.3.5.** **Обработка карты глубины** 27](#_Toc10688196)

[**1.4.** **Анализ продуктов-аналогов на современном рынке** 28](#_Toc10688197)

[**1.5. Постановка задачи** 29](#_Toc10688198)

[**Выводы** 30](#_Toc10688199)

[**Глава 2. Проектирование разрабатываемой системы** 31](#_Toc10688200)

[**2.1. Обоснование выбора способа решения** 31](#_Toc10688201)

[**2.2. Конструкторский раздел** 32](#_Toc10688202)

[**2.3. Метод трехмерной реконструкции** 33](#_Toc10688203)

[**2.4. Поиск сопряженных точек** 33](#_Toc10688204)

[**2.5. Формирование объемной модели** 37](#_Toc10688205)

[**Выводы** 38](#_Toc10688206)

[**Глава 3. Разработка системы и тестирование модели** 39](#_Toc10688207)

[**3.1. Используемые средства разработки** 39](#_Toc10688208)

[**3.1.1. Язык программирования JavaScript (JS)** 39](#_Toc10688209)

[**3.1.2. Библиотека JQuery** 40](#_Toc10688210)

[**3.1.3 Сторонние пакеты** 41](#_Toc10688211)

[**3.2. Алгоритм обработки данных** 41](#_Toc10688212)

[**3.3.3. Дизайн пользовательского интерфейса** 44](#_Toc10688213)

[**3.3.4. Процесс разработки интерфейса системы** 45](#_Toc10688214)

[**3.3.5. Тестирование системы** 50](#_Toc10688215)

[**3.3.6. Функциональные возможности и ограничения** 52](#_Toc10688216)

[**Выводы** 53](#_Toc10688217)

[**Заключение** 54](#_Toc10688218)

[**Литература** 55](#_Toc10688219)

[**Приложение** 56](#_Toc10688220)

# **Введение**

В области распознавания образов часто рассматриваются вопросы построения объемной модели, имея реальный объект наблюдения. Для решения этих вопросов обычно используют ультразвуковые преобразователи, лазеры или машинное зрение. Использование ультразвуковых преобразователей и лазеров дают на выходе более быструю и точную информацию, они также являются дорогостоющими. Поэтому с точки зрения цен целесообразнее будет использовать стереозрение – получение трехмерной модели по набору изображений.

С момента первого появления понятия системы компьютерного зрения систему активно развивают, и к настоящему времени решены большое количество задач. Кроме того накоплено большое количество практических и теоретических материалов, что позволяет облегчить разработку новых систем компьютерного зрения.

Однако, известные решения в особых случая оказываются недостаточно эффективными из-за высоких вычислительных затрат для получения точной детализированной модели. С другой стороны, бюджетные решения не всегда могут обеспечивать необходимую точность.

Основной задачей для получения трехмерной модели является построение карты глубины (depth map). Для решения такой задачи используют две или более камер, направленные на сцену. Карта глубины – изображение, где вместо значений интенсивности (яркостей) пикселей хранятся расстояния до объекта.

При наличии карты глубины можно воспроизвести трехмерные модели ландшафта или любые другие объекты, что позволяет использовать их для создания различных приложений с использованием виртуальной реальности, робототехника и т.д. Например, для робототехники, использование стереозрения и карты глубины дает возможность управлять движением роботов и распознавать человека и его действия.

Различие между машинным зрением и стереозрением заключается в том, что в машинном зрении используется всего одна камера, когда в стереозрении используются как минимум 2 камеры, повторяя в некоторых чертах особенности человеческого зрения, что позволяет получать помимо стандартной информаци такие, как цвет и интенсивность (яркость) объекта, но еще и расстояние до него. Таким образом, благодаря стереозрению появляется возможность обходиться без использования дорогостоящих оборудований, например датчиков, сонаров и лидаров, что позволяет снизить себестоимость аппаратного решения, что в ряде задач является критичным условием.

Целью данной работы является создание системы, которая находит необходимые параметры наблюдаемого объекта из карты глубины по двум изображениям, полученных с одной камеры, и отображение его на экране.

# **Глава 1. Анализ предметной области**

# **Понятие восстановления модели.**

Существуют несколько основных подходов к восстановлению модели:

* ручной - восстановление с помощью специальных программ моделирования (например 3D Max)
* автоматический – использование известных физических характеристик реальных моделей для восстановления по шаблонам
* трехмерная реконструкция – восстановление модели по изображениям объекта.

Ручной подход требует огромного количества усилий и времени, и иногда требуемый результат бывают недостижимы из-за сложности.

Автоматический подход требует точнейшего физического моделирования, в связи с чем это оказывается необычайно сложным и на данный момент это мало осуществим.

Трехмерная реконструкция – процесс построения физически точной модели (получения формы реального объекта в цифровом представлении) учитывая геометрию, положение в пространстве и т.д., и ее представление в удобной для хранения, изменения и визуализация формы. Результат трехмерной реконструкции представлен на рисунке 1.1.1

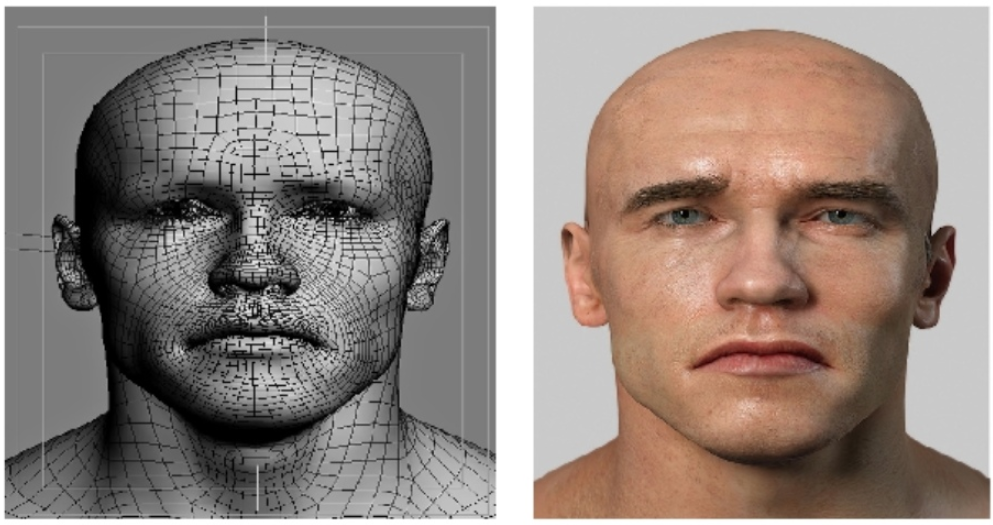


Рисунок 1.1.1 – результат трехмерной реконструкции.

Создание трехмерной модели объекта по набору двумерных изображений или трехмерная реконструкция – является одной из актуальный задач в сфере виртуальной реальности (ВР) и компьютерной графики (КГ).

Трехмерная реконструкция имеет 2 метода, связанные с обработкой данных:

* активный метод;
* пассивный метод;

Активный метод использует такие устройства, как лазерные сканеры, которые находятся в непосредственном взаимодействии с объектами сцены и сами являются ее участниками. Лазерные сканеры позволяют с высокой точностью определять расстояние от лазера до поверхности объекта. Также к такому методу относятся методы, использующие изображения объекта со специальными освещениями: при свете цветных прожекторов и прожекторов со специальными масками. Главные недостатки у данного метода заключается в высокой стоимости оборудования и невозможность восстановления динамических сцен.

Пассивный метод в качестве исходных данных использует данные, полученные с помощью фотокамеры или видеокамеры (последовательности изображений объектов) при естественном освещении или в свете обычных ламп. Данный метод считается наиболее перспективным, поскольку в модели могут участвовать люди. Основная проблема заключается в том, что изображение, полученное с камеры, получают после преобразования трехмерной сцены в двумерное изображение, в связи с чем теряется важная информация, например действительные размеры, углы и т.д.

На этапе сбора данных для получения информации о дальности для различных видов, где содержатся поверхности объекта, требуется знание поставленной задачи, требование к предъявляемой точности и детальности модели. Для простого объекта часто хватает от 1 до 6 видов, однако для сложных объектов и для обеспечения желаемой точности необходимо большее количество видов. Для таких потребностей требуется более высокая вычислительная мощность, поскольку с ростом числа видов, увеличивается и сложность метода реконструкции.

При наличии более 1 вида необходимо использовать процесс совмещение. Каждый вид имеет дальнометрическое изображение поверхности объекта и полутоновое изображение. Процесс совмещения – это процесс, при котором все дальнометрические данные от каждого вида должны быть скомбинированы путем приведения к единой трехмерной системе для получения модели поверхности объекта.

Также реконструкция бывают в двух основных видов:

* разреженная реконструкция;
* плотная реконструкция;

Разреженная реконструкция различается от плотной тем, что она имеет слабую детализацию полученного облака точек, а плотная производит качественную структуру.

Существуют два метода оценки трехмерной формы сцены при использовании трехмерной реконструкции:

* использование информации о геометрии;
* использование фотометрической информации;

Первый метод эффективен, когда производится реконструкция над относительно простыми объектами сцены. Второй метод позволяет оценивать трехмерную форму в более сложных случаях. В данной работе будут рассматриваться методы, использующие фотометрическую информацию.

# **Классификация методов.**

Существуют следующие методы трехмерной реконструкции, которые используют фотометрическую информацию:

* стереозрение;
* метод структурированного света;
* структура из движения (SFM – structure from motion);
* структура из освещенности (SFH – structure from shading);
* времяпролетные сенсоры (ToF – Time of Fligt);

Стереозрение является самым известным методом, использующий фотометрическую информацию. Механизм получения стереоизображения основан на принципах работы человеческого зрения. Благодаря бинокулярному зрению, человек может получить представление о глубине объекта. Пример проецирования точки Х на плоскости изображений xl и xr двух камер показан на рисунке 1.2.1.

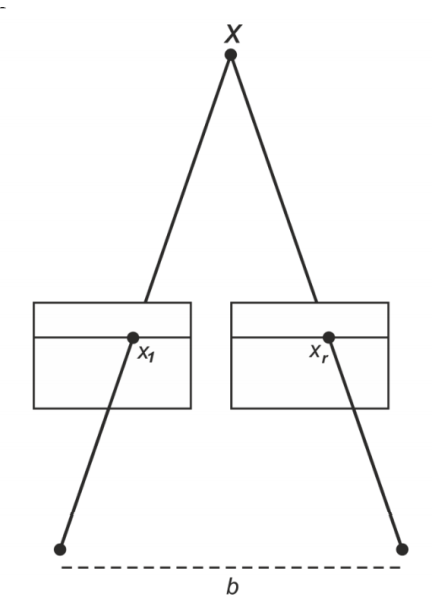


Рисунок 1.2.1 – Проецирование точки на плоскости изображений двух камер

Поскольку глаза у человека располагаются на некотором расстоянии друг от друга в горизонтальной плоскости, каждый глаз видит разную картину. При наблюдении за объектом, находящимся на расстоянии от наблюдателя, из-за углового смещения, называемого параллаксом, возникает эффект объема. Схема параллакса представлена на рисунке 1.2.2.

Данное смещение будет тем больше, чем ближе находится наблюдаемый объект от наблюдателя, и наоборот, чем дальше находится наблюдаемый объект, тем меньше смещение.

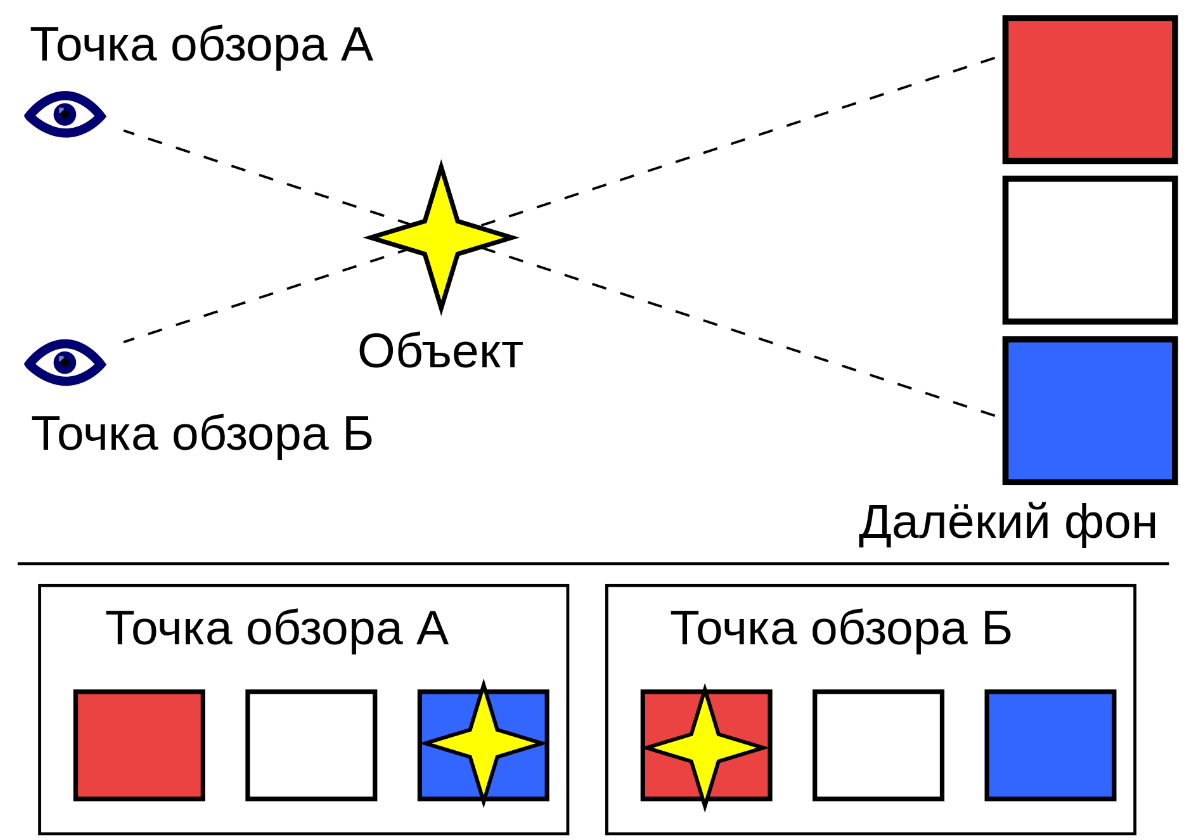


Рисунок 1.2.2 – схема параллакса.

Для реализация данного метода, необходимо получить два изображения от камеры, направленных на сцену. Операция вычисления трехмерной реконструкции по двум изображениям основываются на следующих алгоритмах и исходных предположениях об окружающем мире:

* ламбертовость поверхности – освещение поверхности не зависит от угла зрения;
* поверхности кусочно-гладкие – для избежание погрешностей;
* калибровка камер – нахождение внутренних (оптических) и внешних (положение в пространстве) параметров;
* ограничение упорядоченности – точки на левом и правом изображении должны следовать слева направо;
* выполнение очистки (ректификация) стереопары – выполнение преобразования, чтобы правое и левое изображение проецировались на плоскость, параллельную базовой линии. Изображение до и после ректификации представлены на рисунке 1.2.3 – 1.2.4;

Карта глубины – изображение, где вместо значений интенсивности (яркостей) пикселей хранятся расстояние от плоскости сенсора до объекта сцены.

Для увеличения точности и надежности системы получения карты глубины, можно увеличить количество камер. Тогда одна из камер будет базовой, относительно нее в ее системе координат будет вычисляться дальнометрическое изображение. Точка на поверхности объекта должна быть видима базовой камерой, проектором, и как минимум одной из других камер. При наличии всего одной камеры помимо базовой камеры, которая видит точку, процесс обработки сводится к обычной обработке в стереосистеме. Если точка видима двумя или более камерами помимо базовой камеры, то дополнительные изображения можно использовать для повышения надежности обработки [1].

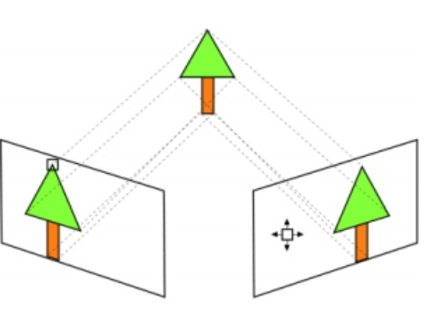


Рисунок 1.2.3 – Изображение, полученное с видео потока.

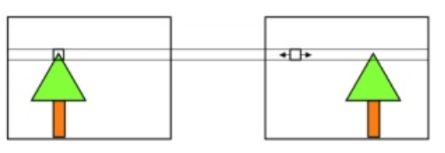


Рисунок 1.2.4 – Изображение после ректификации.

Метод структурированного света часто применяют в промышленности. Принцип работы схож с реконструкцией по стереопаре, однако вместо двух камер используется прожектор и одна камера [1]. Пример использования метода структурированного света показан на рисунках 1.2.5 и 1.2.6.

Поверхности наблюдаемых объектов освещаются через маски в виде сетки. Данный метод может реализоваться с помощью слайд-проектора, проецирующего регулярную сетку из светлых линий на поверхности объектов. Данный метод использует камеру для фиксации результата искажения сетки от формы объекта. Поскольку известна структура сетки, система имеет информацию о том, какие проектирующие лучи формируют точки изображения, соответствующие пересечениям линии сетки [1].

Недостаток данного метода является невозможность работы при наличии кроме прожектора, внешнего источника света или работа с плохо отражающими поверхностями [1].

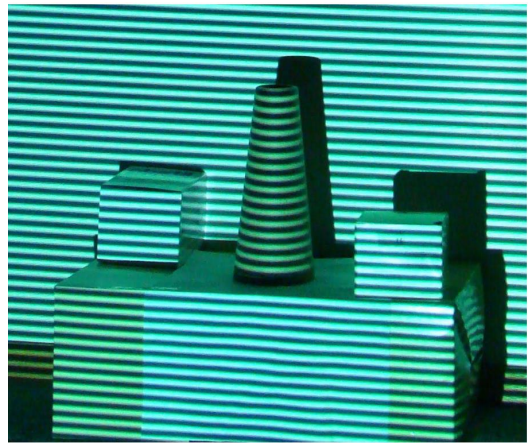


Рисунок 1.2.5 – Структурированный свет.

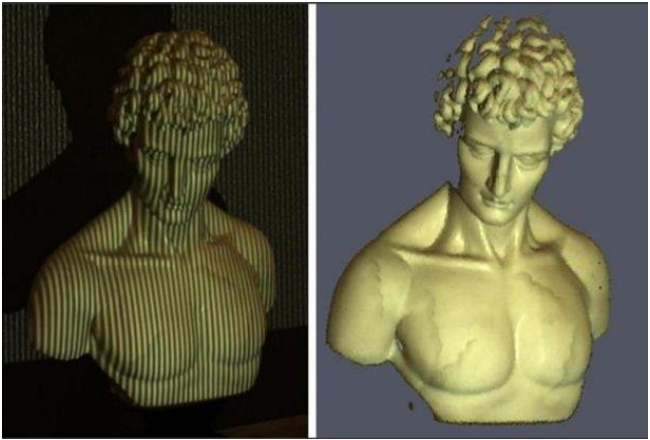


Рисунок 1.2.6 – Объемная модель

Оценка формы по движению – структура из движения (SFM) используется для нахождения трехмерной структуры объекта путем анализа локального движения частей сцены с течением времени [1]. При перемещении камеры или объекта или одновременно сразу камеры и объекта, камера регистрирует информацию из последовательности изменяющихся изображений. Данный метод имитирует человеческое зрение, восстанавливает трехмерную модель объекта по изменениям в двумерном поле зрения. Пример перемещении камер представлен на рисунке 1.2.7.

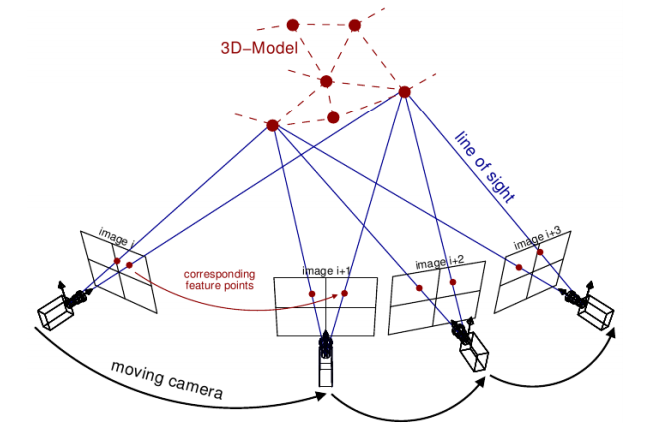


Рисунок 1.2.7 – Структура из движения

Данный метод позволяет восстанавливать поверхности и углы по соответствующим точкам на трехмерных сценах или по векторам оптического потока, а также определить траекторию движения объекта. Нахождение формы модели или структуры из движения представляет собой задачу, аналогичную задаче стереозрения. Однако у данного метода, изображения получаются в разное время. Дня нахождения соответствия используют детекторы особых точек и дескрипторы локальных областей: SHFT или SURF.

Данный метод, как и остальные также имеет недостатки. Алгоритмы, основанные на оптическом потоке, используют небольшие временные промежутки между изображениями и вычисляют плотные трехмерные структуры, а алгоритмы, основанные на соответствии характерных признаков, способны работать с большими временными промежутками, но вычисляют только разреженные трехмерные структуры.

Оценка формы по освещенности – структура из освещенности (SFS) использует свет для освещения сцены, позволяя находить расстояния путем плавно-затененных или плавно-засветленных поверхностей. Различные тени на изображениях, появляющиеся из-за различного условия освещения или ракурса содержат различную информацию о форме поверхности. Пример оценки формы по освещенности представлен на рисунке 1.2.8.

Для вычисления формы используется формула отражения закона Ламберта, которая связывает интенсивность пикселя на поверхности изображения и направление нормали к поверхности в этом месте. С помощью данной формулы, вычисляется нормаль к точке поверхности (1.2.1) по данным об интенсивности ее освещенности, полученным с изображения с поверхности [1]. Значения интенсивности пикселей используются для определения ориентации поверхности.

n = f(x,y), (1.2.1)

где n – нормаль к точке поверхности с координатой (x, y).

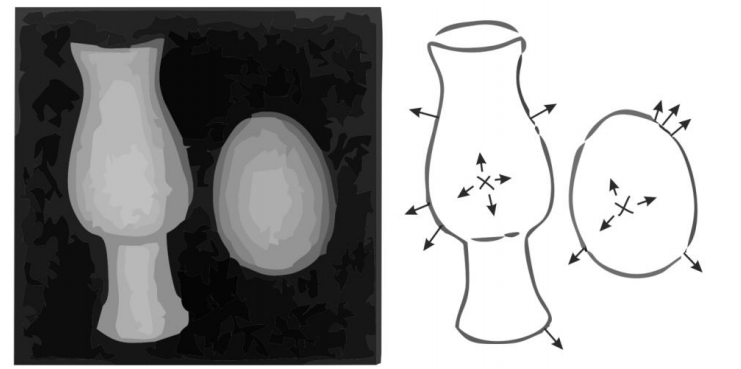
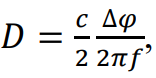


Рисунок 1.2.8 – Оценка формы по освещенности

На рисунке 1.2.8 справа показаны нормали в нескольких точках на поверхности объекта. Отмеченные крестики показывают, что данные точки на поверхности изображения, нормали к поверхности направлены прямо на источник света. А слева показано изображение объектов, которые описываются законом отражения Ламберта – интенсивность освещенности на изображении пропорциональна углу между нормалью к поверхности и направлением падающего света. Нормали к поверхности перпендикулярны и к направлению наблюдениях, и к границе поверхности на изображения в точках лимба [1]. Благодаря этим условиям накладывается ограничения на нормаль в трехмерном пространстве, что позволяет распространить векторы нормали по всей поверхности. Следовательно, можно получить частичное внутреннее изображение, а для построения карты глубины, необходимо присвоить произвольное значение одной из наиболее ярких точек и распространить значения глубины для остальных точек, учитывая изменение направлений нормалей.

К недостаткам данного метода можно отнести невозможность использования в обычных условиях, а также сложность процедуры калибровки.

Времяпролетные сенсоры (ToF) – частный случай использования метода структурированного света для получения высокоточных облаков точек. Данный метод использует лазер, чтобы получить трехмерное изображение. Суть работы заключается в том, чтобы получить время пролета лазерного луча до объекта и обратно с помощью специального сенсора. При известном значении скорости распространения импульсов и значении времени, можно найти расстояние с помощью формулы (1.2.2). Принцип работы метода показан на рисунке 1.2.9.

(1.2.2)

Где с – скорость распространения света,

∆ϕ – разность фаз между принятым и отправленным сигналом,

f – частота.

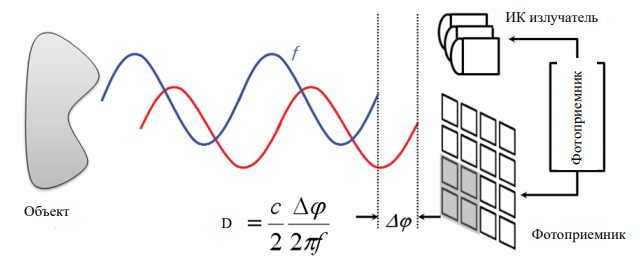


Рисунок 1.2.9 – Времяпролетные сенсоры

Недостаток данного метода заключается в том, при наличии углов и изогнутых поверхностей или зеркал возникают ошибки вследствие многократного отражения луча света. Объекты с высокой отражающей способностью могут повлиять на точность расчета расстояния.

# **Принцип построения карты глубины.**

# **Обобщенный алгоритм построения карты глубины**

В задачах получения карты глубины или карты диспаратности по двум ректифицированным изображениям в стереозрении ключевую роль играет понятие диспаратности. Диспаратность – это различие положении точки в пикселях на правом изображении относительно левого изображения. Рассмотрим рисунок 1.10. D – расстояние от плоскости положении камер до точки, P – наблюдаемая точка, p1 и p2 – это перспективные проекции на плоскость изображения, b – расстояние между камерами или база, f – фокусное расстояние камеры, С1 и С2 – расположение камер, IP1 IP2 – плоскости изображения или фотография. Следовательно, чтобы получить расстояние D используем формулу (1.3.1.1)

(1.3.1.1)

где d – диспаратность,

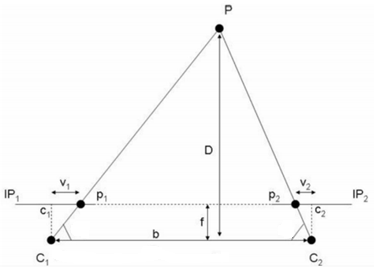


Рисунок 1.3.1.1 – Геометрическое представление расположение камер относительно объекта.

Для расчета диспаратности применяется формула (1.3.1.2).

(1.3.1.2)

где V1 – смещение точки P1 относительно центра первой камеры,

V2­ – смещение точки P2 относительно центра второй камеры.

Поскольку значения у фокусного расстояния камеры и расстояние между камерами находятся на этапе калибровки, они не меняются. Тогда диспаратность можно рассматривать как относительную глубину точек. Следует также учитывать, что диспаратность пропорциональна фокусному расстоянию, при увеличении фокусного расстояния увеличивается и изображение.

Таким образом задача построения карты глубины можно разделить на следующие этапы:

* Ректификация стереопары;
* Поиск сопряженных точек и сопоставление изображений;
* Формирование карты глубины;
* Обработка карты глубины;

# **Ректификация**

При идеальных условиях положения двух камер должны быть фиксированы и жестко связаны между собой так, чтобы их оптические оси были параллельны. Тогда задача реконструкции решается очень просто. Однако на практике это бывает очень редко. Две камеры должны быть выставлены идеальным образом, чтобы оптические оси камер сходились друг к другу в области расположения элементов сцены, а их поле зрения попадали в одни и те же элементы сцены. После получения изображений от двух камер, они должны находиться в единой прямоугольной системе координат с приведением изображения к эпиполярной стереопаре.

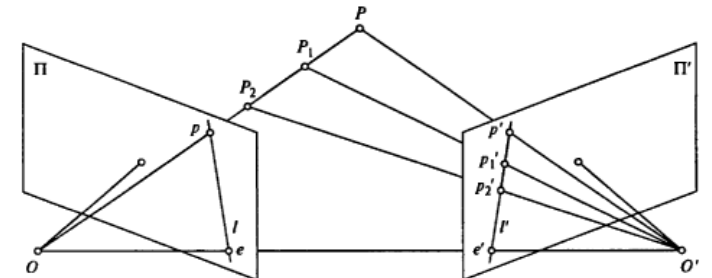


Рисунок 1.3.2.1 – Модель стереопары

Для эффективного поиска диспаратности, необходимо чтобы объект изображался на эпиполярной линии на левом и правом снимках. Для этого примеряют ректификацию, приводящее стереопару к эпиполярной. Эпиполярные линии в таком случае от двух снимков становятся параллельными одной из ее осей, а расчет диспартаности будет производиться вдоль одной оси.

# **Поиск сопряженных точек**

При некоторых обстоятельствах точка поверхности объекта может оказаться за пределами изображения. В случае, если точка поверхности объекта видна на обоих изображениях, то это превращается в задачу нахождения и установления соответствии двух точек на двух изображениях, являющимися проекциями одного и того же объекта и лежащие на одной эпиполярной линии.

Существует множество подходов поиска соответствующих точек. Одни методы предлагают анализировать изображения по отдельности, выделяя на них особые точки. Другие методы предлагают анализировать совместно.

Принята также различать два широких класса алгоритмов для решения задачи на соответствие:

* глобальные;
* локальные;

Глобальный алгоритм – это поиск наилучшей карты диспаратности для всего изображения. Данный алгоритм дает хорошие результаты в поиске карты диспаратности с минимальными количествами ошибок.

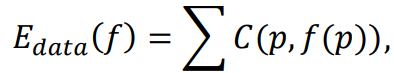
Данная задача решается путем минимизации функционала. Основная задача состоит в поиске наилучшей конфигурации f, минимизирующий функционал E формула (1.3.3.1).

(1.3.3.1)

где Edata(f) – оценка конфигурации,

Esmooth(f) – штраф за конфигурацию, в случае, если она нарушает условие непрерывности диспаратности,

λ – коэффициент для баланса между слагаемыми.

Для расчета Edata(f) используется попиксельный расчет счетов с помощью метрик из локальных алгоритмов (1.3.3.2).

(1.3.3.2)

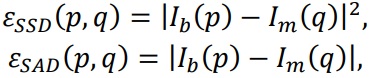
Для расчета Esmooth(f) учитываются соседние пиксели, к которым применяются монотонно-убывающая функция, снижая счета для областей, где меняется значительно значение при переходе от одного пикселя к другому пикселю. Коэффициент λ необходим для уменьшения степени влияния Esmooth на результат минимизации.

Недосток у глобального алгоритма, это большие вычислительные затраты, время расчета и сложность реализации в режиме real-time.

Локальный алгоритм – в отличие от глобального алгоритма, подсчитывает диспаратность каждого пикселя по отдельности. Алгоритм считается популярным, поскольку можно быстро получить карту диспаратности, однако уменьшается точность при построении карты глубины.

Алгоритм подразумевает использование окна фиксированного или адаптируемого размера, который выделяет область на левом изображении считающимся опорным. С помощью корреляции ищут данную область на правом изображении считающимся парным. Для стандартного подхода, области имеют прямоугольные окна фиксированного размера, определяемого опытным путем. Для определения схожести между двумя областями используется счета соответствия. Большинство алгоритмов используют всего 1 цветовой канал, который принимает значения в диапазоне от 0 до 255. Суммирование производится вокруг пикселя, который находится в центре окна.

Существуют различные способы использования метрики соответствия и методы суммирования счетов для локального алгоритма, из-за размера окрестности соседних пикселей, влияющих на результат. Для базового изображения выбирается произвольный пиксель, и вокруг нее строится окно. Такое же окно строится и на парном изображении для ряда пикселей с подозрительными соответствиями выбранному пикселю, впоследствии производится расчет счет соответствия [2].

Для расчета счетов часто применяют формулу (1.3.3.3).

(1.3.3.3)

где p – пиксель базового изображения,

q = e(p, d) – пиксель парного изображения,

d – значение диспаратности,

Ib(p) и Im(q) – яркости пикселов p и q,

Функция e(p, d) обозначает эпиполярную линию на парном изображении для пикселя p базового изображения с параметром d.

После подсчета стоимости соответствия для каждого пикселя в базовом изображении и для каждой возможной диспаратности от 0 до максимального значения диспаратности, производится суммирование стоимостей. Данный подход производит усреднение стоимостей соответствия в некоторой области с помощью свертки, чаще всего с ядром Гаусса.

# **Формирование карты глубины**

Формирование карты глубины производится за счет карты диспаратности или карты смещений. Значение диспаратности получается от пары сопряженных точек на двух изображениях. Значение диспаратности является таким, при котором достигается минимальное значение счета для пары q и p для заданного пикселя p базового изображения. Карта диспаратности приведена на рисунке 1.3.4.1.



Рисунок 1.3.4.1 – Карта диспаратности.

# **Обработка карты глубины**

При построении карты глубины, недостатки большинства методов – это плохо определяется значение диспаратности в местах медленного изменения интенсивности. При работе со сложными сценами могут возникать проблемы в случае применении алгоритмов стереозрения, например применение высоко-оптимизированных алгоритмов, предназначенных для обработки в режиме real-rime на встраиваемых процессорах. Также могут возникать проблемы при построении карты глубины для малодетальных объектов. Поскольку большие области на изображении объекта содержат приблизительно одинаковые значения интенсивности для всех пикселей.

Для обработки карты глубины с зашумленными или разреженными значениями применяют процедуры фильтрации и интерполяции. Благодаря таким процедурам позволяют получить более точные значения при помощи распространении значения дальностей с участков с высокой достоверностью на участках с низкой достоверностью.

# **Анализ продуктов-аналогов на современном рынке**

**RealityCapture** – это ПО, предназначенное для реконструкции трехмерной модели. Данная программа позволяет создавать трехмерные модели из набора фотографий и лазерных сканирований без швов. Программа включает в себя следующие функции: масштабирование, фильтрацию, автоматическую калибровку, раскраску, сглаживание и т.д. RealityCapture был разработан словацкой компанией Capturing Reality в 2016 году. В качестве входных данных могут участвовать: изображения, видео.

Данная программа поддерживается только в операционной системе Windows и имеет один английский язык. Для работы с этой программой необходимо иметь мощный процессор, большую оперативную память, 64 битную версию Microsoft Windows 8/8.1/10. При отсутствии одного или нескольких пунктов, программа либо не сможет работать, либо будет иметь ограниченную функциональность.

В зависимости от функциональности, данная программа будет стоить от 100 до 15000 €.

**Pix4Dmapper –** программное обеспечение, использующее фотограмметрию и алгоритмы компьютерного зрения для получения трехмерной модели из набора неупорядоченных изображений. В качестве входных данных могут быть различного рода изображения, например тепловые изображения, мультиспектральные изображения, DSLR, “рыбий глаз”, RGB, и видео. Pix4Dmapper был разработан швейцарской компанией Pix4D в 2011 году.

Данная программа не нуждается в значительной мощности устройства, так как она может работать на мобильных, настольных и облачных платформах. Программа поддерживается на таких ОС, как Linux, MacOS, Windows. Имеет возможность переключаться между десятью языками включая русский и английский языки.

Новые пользователи имеют 25 дней бесплатного пользования. После для работы необходимо оплачивать каждый месяц по 350$.

[**PhotoModeler**](https://en.wikipedia.org/wiki/PhotoModeler) – программное обеспечение, которое получает трехмерную модель на основе изображений и стереофотограмметрию с близкого расстояния. Компания Eos System Inc разработала программу в 1994г и поддерживает по сей день.

Фотограмметрия с близкого расстояния понимается, как снятые с земли ручной камерой или с использованием дрона на небольшом расстоянии.

Поддерживается только операционная система Windows. Цена на рынке может варьироваться от 995 до 2995$ в зависимости от назначения или 49$ в месяц.

# **1.5. Постановка задачи**

Основываясь на проведенном анализе предметной области, цель данной задачи можно сформулировать следующим образом:

Разработать систему для построения объемной модели по двум изображениям. Данная система должна обеспечивать хорошую точность построения.

Для достижения поставленной цели необходимо решить следующие задачи:

* провести обзор литературы;
* выбрать алгоритм для построения карты глубины;
* получить первое изображение;
* произвести начальные калибровки;
* получить второе изображение;
* найти параметры объекта;
* вывести трехмерную модель;

Входными данными являются:

* видеопоток;
* фокусное расстояние камеры;
* значение базы.

Выходными данными является трехмерная модель наблюдаемого объекта.

Начальные ограничение, в рамках которого будет решаться задача:

- наличие одной камеры;

# **Выводы**

Проведенные исследования существующих алгоритмов трехмерной реконструкции, а также программы, которые их используют позволили установить, что:

* Большинство программ имеющие мощные функциональности, позволяющие им моделировать сложные модели являются дорогостоящими;
* Для построения сложных моделей необходимо использовать алгоритм оценки формы по движению;
* Для получения более точных результатов необходимо применять автоматические калибровки;

# **Глава 2. Проектирование разрабатываемой системы**

# **2.1. Обоснование выбора способа решения**

В соответствии с целями разработки системы, программа должна предоставлять пользователю необходимые результаты расчета, учитывая все ограничения, накладываемые на нее. На основе вышеперечисленных алгоритмов трехмерной реконструкции и способов обеспечении требуемого результата будут выбраны следующие алгоритмы:

Для решения задачи трехмерной реконструкции будет использован метод стереозрение. По сравнению с другими методами, метод стереозрение является менее затратным по времени и с точки зрения аппаратной части, а также данный метод позволят достичь требуемой точности.

Для решения задачи поиска сопряженных точек будет использоваться локальный метод поиска. Данный метод позволяет находить соответствие между двумя точками на двух изображениях за короткое время. А также если минимизировать участие сторонних предметов, влияющих на точность работы, данный метод может предоставить точные координаты расположения двух соответствующих точек.

Таким образом можно представить алгоритм решения в виде схемы как показано на рисунке 2.1.1.



Рисунок 2.1.1 – алгоритм формирования трехмерной модели

# **2.2. Конструкторский раздел**

Для реализации системы будет использована экспериментальная установка состоящей из камеры и наблюдаемого объекта. Схема установки продемонстрирована на рисунке 2.2.1. 1 – брусок высотой 10см и c основанием 4.5х4.5 см, 2 – камера SJ4000, 3 – персональный компьютер.



Рисунок 2.2.1 – принципиальная схема экспериментальной установки.

# **2.3. Метод трехмерной реконструкции**

Поскольку для данной установки используется всего одна камера, нахождение карты глубины будет производиться после получения двух изображений с видео потока со сдвигом камеры вбок на 1.5см.

В определенных ситуациях, для достижения лучшего результата может быть использован дополнительный источник освещения, направленный на сцену для выделения объекта от заднего фона.

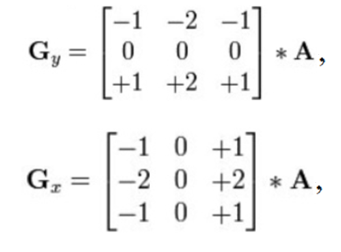
# **2.4. Поиск сопряженных точек**

Перед тем как произвести поиск сопряженных точек для получения карты диспаратности, применяем фильтрацию для каждого изображения, полученных с видеопотока. Таким образом, это позволяет ускорить время работы программы и уменьшить погрешности.

Данный фильтр основан на операторе Собеля. Оператор находит градиент интенсивности для каждого пикселя на изображении. Благодаря этому находится направление наибольшего увеличения интенсивности и величина ее изменения в этом направлении. Результат фильтра показывает, насколько резко или плавно меняется интенсивность в каждой точке. Иными словами, результатом оператора Собеля в точке, находящийся в области постоянной яркости, будет нулевой вектор, а в области различной яркости – вектор, направленный в сторону увеличения яркости.

Для реализации используется ядро 3x3, с которым производят свертку для каждого значения интенсивности в окрестности 3х3, чтобы вычислить приближенные значения производных по вертикали и по горизонтали. Пример работы фильтра Собеля приведены на рисунках 2.4.1 – 2.4.4.

Пусть Gx и Gy – это два изображения, где каждая точка содержит приближенные производные по x и по y, они рассчитываются по формуле 2.4.1.

 (2.4.1)

где А – исходное изображение.

После вычисления Gy и Gx они суммируются по формуле 2.4.2.

(2.4.2)

Где G – результат, который записывается в новое изображение с текущими координатами x, у.

Далее для поиска сопряженных точек будет использоваться локальный метод поиска. Использование счета соответствия в локальном методе позволяет описать схожесть двух точек и выявлять сопряженные точки. В случае, если пары пикселей похожи, значение будет малым, а для различных значение будет велико. При условии, что изображения ректифицированы и откалиброваны, поиск сопряженных точек ограничен одной строкой. Для нахождения соответствия сравнивается каждая точка левого изображения с каждой точкой соответствующей строки правого изображения. Поскольку на сцене присутствует всего 1 объект наблюдения, и с течением времени не меняется расстояние до камеры, для решения задачи достаточно получить карту диспаратности в области, где находится объект.



Рисунок 2.4.1 – полутоновое изображение.



Рисунок 2.4.2 – применение оператора Собеля к изображению.

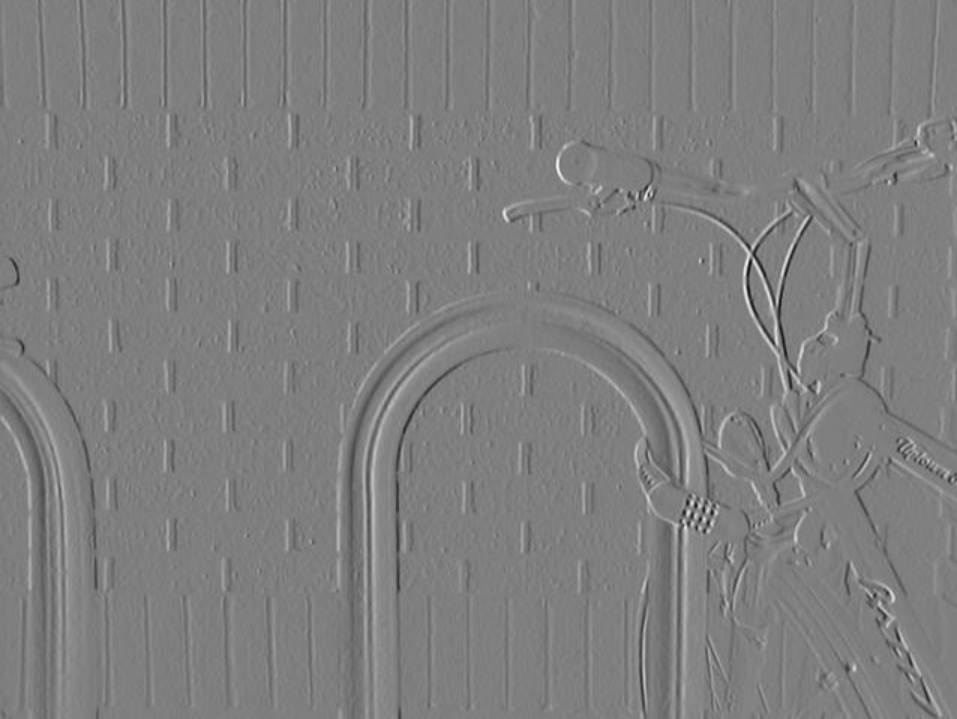


Рисунок 2.4.3 – Применение оператора Собеля к изображению по оси х.

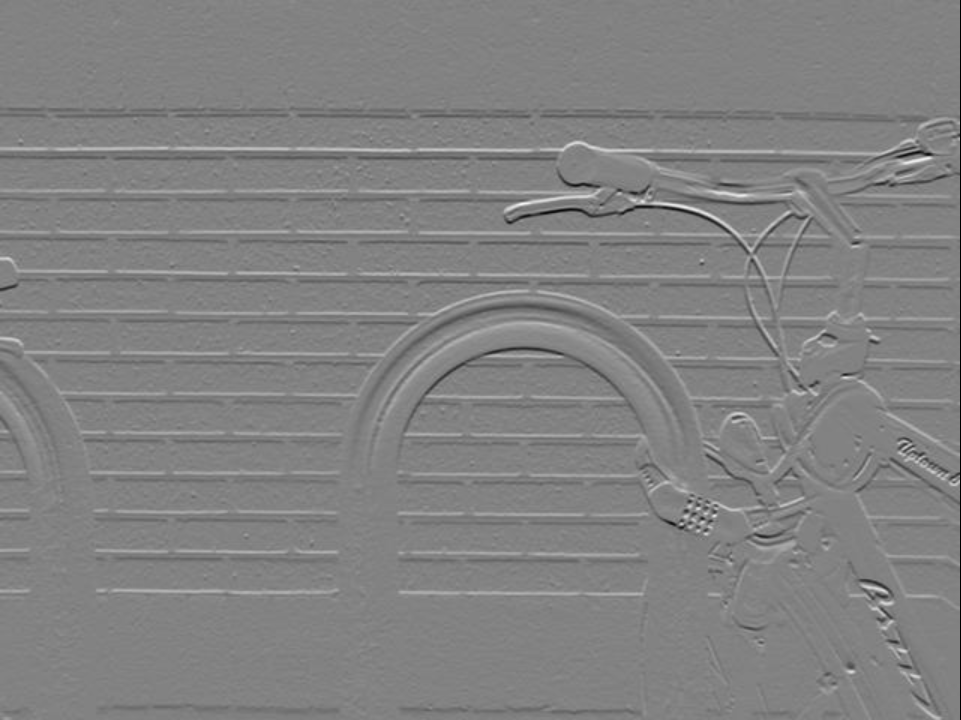


Рисунок 2.4.4 – применение оператора Собеля к изображению по оси у.

# **2.5. Формирование объемной модели**

Для формирования трехмерной модели создается виртуальная сцена, которая должна будет получить необходимые параметры о фигуре для отрисовки. Данные параметры находятся с помощью карты глубины.

В ходе создания экспериментальной установки, изначально камера находится слева от центра наблюдаемого объекта на 0.75см. После получения первого изображения, происходит сдвиг вправо на 1.5см, так, что камера находится правее от центра наблюдаемого объекта на 0.75см. Таким образом объект всегда находится примерно в центре изображения. Учитывая эти моменты, достаточно найти минимальное значение интенсивности на карте глубины внутри области наблюдаемого объекта и значения на границах, с помощью которых можно будет получить необходимые параметры фигуры для построения объемной модели.

# **Выводы**

В ходе исследования существующих алгоритмов для решения поставленной задачи, были выбраны алгоритмы стереозрение и локальный метод поиска сопряженных точек. Поскольку они позволяют достичь желаемого результата за оптимальные трудозатраты.

Программа должна предоставлять пользователю следующие возможности:

* получение видеопотока с камеры;
* указание позиции захвата изображения;
* отображение ключевых параметров фигуры с погрешностью;
* создание сцены для вывода трехмерной модели;

# **Глава 3. Разработка системы и тестирование модели**

# **3.1. Используемые средства разработки**

# **3.1.1. Язык программирования JavaScript (JS)**

JS – это мульти-парадигменный язык программирования, являющийся прототипно-ориентированным, функциональным и императивным. Особенностью данного языка является ориентированность на удобства использования. JS как самостоятельный язык является узкоспециализированным, однако с использованием специальной программной платформы Node.js, JS превращается в язык общего назначения.

JS в настоящее время используется для разработки приложений и веб-страниц, а также программировать микроконтроллеры. Благодаря Node.js появляется возможность подключать сторонние библиотеки, написанные на различных языках, а также взаимодействовать с устройствами ввода-вывода через интерфейс приложения (API) написанный на C++. Мощность языка характеризуется в огромном количестве пользовательских библиотек, направленных на решение различных задач в различных областях, в том числе и в компьютерное зрение.

JS программы в браузере подключаются напрямую к HTML и сразу запускаются после загрузки страницы. Помимо браузера программа может работать там, где присутствует интерпретатор, а процесс выполнения программы называется интепретацией.

JS как язык программирования изначально создавался для того, чтобы делать различные веб-сайты. Программы, написанные на языке JS, называется скриптами. Однако благодаря различным фреймворкам, например Electron, AppJS или NW.js появляется возможность разрабатывать нативные графические приложения для настольных операционных систем такие, как Linux, Windows, macOS с помощью веб-технологий.

Основные недостатки языка JS является:

* отсутствует стандартная библиотека. JS не имеет возможности самостоятельно работать с файлами, потоками ввода-вывода;
* сложность понимании кода, плохая читабельность;
* в JS используется другая концепция ООП, отличающее от других языков программирования, из-за чего возникают сложности в понимании при переходе с других языков на JS;

Однако помимо недостатков, JS имеет следующие преимущества:

* гибкость языка. Возможность использовать большое количество шаблонов программирования и решать разнообразные задачи;
* огромное количество готовых библиотек, которые позволяют упростить написание кода и избегать несовершенства синтаксиса;

# **3.1.2. Библиотека JQuery**

JQuery – это библиотека JS, которая содержит в себе готовые функции, позволяющие получить доступ к любому элементу DOM, обращаться к содержимому элементов и манипулировать ими. Хотя с помощью библиотеки можно решать большое количество различных задач, разработчики библиотеки не стремились совмещать в JQuery функции, которые бы подошли всюду, поскольку это привело бы к большому нагромождению, часть которого не была бы востребована.

В JQuery происходит разделение поведения от структуры HTML. Такое разделение также называется принципом ненавязчивого JavaScript. Принцип ненавязчивого JavaScript заключается в том, что, например, вместо прямого указания события нажатия кнопки, управление передается JQuery, которая в свою очередь идентифицирует кнопку и преобразует ее в обработчик события клика.

# **3.1.3 Сторонние пакеты**

Для установки сторонних пакетов используется специальный пакет менеджер – node package manager (npm). Данный менеджер используется для скачивания пакетов из облачного сервера, либо загрузки пакетов на эти сервера.

В качестве сторонних пакетов используются:

* пакет Sobel – позволяет производить фильтр Собеля к изображениям;
* пакет Three – позволяет создать сцену для трехмерного моделирования;

# **3.2. Алгоритм обработки данных**

**3.2.1. BMP формат**

В ходе получения изображений из видеопотока, для работы с ними необходимо привести их к единой прямоугольной системе. Для этого все изображения переводится в формат BMP.

BMP – аппаратно-независимое побитовое изображение. Структура файла используется для хранения растровых изображений. Формат BMP состоит из 3 основных блоков различного размера:

* заголовок файла, состоящий из BITMAPFILEHEADER и BITMAPINFO;
* таблица цветов;
* пиксельные данные;

В свою очередь блок BITMAPINFO состоит из:

* информационные поля;
* битовые маски;
* таблица цветов;

Схема формата файла приведена на рисунке 3.2.1.1.

BITMAPFILEHEADER – 14 байтная структура, которая располагается в начале файла. Показывает какой формат имеет данный файл, размер файла и смещение изображения от начала файла.

BITMAPINFO – идет сразу после BITMAPFILEHEADER. В зависимости от версии имеют различные размеры, чаще всего состоит из 40 байтной структуры. Хранит размер заголовка, ширину и высоту изображения, размер изображения и т.д.

Далее идет таблица цветов состоящая из 1024 байтов, 256 элементов по 4 байта. Каждое поле соответствует своему цвету в палитре, 3 байта являются компонентами RGB составляющих для этого цвета, а 4 байт для каждого поля является зарезервированным и равен 0.

После идет данные изображения со смещением в 1078 байтов. Изображение записано по строкам слева направо и снизу вверх.

**3.2.2. Работа с изображением**

Поскольку после получения изображения в формате BMP, необходимо получить пиксельные данные для дальнейшей работы. Учитывая размер заголовки файла и таблицу цветов, данные находятся со смещением в 1078 байтов. Поскольку изображение изначально является цветным, необходимо преобразовать его в черно-белое, иными словами вместо 4 цветовых каналов: красный, зеленый синий и альфа канал, будет использоваться 1 цветовой канал: значение трех компонентов равны между собой, а альфа канал остается без изменения. Для этого применяется фильтр Собеля, в каждой точке которого будет храниться интенсивность изменения цвета. Далее преобразовываем изображение размером 320 пикселей в высоту и 640 пикселей в ширину в матрицу такого же размера.

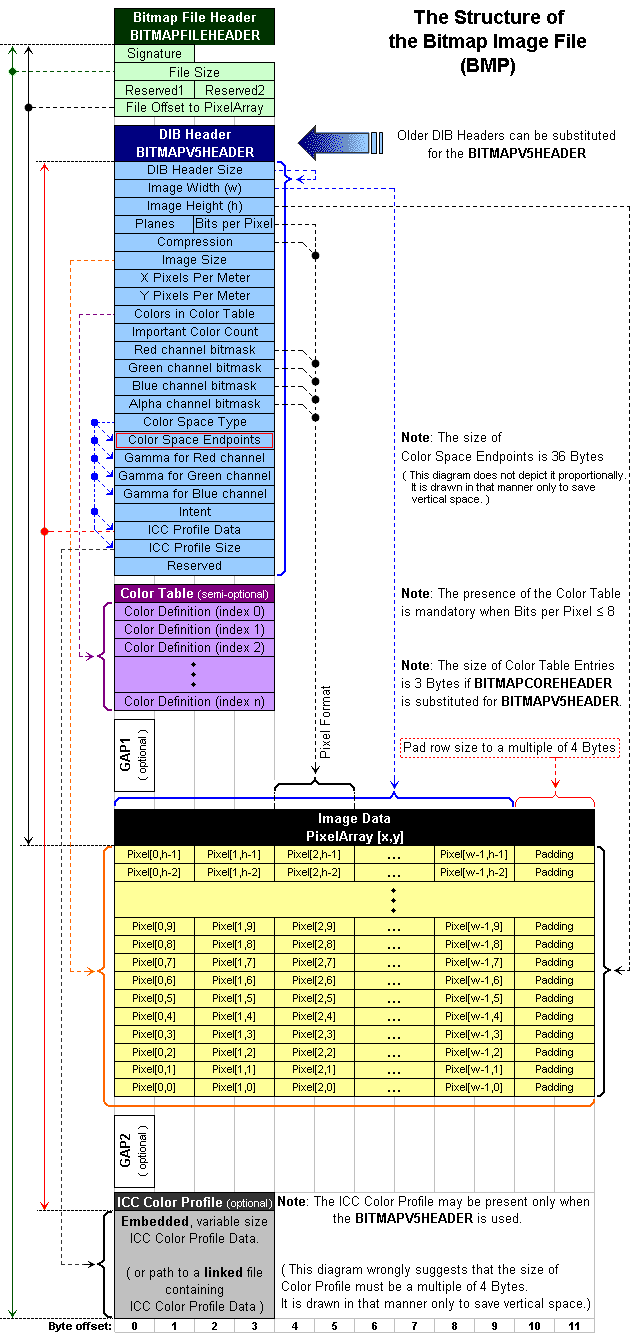


Рисунок 3.2.1.1 – структура формата BMP.

# **3.3.3. Дизайн пользовательского интерфейса**

Под пользовательским интерфейсом программы понимается канал взаимодействия между пользователем и программы. Для взаимодействия с программой используются специальные элементы управления, позволяющие управлять ее работой и получать необходимые результаты.

Пользовательский интерфейс программы включает:

* средства отображения информации;
* устройства ввода данных;
* двустороннее взаимодействие между пользователем и компьютером;
* порядок использования программы;

Совокупность всех элементов и компонентов программы позволяют оказывать влияние на взаимодействие пользователя при работе с программой. Однако помимо этих вещей, существуют еще и внешние факторы, которые также могут оказывать влияние.

К этим факторам относятся:

* набор задач, которые решает пользователь при помощи программы;
* визуальный дизайн программы;
* порядок использования программы;
* документация;

Интерфейс должен характеризоваться удобством в использовании, эффективностью, понятностью и понятием “дружественности”.

Принцип дружественного интерфейса:

* защита системы от непрофессиональных действий на ПК – предоставляет пользователю право на ошибку при работе со системой;
* существование системы “отката”, в случае если при выполнении программы, результаты не удовлетворили пользователя, выполнить программу еще раз;
* простой дизайн – облегчает процесс работы пользователя с программой;

По такому принципу, дружественный интерфейс должен предоставлять пользователю наиболее удобный способ работы с программой путем обеспечения простоты расположения элементов управления, безопасности и логичности.

Для создания хорошего дизайна требуется знания о множестве факторов, которые обеспечивают пользователю удобность работы.

Для придания дизайна используется CSS – формальный язык описания внешнего вида документа, написанного с использованием языка разметки.

Для добавления новых элементов используется HTML – стандартизированный язык разметки документов.

# **3.3.4. Процесс разработки интерфейса системы**

Внешний вид начального окна можно разделить на 7 разделов.

* первый раздел – отвечает за получение видеопотока в режиме real-time от видеокамеры. Рисунок 3.3.4.1;

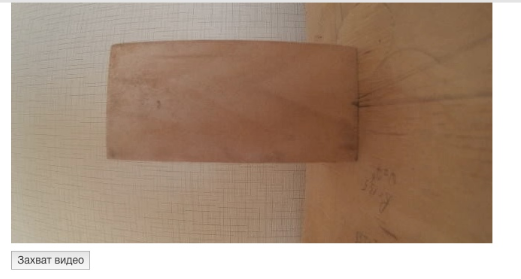


Рисунок 3.3.4.1 – захват видеопотока.

* второй раздел – отвечает за очередность получения изображения. Рисунок 3.3.4.2;

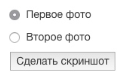


Рисунок 3.3.4.2 – выбор очередность получения изображения.

* третий раздел – отвечает за вид отображаемых изображений, в целях удобства пользователя. Рисунок 3.3.4.3;



Рисунок 3.3.4.3 – выбор способа отображения изображений.

* четвертый раздел - отвечает за отображение полученных изображений, в целях удобства пользователя. Рисунок 3.3.4.4 – 3.3.4.5;

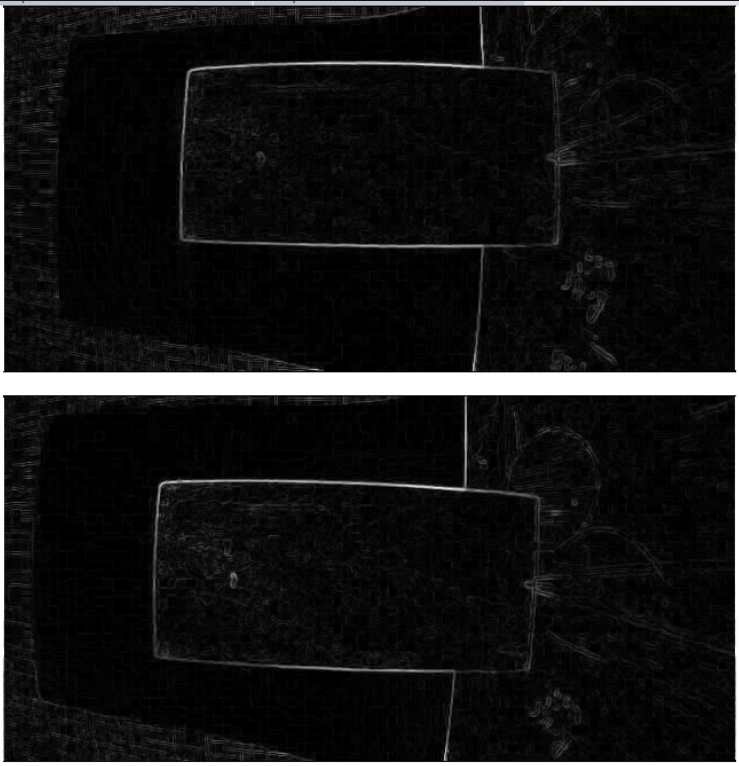


Рисунок 3.3.4.4 – изображения с фильтром.

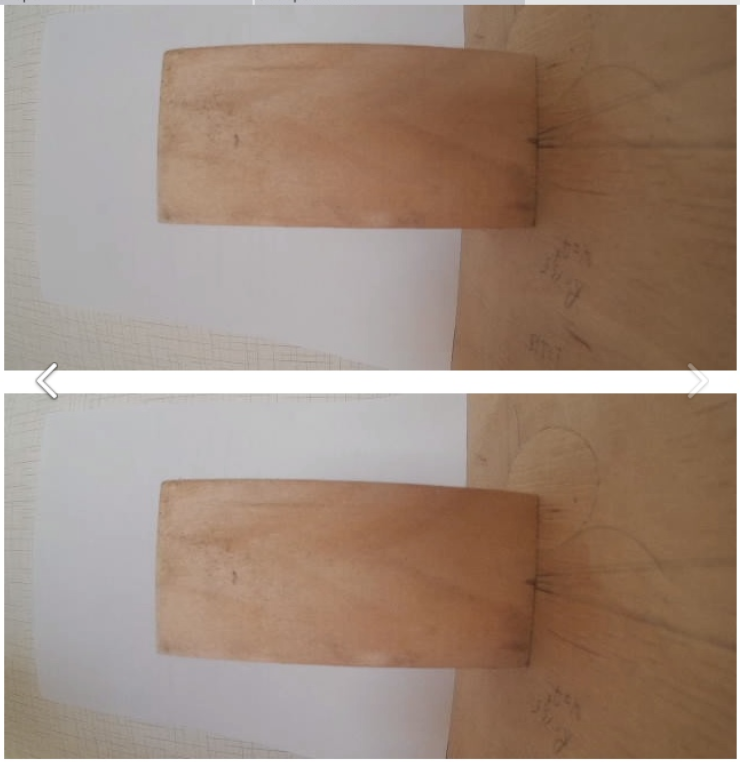


Рисунок 3.3.4.5 – изображения без фильтра.

* пятый раздел –показывает полученные результаты и погрешности. Рисунок 3.3.4.6;

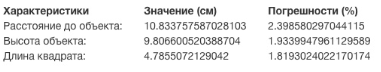


Рисунок 3.3.4.6 – получение параметров фигуры.

* шестой раздел – сохраняет полученные результаты для дальнейшего отображения трехмерной модели. Рисунок 3.3.4.7;



Рисунок 3.3.4.7 – сохранение результатов подсчета.

* седьмой раздел – меню для перехода. Рисунок 3.3.4.8;



Рисунок 3.3.4.8 – верхнее меню.

Общий внешний вид начального окна программы показан на рисунке 3.3.4.9.

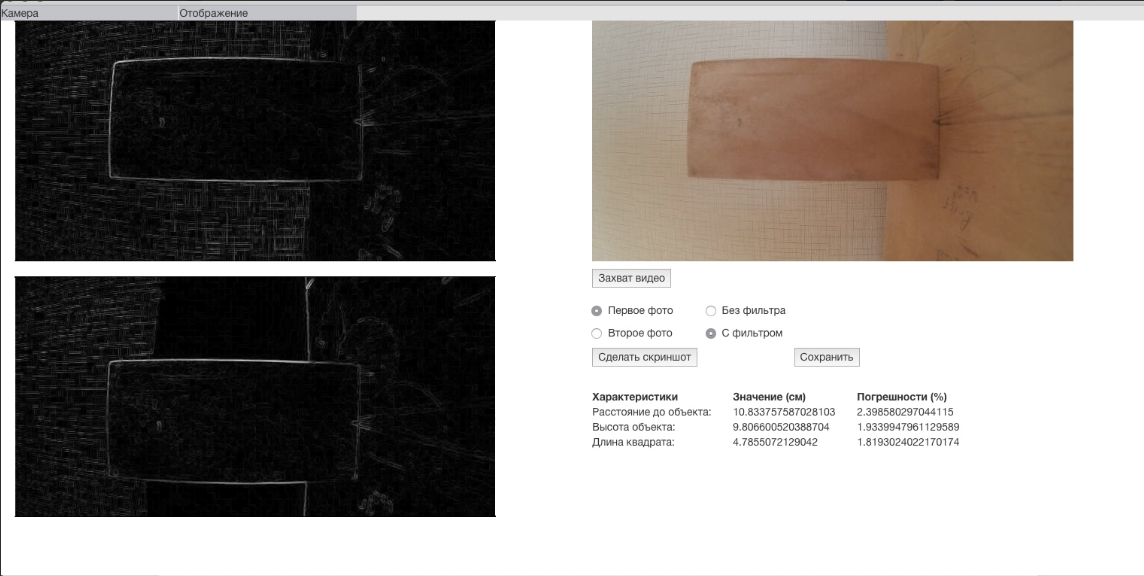


Рисунок 3.3.4.9 – внешний вид главного окна.

При нажатии на кнопку сохранить на рисунке 3.3.4.7, программа автоматические переключается на следующее окно для отображения результата. Рисунок 3.3.4.10.

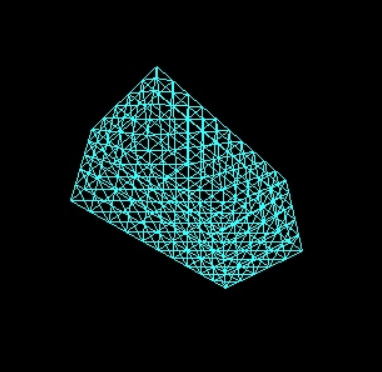


Рисунок 3.3.4.10 – результат работы программы.

Для начала работы необходимо подключить видеокамеру к персональному компьютеру. Далее для того, чтобы получить видеопоток необходимо нажать на кнопку “Захват видео”. При успешной работе событии нажатия кнопки, появляется окно для вывода видеопотока, как показано на рисунке 3.3.4.1. За счет данного окна и будет происходить захват изображения. Выбор в какую переменную сохранять изображение и в каком виде отображать на экране, происходит при нажатии на кнопки переключателя, рисунки 3.3.4.2 - 3.3.4.3.

После получения двух изображений, программа автоматические выводит результаты параметров фигуры. Чтобы получить трехмерную модель объекта, необходимо нажать на кнопку “сохранить”, рисунок 3.3.4.7, либо нажать на “Отображение” в разделе меню.

# **3.3.5. Тестирование системы**

В данном подразделе приводится результаты исследования, цель которого – оценить точность работы программы.

Для данной экспериментальной установки известны следующие данные: высотка бруска составляет 10см, размер основания бруска 4.5х4.5см, расстояние от камеры до бруска 11.1см, фокусное расстояние камеры 2.8 мм с углом обзора 170°, значение базы 1.5см.

О работоспособности системы можно судить на основании сравнения реальных и полученной программно параметры объекта. Для удобства сравнения дополнительно указывается погрешность.

Результаты тестирования для нескольких опытов:

* первый опыт, рисунок 3.3.5.1;



Рисунок 3.3.5.1 – результаты первого опыта.

* второй опыт, рисунок 3.3.5.2;



Рисунок 3.3.5.2 – результаты второго опыта.

* третий опыт, рисунок 3.3.5.3;



Рисунок 3.3.5.3 – результаты третьего опыта.

Погрешность рассчитывается по формуле 3.3.5.1;

(3.3.5.1)

где Vреал – реальные значения,

Vпрогр – значение, полученное программно.

# **3.3.6. Функциональные возможности и ограничения**

Цель данной работы является создание программы, которая должна получить необходимые ей данные, найти и вывести параметры наблюдаемого объекта, а также создать трехмерную модель объекта.

Основными преимуществами использования такой программы являются:

* дешевизна аппаратного решения – используется всего одна обычная камера вместо двух;
* достижение хорошей точности;
* простота установки;

Однако у данной программы присутствуются следующие ограничения:

* невозможность проведения автоматической калибровки. Поскольку калибровка — это задача получения внешних и внутренних параметров стереокамеры по имеющим фотографиям. Данный этап опускается, по причине того, что для данной экспериментальной установки используется всего одна обычная камера. Поэтому эти значения подбираются вручную;
* данная программа пока умеет работать только с простыми объектами, например с брусоком. Использование метода оценка формы по движению также не представляется возможно из-за использования одной камеры;
* данная программа работает только с двумя изображениями, поэтому если, например, в качестве наблюдаемого объекта будет участвовать брусок, имеющий различные вырезы по всему периметру. Программа будет рассматривать только переднюю сторону объекта;
* при плохом освещении, брусок может сливаться с задним фоном, из-за чего результаты могут быть некорректными;

# **Выводы**

Разработана программа, позволяющая создавать 3D модель объекта по набору двумерных изображений. Для этого был изучен метод преобразования изображении из одного формата в другой. Разработан пользовательский интерфейс, позволяющий получать видеопоток с видеокамеры в режим real-time и отображать результаты работы программы.

По итогам тестирования можно сделать вывод, что значения, полученные прогроммно близки к реальным значениям.

# **Заключение**

В ходе выполнения задания на выпускную квалификационную работу, была реализована программа, позволяющая создавать трехмерную модель по двум изображениям. В процессе выполнения были получены следующие результаты:

* проведен обзор литературы;
* рассмотрены существующие методы для создания трехмерной модели;
* изучены ограничения, накладываемые на систему;
* реализована программа, позволяющая создавать трехмерную модель.

# **Литература**

1. Научно-образовательный курс «Задача сопоставления локальных точек в 3D реконструкции». [Электронный ресурс]. URL: http://mmdsp.com/files/задача совмещения сопоставления локальных точек в 3D реконструкции.pdf (дата обращения: 12.04.2017).
2. Schlesinger M.I., Flach B. Some solvable subclass of structural recognition problems. CzechPatternRecognitionWorkshop 2000, pages 55-62.

<http://imaging.cs.msu.ru/pub/2009.Izevsk.yurin_volegov.3D.ru.pdf>

<http://ict.informika.ru/ft/002408/num2recon.pdf>

# **Приложение**