

Программно-аппаратный комплекс получения фотоизображений на основе технологии ИОТ и анализ точности различных алгоритмов цифровой генерации 3D моделей на основе принципа фотограмметрии

В.В. Коньков, А.Б. Замчалов, М.Г. Жабицкий

Аннотация. Авторами рассмотрены способы получения, передачи и накопления фотоснимков на основе технологии интернета вещей и проведен анализ точности различных алгоритмов цифровой генерации 3D моделей на основе принципа фотограмметрии. Данные методы были применены для разработки программно-аппаратного комплекса, позволяющего производить съемку объектов с разных сторон для последующего преобразования полученных фотоснимков в трехмерные объекты для дальнейшего сравнительного анализа различных алгоритмов цифровой генерации 3D изображений. Были изучены особенности использования различных методов позиционирования шаговых двигателей для перемещения фотокамеры вокруг объекта съемки. В ходе работы разработан программно-аппаратный комплекс для автоматизированных обмеров относительно малых объектов снаружи и помещения изнутри с применением как фотофиксации, так и лидарного сканирования.

Полученные фотоснимки геометрических фигур различной сложности передавались на удаленный компьютер и обрабатывались различными алгоритмами.

Целью исследования являлась автоматизация сквозного процесса от получения серий фотоснимков объектов с различных ракурсов и последующей генерации 3D моделей нужного качества за счет комбинирования разных методов и алгоритмов обработки серий фотонизображений.

Ключевые слова: интернет вещей, шаговые двигатели, фотограмметрия

I. ВВЕДЕНИЕ

Фотограмметрия — это метод исследования объектов и среды путем получения и обработки фотографических изображений. Она использует основные принципы геометрии, оптики и фотографии для определения черт и строения объектов. С помощью фотограмметрии можно измерить расстояния, высоту, ширину, площадь, объем объектов и другие параметры [1].

Она используется для создания 3D-моделей, карт высот, карт глубин и других типов карт, а также для измерения различных свойств объектов, таких как размеры, формы, деформации и т.д. Для этого

используются различные процессы и алгоритмы, такие как стереозрение, восстановление облака точек, коррекция смещения камеры и другие.

Объектом исследования является фотограмметрия как способ создания 3D моделей.

Предметом исследования является процесс определения формы, размеров, положения и иных характеристик объектов по их фотоизображениям с последующим преобразованием 3D модель.

Новизна работы заключается в комплексном анализе текущих алгоритмов с проверкой точности генерации различных 3D объектов различными метриками.

На текущий момент существует большое количество программного обеспечения для обработки разных типов поверхности. Данное исследование позволит выявить наиболее оптимальные алгоритмы для различных геометрических объектов и сократить пользователям время на поиск оптимальных методов преобразования объектов в 3D модели.

В качестве прогнозируемой востребованности результатов работы можно указать то, что данное решение позволит сократить время на поиск разностороннего программного обеспечения для преобразования реальных объектов в 3D-модель.

В дальнейшем, результаты данного исследования предлагается использовать при создании комплексной информационной системы автоматизированного формирования и накопления данных для анализа эффективности алгоритмов и программных средств фотограмметрии для различных внешних условий и методов получения исходных данных.

В исследовании рассматривается фотограмметрия как наука, позволяющая создавать 3D объекты из различных снимков соответствующих объектов. Изучаются области, в которых применяется фотограмметрическая обработка, выделяются и рассматриваются задачи, которые решаются в процессе обработки, анализируются алгоритмы, изучаются аналогичные решения анализа точности, а также приводятся существующие исследования на эту тему. Также выполняется сбор данных, полученных в результате работы программно-аппаратного комплекса получения, передачи и накопления серий фотоизображений для генерации 3D моделей для систем автоматизированного проектирования на

основе принципа фотограмметрии, происходит процесс обработки соответствующих данных в заранее определенном в рамках первого раздела программном обеспечении

В заключении выполняется сравнительный анализ точности каждого программного обеспечения для различных наборов данных, полученных в различных внешних условиях.

II. ОПИСАНИЕ ТЕХНОЛОГИИ И ПРИНЦИПА РАБОТЫ ФОТОГРАММЕТРИИ

Процесс фотограмметрии включает в себя несколько этапов, каждый из которых имеет свои задачи и подзадачи.

1. Съемка фотографий

Первый этап процесса фотограмметрии — это съемка фотографий объекта. Задача этого этапа состоит в создании геометрически точных изображений, с учетом особенностей объекта и съемочной техники.

Подзадачи этапа съемки фотографий:

- выбор съемочного оборудования с нужными параметрами (фокусное расстояние, угол обзора, разрешение и др.)
- определение точек установки камеры для правильной ректификации изображений;
- проведение съемки в нужных ракурсах для создания трехмерной модели.

Этот этап включает в себя следующие шаги:

- Выбор съемочной техники: на выбор съемочной техники влияют такие факторы, как размер объекта, расстояние до объекта, освещение, скорость съемки и стабильность камеры.
- Создание съемочного плана: съемочный план определяет ракурсы и углы, с которых будут сняты фотографии. План может быть создан вручную или с помощью специального программного обеспечения.
- Подготовка объекта: перед съемкой объект должен быть подготовлен. Например, внешние поверхности должны быть чистыми и отсутствовать элементы, которые могут повлиять на точность получаемых изображений.
- Фиксация камеры: камера должна быть зафиксирована на штативе или другом оборудовании для предотвращения движения во время съемки.
- Съемка изображений: фотографии должны быть сняты с разных ракурсов и углов, с ориентацией на съемочный план.

Требования к этому этапу включают в себя:

- Использование высококачественной съемочной техники для получения изображений
- Точное соответствие плана съемки и фактически снятых изображений
- Убеждение, что изображения высокого качества, чтобы была возможность использовать их в дальнейшем.

Ограничения по съемке могут включать в себя:

- Отсутствие доступа к некоторым ракурсам объекта
- Ограничения на использование летательных аппаратов (например, в аэропортах или на некоторых территориях заповедников).

Рекомендации по съемке фотографий объекта могут включать в себя:

- Использование фотоаппарата с широким углом обзора для получения наибольшего количества изображений
- Использование специального программного обеспечения для создания съемочного плана
- Оптимальное освещение объекта
- Использование устойчивого оборудования для фиксации камеры на месте без движения.

Пример: если требуется создать точную модель здания, фотографии должны быть сделаны в таком ракурсе, чтобы были учтены особенности потолочной структуры, пола, стен, а также оконных и дверных проемов.

2. Ректификация изображений

Второй этап фотограмметрии - ректификация изображений.

Задача этапа - преобразование фотографий в графические изображения с геометрической точностью.

Подзадачи этапа ректификации изображений:

- компенсация искажений, вызванных неоптимальной установкой камеры;
 - корректировка изображений, снятых при различном освещении, чтобы их контрастность была одинаковой;
 - уточнение геометрических параметров, которые включают интерпретацию областей изображения в пространственных координатах.
- Этот этап включает в себя следующие шаги:
- Определение ориентации камеры: определение ориентации камеры — это процесс, в котором измеряются углы наклона и поворота камеры, а также его положение и высота.
 - Идентификация пикселей: каждый пиксель на фотографии должен быть идентифицирован с соответствующим 3D-пикселем на объекте.
 - Коррекция дисторсии изображения: при каждой фотосъемке происходит искажение изображения, вызванное линзой камеры, которое требует коррекции.

Компенсация перекрытия изображений: поскольку изображения сняты с разных углов и ракурсов, часть объекта может быть перекрыта на одном изображении и открыта на другом. Эту компенсацию можно провести, чтобы на всех снимках объект был полностью виден.

Ректификация: после обработки каждого изображения, все изображения координатно связанные в единой системе координат, что позволяет видеть местоположение объекта точно.

Требования к этому этапу включают:

- Математическая точность и корректность расчетов
- Приведение изображений к единой системе координат
- Наличие точной камеры для съемки и определения ориентации

Ограничения по ректификации могут включать:

- Ограничения на доступ к изображению объекта
- Нехватка времени на обработку изображений

Рекомендации по ректификации могут включать в себя:

- Использование высококачественного оборудования для обработки изображений
- Проведение проверки результатов для установки корректности и точности делаемых расчетов
- Следование правилам и государственным стандартам в области фотограмметрии.

Пример: В процессе ректификации изображений линии на фотографии должны быть преобразованы в прямые линии. Если на фотографии здания линии крыши кажутся кривыми, то в процессе ректификации они должны быть преобразованы в прямые линии.

3. Триангуляция

Третий этап фотограмметрии — это триангуляция. Задача этого этапа - создание точной трехмерной модели объекта на основе двухмерных фотографий.

Подзадачи этапа триангуляции:

- установление соответствия между пикселями на нескольких фотографиях, которые касаются одной и той же точки;

- расчет координат точек в пространстве;
- создание 3D-модели объекта.

Этот этап включает в себя следующие шаги:

- Определение координат точек на изображении: определение точек на изображении объекта, которые будут использоваться для создания трехмерной модели. Это может быть выполнено при помощи получения координат точек на изображении и соответствующих им трехмерных координат.
- Триангуляция и создание трехмерной модели: это единственный этап в фотограмметрии, где к локализованному изображению применяется механизм триангуляции, которая использует как механические, так и оптические принципы [2].

- Обработка и визуализация: после триангуляции трехмерная модель обрабатывается и визуализируется в соответствии с требованиями пользователя.

Требования к этому этапу включают:

- Определение точек на изображении с высокой точностью;
- Использование высококачественного оборудования для обработки изображений;
- Наличие точных камер для съемки.

Ограничения по триангуляции могут включать в себя:

- Ограничения на доступ к изображению объекта;
- Ограничение на качество изображений объектов.

Рекомендации по триангуляции могут включать в себя:

- Использование большого количества изображений объекта, для повышения точности и детализации модели;
- Эффективное использование алгоритмов триангуляции как механических так и оптических принципов, для минимизации ошибок;
- Следование правилам и государственным стандартам в области фотограмметрии.

Пример: Фотографии здания могут быть использованы для создания трехмерной модели в соответствии со всеми объемными элементами, используемыми в его конструкции, такими как крыша, стены, окна и двери.

III. ПРИМЕРЫ И СРАВНЕНИЕ АНАЛОГИЧНЫХ ПРОГРАММНО-АППАРАТНЫХ КОМПЛЕКСОВ В МИРЕ И В РОССИИ

В рассматриваемой предметной области были изучены существующие программные решения, похожие на разрабатываемый программно-аппаратный комплекс.

В статье “Building a Photogrammetry Library” [3] говорится о разработанном устройстве съемки для получения 3D модели с помощью принципов фотограмметрии. Сделанная установка, представленная на рис. 1 под литерой А, представляет закрытый от дополнительного света куб, в котором помещено два дополнительных светодиода мощностью 50 Вт (рис. 1, Б).



Рис. 1 – А. Вид установки снаружи, Б. Вид установки внутри

Для съемки автор статьи использует зеркальную фотокамеру Sony, а в качестве вращения фотографируемого объекта используется поворотный стол (то есть объект вращается со столом, а камера остается на месте).

В качестве результатов, автор предоставил скриншоты обработанных после съемки объектов, один из них представлен на рис. 2.



Рис. 2 – Обработанная 3d модель

Таким образом, автор статьи разработал устройство съемки для получения 3d моделей с помощью принципов фотограмметрии. Из достоинств данного решения можно выделить использование хорошей фототехники, отличную освещенность, отсутствие лишних источников света.

Также в качестве аналогичного устройства можно выделить сканер Artec Ray. Artec Ray [4] (Рис. 3) — это профессиональный лазерный сканер, предназначенный для съемки объектов больших размеров на расстояниях до 110 метров. Он позволяет создавать точные 3D модели объектов, включая

промышленные установки, здания, мосты и другие объекты.



Рис. 3 – Сканер Artec Ray

Сканер Artec Ray обладает высокой скоростью съемки, надежностью и точностью до 0,7 мм на 10 метров. Он подходит для работы как внутри помещений, так и на открытых пространствах, благодаря своей высокой чувствительности к свету и возможности работы при различных уровнях освещенности.

С помощью Skylight, программного обеспечения от Artec, которое поставляется вместе со сканером, данные могут быть обработаны и использованы для создания точных 3D моделей объектов. Программа позволяет объединять различные наборы данных в единые проекты, визуализировать данные в различных форматах, а также создавать готовые файлы для коммерческих средств, таких как AutoCAD и SolidWorks. Результат отсканированного объекта представлен на рис. 4.

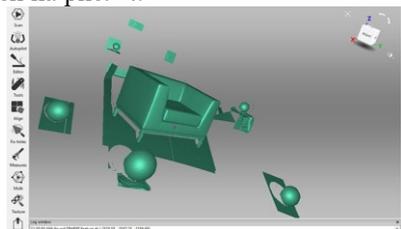


Рис. 4 – Обработанная 3д модель

Таким образом, рассмотрев данные аналогичные разработки, можно рассмотреть или позаимствовать их методы работы для дальнейшей разработки нашего разрабатываемого программно-аппаратного комплекса.

IV. ПРИМЕРЫ И СРАВНЕНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ 3D ОБРАБОТКИ В РОССИИ И МИРЕ

В качестве примера использования фотограмметрии можно привести работу ОЦЕНКА ТОЧНОСТИ 3D-МОДЕЛЕЙ, ПОСТРОЕННЫХ С ИСПОЛЬЗОВАНИЕМ БЕСПИЛОТНЫХ АВИАЦИОННЫХ СИСТЕМ [5], где авторы проводят создание трехмерной модели эталонного полигона в Agisoft PhotoScan в 4 этапа:

1. Выравнивание снимков и построение разреженного облака точек (по терминологии документации программы).
2. Построение плотного облака точек.
3. Создание полигональной модели объекта.
4. Построение текстуры модели и ортофотоплана.

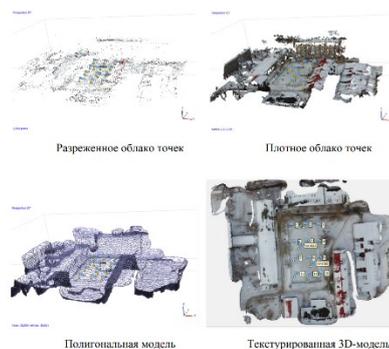


Рис. 5 – Моделирование полигона

В качестве другого примера можно привести исследование «СОЗДАНИЕ ОРТОФОТОПЛАНА В ПРОГРАММЕ AGISOFT PHOTOSCAN», [6] в которой авторы в лице Михеева А.А., Ялтыхов В.В., Парадня П.Ф. строят полигональную модель здания в программе Agisoft PhotoScan и проходят через следующие этапы:

1. Построение плотного облака точек.
2. Формирование полигональной модели местности.
3. Проецирование отдельных полигонов на заданную поверхность.
4. Создание матрицы высот.
5. Создание ортофотоплана.

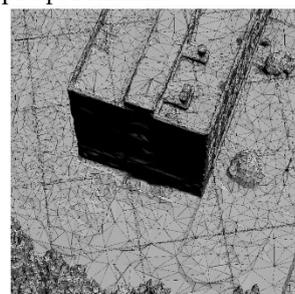


Рис. 6 – Моделирование ортофотоплана

В качестве еще одного примера можно привести статью «Исследование точности фотограмметрии как метода определения объема объекта», где автор анализирует простейшие 3D объекты в Agisoft Metashape.



Рис. 7 - Подопытные объекты, располагающиеся на вспомогательном паттерне «шахматная доска»

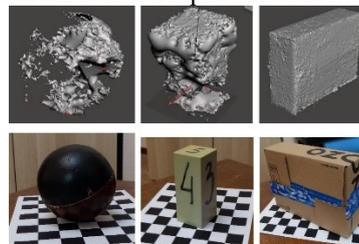


Рис. 8 - Результаты экспериментов. Снизу реальные объекты, сверху их 3D модели (справа налево — шар, параллелепипед №1, №2)

Существуют различные программные продукты для фотограмметрии, которые включают в себя как коммерческие, так и бесплатные решения (таб. 1).

Таблица 1 – Сравнение ПО обработки изображений

№	Программное обеспечение	Лицензия	Поддерживаемые форматы	Основные возможности
1	Autodesk ReCap	Платная	OBJ, RCM, RCS, RCP	Создание точных моделей, поддержка облаков точек
2	RealityCapture	Платная	OBJ, PLY, XYZ	Быстрая обработка, высокая точность, поддержка GPU
3	Agisoft Metashape	Платная	OBJ, PLY, STL, VRML, FBX	Обработка больших объемов фотографий, возможность создания топологических карт
4	AliceVision Meshroom	Бесплатная	OBJ, PLY, MVSM, MVS, SFM	Автоматическая обработка, поддержка больших объемов фотографий
5	3DF Zephyr	Платная и бесплатная	OBJ, PLY, STL, VRML, FBX	Широкие возможности ретопологии, обработка больших объемов фотографий
6	Colmap	Бесплатная	PLY, OBJ, VRML	Быстрая обработка, большое количество настроек
7	Regard3D	Бесплатная	PLY, OBJ, STL	Автоматическая обработка, поддержка больших объемов фотографий
8	VisualSFM	Бесплатная	OBJ, PLY	Быстрая обработка, поддержка больших объемов фотографий
9	iWitness	Платная	DXF, LAS, PLY, RCP, STL, VRML, ZFS	Поддержка большого количества форматов, высокая точность моделей
10	Photomodeler	Платная	FBX, SKP, STL, DWG, DXF, OBJ, PLY, WRL, VRML	Широкие возможности ретопологии, автоматическая обработка
11	Qlone	Бесплатная	OBJ, STL	Простота в использовании, быстрая обработка, настройка освещения

Например, в статье «Photogrammetry: Step-by-Step Guide and Software Comparison» [7] авторами подробно рассматривается весь процесс, используемый в фотограмметрии, в котором авторы выделяют следующие стадии

- о Создание снимков объектов;
- о Загрузка полученных данных в каждую из программ с различными возможностями конфигурации параметров;

- о Создание 3D моделей, который включает в себя:

- о Извлечение признаков, где авторы указывают, что программа ищет особенности, которые могут быть однозначно определены на нескольких изображениях. Профессиональные инструменты для этого используют кодированные маркеры, что является точным методом, работающим на различных поверхностях. Однако, большинство программ применяет более обобщенный метод, SfM, основанный на поиске плотных текстур на объектах, таких как тексты, деревья и черты лица.

- о Триангуляция, где авторы рассказывают, что еще в 1480 году Леонардо да Винчи разработал метод определения местоположения художника по картине, что легло в основу метода Structure From Motion

(SfM), где 3D Координаты точек поверхности оцениваются на основе выходных данных предыдущей фазы, графа сцены. Линии обзора от камеры до объекта реконструируются, в результате чего образуется так называемое лучевое облако. Пересечение многочисленных лучей определяет конечные 3D-координаты объекта. После определения глобальной геометрии с помощью разреженного облака точек программное обеспечение для фотограмметрии анализирует освещение и текстуру сцены для создания карты глубины. Затем как так называемая плотная реконструкция, включающая карту глубины, так и разреженная реконструкция, отображающая все визуальные особенности, обнаруженные на более ранних этапах, объединяются в формат 3D-сетки, такой как FBX, OBJ, PLY или STL.

- о Постобработка полученных 3D моделей с последующей корректировкой параметров качества созданной модели.

В заключении авторы приводят сравнительную таблицу, где на основе качества, скорости, дополнительных функций, удобного интерфейса и цены производится выставление оценок программам Autodesk ReCap, RealityCapture, Agisoft Metashape, AliceVision Meshroom, 3DF Zephyr, Colmap, Regard3D, VisualSFM, iWitness, Photomodeler, Qlone.

Авторы в статье приводят обзор программных средств, используемых в фотограмметрии, и делают следующие выводы:

- о Autodesk ReCap: Удобный интерфейс, поддержка облаков точек и множественных камер. Стоимость высокая, процессинг может быть медленным [8].

- о RealityCapture: Одна из самых точных программ, имеющих на рынке. Она может обрабатывать миллионы точек и предоставлять высококачественные 3D-модели даже при работе с фото высокого разрешения. Но стоимость ее очень высока [9].

- о Agisoft Metashape: Доступная цена, удобный интерфейс, быстрый и точный процессинг. Но может быть сложно работать с многокамерными настройками [10].

- о AliceVision Meshroom: Бесплатная программа с открытым исходным кодом. Она быстро обрабатывает фото и создает точные 3D-модели, но может иметь проблемы с работой на слабых компьютерах [11].

- о 3DF Zephyr: Удобный интерфейс, точный и быстрый процессинг, поддержка множественных камер. Но стоимость программы может быть высокой для некоторых пользователей [12].

- о Colmap: Бесплатная программа с открытым исходным кодом. У нее простой интерфейс, хорошая точность, и она хорошо работает с многокамерными системами, но ее процессинг временами может быть медленным [13].

- о Regard3D: Бесплатная программа с открытым исходным кодом. Она имеет простой интерфейс и может работать со множеством камер, но ее процессинг может быть медленным [14].

о VisualSfM: Бесплатная программа с открытым исходным кодом. Она хорошо работает со многокамерными системами и обрабатывает большое количество фотографий, но ее точность часто оставляет желать лучшего [15].

о iWitness: Программа, специально разработанная для правоохранительных органов. Она имеет хорошую точность и может работать со многими камерами, но ее доступность может быть ограничена только для определенных организаций [16].

о Photomodeler: Удобный интерфейс и точный процессинг, но стоимость программы может быть высокой для некоторых пользователей [17].

о Qlone: Бесплатная мобильное приложение, которое может работать с фото на смартфоне пользователя. Оно хорошо работает с простыми моделями, но может иметь проблемы с точностью и сложными объектами [18].

Также авторами приводится иллюстрация результата построения облака точек, полученного в ColMap и MeshRoom на примере объекта «Компьютерная мышь»

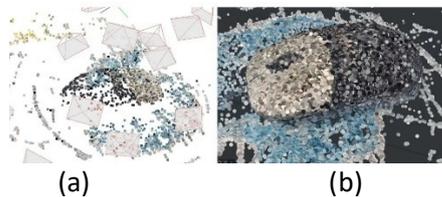


Рис. 9 - иллюстрация результата, полученного в ColMap и MeshRoom на примере объекта «Компьютерная мышь»

Далее авторы приводят иллюстрацию результата построения плотного облака точек – Mesh для программ Colmap, Meshroom, Qlone, Regard3D, 3DF Zephyr, and RealityCapture

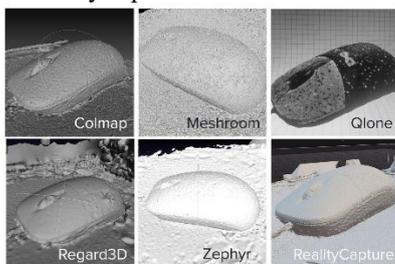


Рис. 10 - иллюстрация результата получения Mesh на примере объекта «Компьютерная мышь»
В таблице 2 представлен сравнительный анализ методов.

Таблица 2 – Сравнительный анализ методов

Программное обеспечение	Методы / Алгоритмы	Описание
Agisoft Metashape	Structure from Motion, Multi View Stereo, Depth Maps, Densely Packed Point Clouds	Agisoft Metashape использует популярные методы фотограмметрии, включая Structure from Motion и Multi View Stereo, для создания точных 3D-моделей. Это программное обеспечение также использует Depth Maps для создания точных моделей и Densely Packed Point Clouds для повышения качества реконструкции.
RealityCapture	Multi View Stereo, Depth Maps, Densely Packed Point Clouds, Mesh	RealityCapture использует Multi View Stereo и Depth Maps для создания точных 3D-моделей, а также использует Densely Packed Point Clouds для улучшения

	Reconstruction, Texture Mapping	реконструкции. Дополнительные методы включают построение сетки и создание текстур.
Pix4D	Structure from Motion, Multi View Stereo, Depth Maps, Densely Packed Point Clouds	Pix4D использует Structure from Motion и Multi View Stereo для создания точных 3D-моделей, а также использует Depth Maps и Densely Packed Point Clouds для повышения качества реконструкции. Оно также предоставляет полный набор средств для работы с аэрофотоснимками и веб-сервисом.
OpenDroneMap	Structure from Motion, Multi View Stereo, PCL	OpenDroneMap использует Structure from Motion и Multi View Stereo для создания точных 3D-моделей, и включает Point Cloud Library (PCL) для обработки облака точек. Это бесплатное программное обеспечение и имеет открытый исходный код.
COLMAP	Structure from Motion, Multi View Stereo, Depth Maps	COLMAP использует Structure from Motion и Multi View Stereo для создания точных 3D-моделей, а также использует Depth Maps для повышения качества реконструкции. COLMAP бесплатен и имеет открытый исходный код.

Structure from Motion (SfM) — это метод создания 3D-моделей объектов на основе набора 2D-изображений. Он использует различные математические алгоритмы для определения геометрических параметров объекта и реконструкции его 3D-формы [19]. Алгоритм работы метода можно описать следующим образом:

- В начале система получает набор из двух или более 2D-изображений объекта с разных ракурсов.
- Далее, система определяет ключевые точки на каждом изображении — это может быть ребро, угол или другой уникальный объект на изображении, который помогает системе устанавливать соответствия между разными изображениями.
- Затем система находит соответствующие ключевые точки на каждом изображении и определяет камерную матрицу для каждого изображения. Камерная матрица позволяет системе знать положение и ориентацию камеры в момент съемки.
- После этого система использует эти камерные матрицы и ключевые точки для выполнения реконструкции структуры и движения объекта. Алгоритм SfM позволяет найти трехмерное расположение ключевых точек объекта, используя геометрическую информацию из камерных матриц каждого изображения.
- Наконец, система создает 3D-модель объекта на основе информации о точках объекта и их трехмерном расположении.

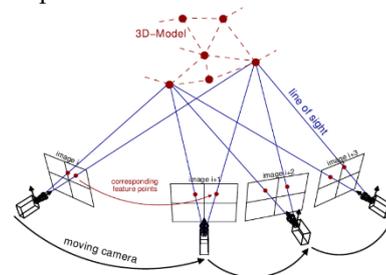


Рис. 11 - Structure from Motion (SfM)

OpenMVS — это программное обеспечение, которое используется для построения плотной 3D-модели на основе набора изображений.

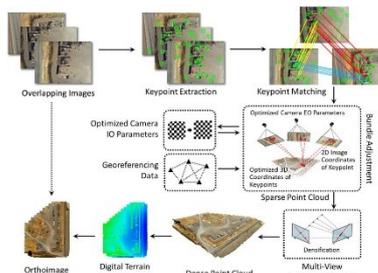


Рис. 12 – Шаги от SfM до MVS

Рассмотрим процесс работы OpenMVS на шагах:

1. Сегментация изображений

Первый шаг заключается в разбиении каждого изображения на сегменты, которые представляют одноцветные области. Для этого используются методы компьютерного зрения, такие как алгоритмы сегментации на основе цвета.

2. Создание облака точек

Далее выбираются точки интереса из каждого сегмента изображения, после чего вычисляются их дескрипторы. На основе этих дескрипторов строится облако точек для каждого изображения.

3. Оценка внутренних и внешних параметров камер

На этом этапе вычисляются внутренние и внешние параметры камер для каждого изображения. Начальные приближения вычисляются при помощи алгоритма реконструкции, такого как преобразование Хафа.

4. Построение плотной 3D-модели

Используя внутренние и внешние параметры камер и матрицу корреспондент, OpenMVS выполняет MVS-реконструкцию, которая позволяет построить плотную 3D-модель. В этом процессе для каждой точки объекта на поверхности вычисляется ее глубина по всем изображениям с помощью корреляции пикселей и других методов. Далее, на основе полученных глубинных значений строится плотную 3D-модель, которая описывает форму объекта.

Для построения плотной 3D-модели OpenMVS использует метод множественного обхода (multi-view stereo, MVS), который позволяет оценить глубины поверхности объекта, используя множество изображений с разных ракурсов. Этот метод основывается на поиске соответствий между одинаковыми точками на разных изображениях. Найденные соответствия служат для построения структурного скелета, который затем преобразуется в плотную 3D-модель.

Для решения задачи множественного обхода OpenMVS использует алгоритм построения графа смежности, который строит граф, связывающий все точки на поверхности объекта с соответствующими изображениями. Затем алгоритм обходит граф, определяя глубину каждой точки на поверхности.

5. Оптимизация 3D-модели

После построения плотной 3D-модели OpenMVS выполняет ее оптимизацию с помощью метода нелинейной оптимизации, который позволяет улучшить качество модели. Этот метод минимизирует различные ошибки, такие как ошибки измерений, ошибки внутрикамерной калибровки и другие. Оптимизация 3D-модели помогает улучшить точность и качество модели.

В заключение, работа OpenMVS состоит в построении плотной 3D-модели на основе набора изображений. Для этого программа использует методы компьютерного зрения, множественный обход и методы оптимизации. В результате работы получается точная и детальная модель объекта, которая может использоваться в различных областях, включая медицину, архитектуру, инженерные и научные исследования.

V. ОПЫТ ВИШ МИФИ В РАЗРАБОТКЕ ПРОГРАММНОГО АППАРАТНОГО КОМПЛЕКСА ПОЛУЧЕНИЯ ФОТОИЗОБРАЖЕНИЙ НА ОСНОВЕ ТЕХНОЛОГИИ IoT

Рассматриваемый инженерный объект включает в себя модули, необходимые для автоматизации съемочного процесса объекта с различных сторон и определения формы, размеров, положения и иных характеристик объектов по их фотоизображениям с последующим преобразованием 3D модель. Данный инженерный объект должен автоматизировать подготовку исходных данных (фотоснимков) объекта, перемещая одну фотокамеру в пространстве вокруг объекта с использованием шагового двигателя. Ранее перемещался либо сам объект съемки, либо производилась съемка несколькими камерами. Рассматриваемый инженерный объект может быть использован не только для снимков внутреннего объекта, но и для внешней съемки окружения, например, при помощи лидара. С помощью данного инженерного объекта будет возможность выявить наиболее оптимальные алгоритмы для различных геометрических объектов и сократить пользователям время на поиск оптимальных методов преобразования объектов в 3D модели. В работе рассматривается деятельность по процессу получения, передачи и накопления серий фотоизображений и анализа точности для различных алгоритмов цифровой генерации 3D моделей для систем автоматизированного проектирования на основе принципа фотограмметрии.

Как и любая другая предпринимательская деятельность, подготовка снимков объекта для создания 3D модели и, непосредственно, создание 3D модели объекта из полученных снимков сочетает в себе множество факторов, напрямую влияющих на его успех.

Ими являются съемка объекта с разных сторон и сохранение фотографий в памяти устройства и обработка полученных фотографий для создания 3D объекта.

Одним из способов обеспечения данного контроля является использование автоматической подготовки фотографий объекта и последующая обработка оптимальным алгоритмом для конкретного типа объекта.

На текущий момент деятельность подготовки данных для создания 3D модели осуществляется вручную – съемка объекта ведется постоянно изменяющимися свое положение сотрудниками и далее происходит ручной перенос полученных фотографий на удаленное устройство для обработки. Затем сотрудниками выполняется обработка различных 3D объектов одним алгоритмом, показывающими

различную точность на разных данных, ручная обработка фотографий, генерация облака точек и наложения текстуры.

Подобный способ реализации деятельности приводит к необходимости большого количества фотокамер для съемки статичного объекта с различных сторон, необходимости постоянного перемещения вокруг объекта, отсутствия возможности автоматически контролировать количество фотографий объекта и автоматически задавать идентифицировать конкретный объект среди фотографий. Дополнительно присутствует проблема автоматической отправки полученных снимков на устройство для их обработки и хранения.

При обработке данных на удаленном устройстве пользователь сталкивается с невозможностью автоматически получить исходные фотографии объекта, страдает от отсутствия удаленного интерфейса управления, вынужден вручную определять положение снимков объекта в пространстве и контролировать наложение текстуры на полученное облако точек. Отсутствие возможности автоматически анализировать точность полученного 3D объекта вынуждает пользователя повторно подготавливать исходные данные и проводить повторную обработку, что сильно влияет на время создание 3D объекта из исходной модели.

Для сокращения количества возникающих проблем было решено изменить текущую реализацию деятельности на более безопасную и продуктивную.

Далее была составлена контекстная диаграмма решения на системном уровне для определения участия системы в рассматриваемой деятельности [20].

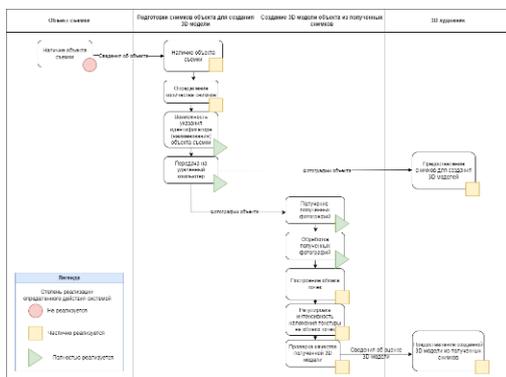


Рис. 13 – Контекстная диаграмма решения на системном уровне

Для работы цифровой модели получения, передачи и накопления серий фотоизображений и анализа точности для различных алгоритмов цифровой генерации 3D моделей для систем автоматизированного проектирования на основе принципа фотограмметрии необходимо устройство съемки объекта на базе Raspberry Pi с установленным программным обеспечением для запуска дополнительных пакетов языка программирования Python и удаленный компьютер для обработки полученного информации и создания 3D объекта с проверкой точности.

Выбранная цифровая модель является инструментом для упрощения получения, передачи и накопления серий и фотоизображений, и анализа

точности для различных алгоритмов цифровой генерации 3D моделей в части:

- упрощения получения, передачи и накопления серий и фотоизображений;
- автоматической установки количества фотографии по вертикали и горизонтали для получения необходимого и достаточного количества снимков объекта;
- возможности ввода уникального имени объекта для набора фотографий с целью однозначной идентификации объекта;
- отправки фотографий на удаленный компьютер для обработки;
- получения фотографий на удаленный компьютер для обработки;
- определения места хранения файлов фотографий на удаленном компьютере;
- определения IP адреса удаленного компьютера;
- регулировки количества фотографий, которые будут участвовать в построении 3D модели;
- коррекции автоматического определения положения снимков в пространстве;
- проверки точности полученной 3D модели.

Был проведен анализ альтернативных варианты изменения разрабатываемого решения. В результате получены следующие варианты:

- Другая реализация установки системы (Метод поворотного стола: объект вращается, камера остается неподвижной);
- Использование видеосъемки вместо снимков для увеличения точности создаваемого 3d-объекта в системе (видеосъемка позволит сделать частое кадрирование, но уступает качеству отдельно взятой фотографии);
- Использование методов машинного обучения и методов оптимизации для автоматического определения количества снимков в зависимости от геометрической формы фотографируемого объекта;
- Изоляция объекта съемки от внешних источников света и использование одного источника освещения, который перемещается вместе с камерой, либо многоступенчатое освещение по всему периметру вокруг объекта;
- Использование облачного хранилища для хранения фотографий вместо прямой пересылки фотографий объекта с Raspberry Pi на внешнее устройство обработки.

На рис. 14 представлена концептуальная схема физической установки разрабатываемой системы.

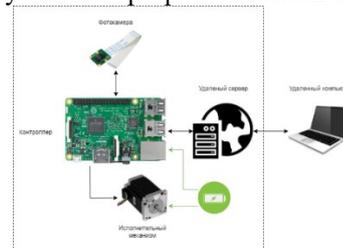


Рис. 14 - Концептуальная схема физической архитектуры разрабатываемой системы

Данная система включает следующие компоненты:
- Фотокамера, которая делает съемку для создания 3D-модели;

- Исполнительный механизм, который вращает установку с фотокамерой и контроллером;
- Контроллер, который управляет работой исполнительного механизма и получает данные с фотокамеры для передачи на удаленный компьютер;
- Удаленный сервер – для управления разрабатываемой установкой и загрузки готового датасета (фотоснимки) на удаленном компьютере для дальнейшего создания 3D-модели.

Для создания системы по описанной концепции необходимо учитывать ряд технических характеристик компонентов: мощность, номинальное напряжение электрических устройств, их интерфейсы, крутящий момент исполнительного механизма и пр.

К основным элементам разрабатываемой системы относятся:

- Плата Raspberry Pi model 3B+ (Рис. 15, А) — это одноплатный компьютер, разработанный основным разработчиком, который работает на чипе Broadcom ARM Cortex-A53 [21]. Это очень маленькая плата, которая может использоваться как мини-компьютер для различных проектов, таких как робототехника, домашняя автоматизация, программирование, медиа-центр и многое другое. Также он оборудован Wi-Fi-модулем с помощью которого можно подключаться к ней удаленно, имеет полноценную операционную систему, а также имеет возможность подключения дополнительных компонентов, такие как камера, дисплей, USB-устройства. Плата имеет 40-пиновые разъемы (GPIO) с помощью которых можно подключать устройства для программирования и работы. Распиновка данных разъемов представлена на рис. 15 под литерой Б. Питается данная плата с помощью micro-USB разъема от источника питания напряжением 5В и требует минимальную силу тока от 2.5А.



Рис. 15 – А. Raspberry Pi model 3B+, Б. Распиновка платы Raspberry Pi model 3B+

- Raspberry Pi camera v2.1 (рис. 16) — это устройство, которое позволяет легко подключить камеру к плате Raspberry Pi и использовать ее для различных проектов, таких как системы видеонаблюдения, робототехника, машинное обучение и многое другое. Данная камера имеет разрешение в 5 мегапикселей, максимальное разрешение фото 2592×1944 пикселей, имеет фиксированное фокусное расстояние [22].



Рис. 16 – Raspberry Pi camera v2.1

- Шаговые двигатели Nema 23 HS8603 (Рис. 17, А) и ШДТ57 (Рис. 17, Б) это устройства, которые преобразует электрическую энергию сети в дискретные механические перемещения. Роторы

данных двигателей имеют шаг вращения в 1,8 градус, который определяется числом поданных импульсов на двигатель. Двигатели имеет 2 обмотки и 6 выводов. Данным двигателям требуется источник постоянного тока (блоки питания) с напряжением, для Nema 23 HS8603 – 20 вольт, а для ШД57 – 36 вольт, а также номинальной мощностью 50 ватт и 150 ватт соответственно. Для взаимодействия данных устройств с платой Raspberry Pi необходимы драйверы двигателей, для Nema 23 HS8603 — ТВ6600 (Рис. 17, В), а для ШД57 – DM556D (Рис. 17, Г). Они используются для контроля вращения шаговых двигателей.

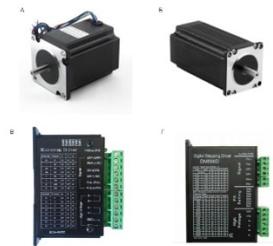


Рис. 17– А. Шаговый двигатель Nema 23 HS8603, Б. Шаговый двигатель модели ШДТ57, В. Драйвер двигателя ТВ6600, Г. Драйвер двигателя DM556D
- Удаленный сервер, который запускается на плате Raspberry Pi и позволяет удаленно управлять установкой для съемки, а также передавать отснятые фотографии на удаленное устройство.

Физическая архитектура системы с учетом подобранных компонентов в виде схемы представлена на рис. 18.

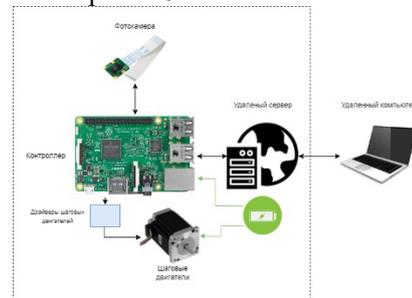


Рис. 18 – Физическая архитектура разрабатываемой системы

К внутренней архитектуре относятся такие элементы как Raspberry Pi camera v2.1, плата Raspberry Pi Model 3B+, шаговые двигатели HS8603 - и ШД57 и блоки питания для питания всех устройств. Они все находятся на установке для съемки объекта.

Далее отобразим основные процессы, которые происходят на физическом уровне системы:

- Одноплатный компьютер Raspberry Pi Model 3B+ запускает удаленный сервер для обмена данными. Она осуществляет съемку и управляет шаговыми двигателями через соответствующие им драйверы;

- Фотокамера крепится на специальное отведенное ей место и присоединена с платой Raspberry Pi через специальный разъем “camera”;

- При помощи удаленного сервера устанавливается связь с удаленным компьютером для обмена данными, а также передачи на него готовых фотоснимков для дальнейшей обработки.

На рис. 19 представлена концептуальная схема логической архитектуры системы.

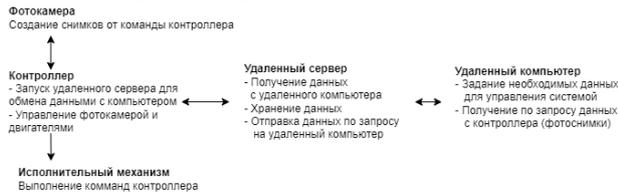


Рис. 19 - концептуальная схема логической архитектуры системы

Она включает в себя следующие компоненты:

- Фотокамера, которая отвечает за создание фотоснимков;
- Контроллер, который, управляет фотокамерой и двигателями, а также запускает удаленный сервер для обмена данными с компьютером;
- Удаленный сервер, который выполняет функции управления данной системой и передачи данных (фотоснимков);
- Удаленный компьютер, на котором задается необходимые данные для управления системой и получения по запросу данных с контроллера (фотоснимки).

В следующих подразделах осуществляется описание подобранных программных решений для основных компонентов и их принцип взаимодействия между собой.

С точки зрения логической организации система состоит из программных модулей, которые отвечают за получение, сбор, хранение и передачу данных. Она включает в себя основные концепции и принципы. За работу разрабатываемой системы отвечает модули клиентской и серверной частей на плате Raspberry Pi.

Данный модуль разрабатывался с помощью Visual Studio Code, который предлагает широкие возможности для разработки и отладки кода в различных языках программирования, включая C++, Java, Python, PHP, JavaScript, TypeScript, Ruby, HTML, CSS и многие другие. VS Code предлагает функциональность, которая делает его одним из самых популярных редакторов среди разработчиков.

В данном модуле использовался язык программирования Python [23], так как Raspberry Pi отлично работает с данным языком.

Данный модуль включает в себя следующие функции:

- а) Управление двигателями и фотокамерой;
- б) Создание, обработка и отправка данных;
- в) Управление и запуск удаленного сервера для синхронизации с удаленным компьютером.

Общая логика работы разрабатываемой системы, следующая:

- 1) Плате Raspberry Pi с запущенным удаленным сервером приходит сигнал на количество итераций (фотографий);
- 2) Плата обрабатывает сигнал, а именно подсчитывает количество фотографий на 1 месте, а также количество итерации (позиций) вокруг объекта, и начинает работу;
- 3) Плата подает сигнал на первый двигатель (движение камеры вниз) и выполняет движение с

остановками для съемки объекта до тех пор, пока не закончатся количество итераций на одной стороне;

4) Затем плата возвращает камеру в исходное состояние при помощи первого двигателя и подает сигнал на второй шаговый двигатель для смены ракурса съемки объекта;

5) Шаги 3-4 повторяются до тех пор, пока установка с камерой не объедет объект вокруг него на 360 градусов с заданными итерациями в шаге 2;

6) Заключительный этапом полученные фотоснимки передаются на удаленный компьютер.

На рис. 20 представлена реализуемая установка разрабатываемого устройства. В основе его лежит:

- Сама установка, которая имеет возможность вращаться на 360 градусов, а также каретка, которая является подвижной на 90 градусов;
- Шаговые двигатели, а также соответствующие им драйвера. На роторах у шаговых двигателей находятся шестеренки диаметром 450 мм для движения по установке по зубчатым рельсам;
- Плата и камера крепятся на подвижной каретке в защищенном корпусе (рис. 21);



Рис. 20 – Установка реализуемой системы для съемки фотоизображений



Рис. 21 – Подвижная каретка с установленной камерой и платой в защищенном корпусе

Для реализации программной составляющей системы использовались следующие библиотеки:

- RPi.GPIO (Библиотека для языка программирования Python, которая позволяет управлять вводом-выводом на Raspberry Pi. Она предоставляет простой и удобный интерфейс для работы с GPIO-штырьками на Raspberry Pi, что делает ее одной из наиболее популярных библиотек для работы с GPIO на этой платформе. RPi.GPIO также поддерживает дополнительные устройства, такие как шины данных и протоколы связи, такие как I2C и SPI. Это делает ее очень гибкой библиотекой, которая может быть использована для многих проектов, использующих Raspberry Pi) [24];

- Flask (Библиотека для веб-приложений на языке Python. Он предоставляет минимальный набор инструментов для создания веб-приложений, не навязывая своей архитектуры. Flask предоставляет множество инструментов, которые упрощают создание веб-приложений, включая поддержку маршрутизации (routing), обработки запросов (request handling), шаблонизации (template engine), работу с базами данных (database integration) и многое другое. С помощью Flask можно создавать различные веб-приложения: от простых сайтов до сложных приложений, работающих с базами данных, авторизацией и т.д.) [25];

- Subprocess (Библиотека используется для запуска дочерних процессов или выполнения команд в различных операционных системах. Она позволяет взаимодействовать с выполненными процессами, получать их вывод и статус завершения. Библиотека subprocess содержит много различных методов для выполнения процессов, в том числе run(), call(), check_output(), Popen(). Кроме того, эта библиотека позволяет создавать трубы для передачи данных между процессами и многое другое) [26]. В первую очередь функция moves () принимает на вход 3 переменные, которые задаются в серверной части, про которую будет написано в серверной части разрабатываемой системы:

- Positions – Целое число (int), которое указывает количество остановок (позиций) вокруг объекта;

- ColPhoto – Целое число (int), которое указывает количество остановок для съемки на одной позиции;

- ObjectName – Текстовый тип данных (string), для указания названия фотографируемого объекта.

После получения основных переменных идет инициализация программы, а именно: задается постоянное число шагов для первого и второго двигателя, которые считались следующим образом:

- 1) Двигатель за один шаг проходит примерно 0,005 градуса;

- 2) Далее с помощью пропорции считаем необходимое количество шагов для двигателей (12100 шагов для первого двигателя и 79000 шагов для второго двигателя).

Далее продолжает проходить инициализация программы, а именно установка пинов для подачи сигналов для драйверов шаговых двигателей. Рассмотрим основных 3 сигнала для драйверов:

- PUL (pulsation) — это импульс, который указывает двигателю, когда нужно совершать шаг. Каждый импульс приводит к шаговому движению на заданное расстояние;

- DIR (direction) — это сигнал, который указывает направление движения двигателя. При подаче сигнала DIR в одном направлении, двигатель будет вращаться в одну сторону, а при подаче в другом - в другую;

- ENA (enabled) - это сигнал, который разрешает или запрещает работу двигателя. При подаче сигнала ENA, двигатель начнет или прекратит вращаться в зависимости от сигнала DIR. При отсутствии сигнала ENA, двигатель не будет реагировать на сигналы PUL и DIR.

Следующим шагом выставляем задержку (delay), которая нужна того, чтобы ensure (гарантировать)

стабильное и плавное движение двигателя. Каждый проходящий пульс (по сигналу PUL) запускает определенное количество шагов в двигателе, и время задержки между этими пульсами влияет на скорость и плавность движения двигателя. Если задержка между пульсами слишком мала, двигатель может двигаться слишком быстро и не стабильно, что может привести к потере шагов или сбою двигателя. В то же время, если задержка слишком большая, двигатель будет двигаться очень медленно, что также может привести к ошибкам в работе.

Затем идет подсчет шагов для одной итерации для шаговых двигателей, например мы назначили 6 позиций вокруг объекта, это значит, что нужно сделать $79000 / 6 = 13167$ шагов за одну итерацию.

Далее были назначены следующие 4 функции, которые далее будем вызывать:

- move_down_fm() – Движение каретки (первого двигателя) вниз;

- move_up_fm() – Движение каретки (первого двигателя) вверх в исходное состояние;

- move_sm() – Движение установки съемки (второй двигатель) в другое положение;

- take_photo() – Модуль съемки фотографий, который выполняется с помощью библиотеки subprocess.

После чего происходят следующие операции:

- Создаем первую фотографию в папке, которая создается с введенным названием (ObjectName);

- Далее даём команду на движение каретки, которая движется с подсчитанными шагами, останавливается и делает фотографию, и она выполняется до тех пор, пока число количество остановок для съемки не станет нулевым значением;

- После этого возвращаем каретку в исходное положение и начинается движение платформы вокруг объекта;

- Предыдущие два шага выполняются до тех пор, пока число, указывающее количество остановок (позиций) вокруг объекта не станет нулем (то есть платформа должна вернуться в исходное состояние);

В конечном итоге, на плате Raspberry Pi сохраняется все фотографии в одной папке.

Передача готовых фотографий выполняется с помощью утилиты scp (secure copy) — это командная утилита UNIX, используемая для копирования файлов между локальной и удаленной машинами через защищенный канал по протоколу SSH (secure shell).

По сути, scp работает так же, как стандартная утилита cp, но добавляет возможность защищенной передачи файлов между компьютерами через зашифрованный канал. С помощью scp можно копировать файлы как с локальной машины на удаленную, так и наоборот.

Утилита scp имеет простой синтаксис, и ее можно использовать для передачи файлов не только между двумя удаленными машинами, но и между удаленной и локальной машинами.

Пример использования команды scp для копирования файла с локальной машины на удаленную машину:

```
scp /path/to/local/file
user@remote:/path/to/remote/directory,
```

где /path/to/local/file - путь к копируемому файлу на локальной машине, user - имя пользователя на удаленной машине, remote - IP-адрес или имя удаленной машины, /path/to/remote/directory - путь к целевой директории на удаленной машине.

VI. ОГРАНИЧЕНИЯ И АЛЬТЕРНАТИВНЫЕ ПОДХОДЫ К РЕШЕНИЮ РЕАЛИЗАЦИИ

Согласно первоначальному плану, в первом прототипе системы использовались два шаговых двигателя Nema 23 HS8603. Однако в ходе тестирования было выявлено, что этот тип двигателя не обладает достаточной мощностью для поднятия каретки вверх. В результате этого было принято решение заменить Nema 23 на ШД 57, более мощный тип шагового двигателя.

Также в первом прототипе системы имелась гипотеза использовать более меньшую по производительности плату (или в паре с Raspberry Pi), а именно Arduino Uno WiFi Rev. 2 (Рис. 22).



Рис. 22 - плата Arduino Uno WiFi Rev. 2

С помощью неё помимо управления шаговыми двигателями, есть возможность мерить угол наклона с помощью встроенного в плату акселерометра. В свою очередь трехосевой акселерометр выводит значения поворота по осям x, y, z от 0 до 1, далее требуется посчитать угол наклона платы в пространстве при помощи сферической системы координат, которая является система координат в трехмерном пространстве, в которой каждая точка задается расстоянием ее положения от начала координат, сферическим (полярным) углом и азимутальным углом [27].

В начальной позиции положение устройства такое, при котором оси x и y находятся в плоскости горизонта, а ось z ортогональна осям x и y (Рис. 23).

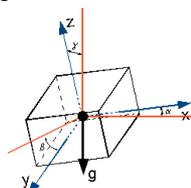


Рис. 23 – измерение угла поворота платы

В начальной позиции, когда сила гравитации действует только на ось z, получим, что все значения углов равны 0. При этом, значения угла может быть вычислена по следующей формуле:

$$\alpha = \arctan\left(\frac{A_x}{\sqrt{A_y^2 + A_z^2}}\right)$$

В последствии экспериментов было выявлено то, что данный акселерометр имеет погрешность в 20-30 градусов впоследствии чего было решение отказаться от него.

Последний эксперимент, который был проведен в рамках реализуемой системы – это использование

компаса для измерения поворота вращающиеся установки. Для этого использовалась плата Arduino Uno WiFi Rev. 2 и подключенный модуль компаса HMC5883L. HMC5883L (Рис. 24) — это модуль компаса, работающий на основе магнитных датчиков [28]. Он может измерять магнитные поля в трех направлениях и давать данные о текущем направлении.



Рис. 24 – модуль компаса HMC5883L

Чтобы получить измерения в градусах, необходимо использовать формулу пересчета значений, получаемых из модуля. Значения, которые получены из модуля, измеряются в Гауссах. Чтобы перевести значения в градусы, необходимо использовать следующую формулу:

$$\alpha = \text{atan2}(Y, X) * (180 / \pi)$$

Здесь atan2 — это функция арктангенса, а PI - число Пи. Результат, полученный по этой формуле, будет показывать текущий угол в градусах.

В результате, удалось сделать измерения компаса, но так как модуль находился на металлической установке, а также компас находился внутри здания, то его значения имели огромную погрешность (+50 градусов), в результате этого было решение отказаться от этого модуля.

Также можно отметить некоторые следующие ограничения разрабатываемой системы:

- Максимальные габариты фотографируемого объекта должны быть не более 20 сантиметров по ширине, длине и высоте;
- При съемке объекта избегать прямых солнечных лучей и теней;
- Избегать однотонных или блестящих/отражающих поверхностей.

Если рассматривать дальнейшее развитие разрабатываемой системы, то можно выделить следующие дополнения и улучшения:

1) Улучшение камеры для съемки, например на Raspberry Pi High Quality Camera (Рис. 25) — это камера высокого качества, разработанная специально для использования с одноплатным компьютером Raspberry Pi [29]. Она позволяет снимать изображения разрешением до 12,3 мегапикселей, а также видео с разрешением до 4К при 30 кадрах в секунду. Камера оснащена сменной оптикой, что позволяет пользователю выбрать подходящую оптику для конкретного проекта. Камера имеет высокую чувствительность и широкий динамический диапазон, что позволяет получать изображения с хорошей детализацией и насыщенными цветами. Таким образом, с помощью данной камеры будет возможность делать более качественные изображения, а также с помощью широкоугольного объектива увеличатся габариты фотографируемого объекта;



Рис. 25 – камера Raspberry Pi High Quality Camera с объективом

2) Использование облачного хранилища данных для хранения фотографируемых объектов. Данное решение имеет множество преимуществ:

- о Надежность сохранения данных. Облачные хранилища обычно обеспечивают более высокий уровень надежности и безопасности данных, чем локальные хранилища, так как они обладают распределенной системой хранения данных и репликацией для обеспечения резервного копирования данных;

- о Доступность данных. Облачные хранилища позволяют получить доступ к данным из любого места, где есть доступ к Интернету, что делает их более удобными для работы с другими людьми, совместной работы над проектами и обмена фотографиями;

- о Удобство использования. Облачные хранилища обычно имеют хороший пользовательский интерфейс и простой в использовании, позволяя легко загружать, скачивать и просматривать фотографии;

- о Расширяемость. Облачные хранилища позволяют легко расширять хранилища данных по мере необходимости, без необходимости покупать новое оборудование и увеличивать затраты на обслуживание;

- о Стоимость. Использование облачного хранилища может быть более выгодным, чем содержание собственного локального сервера или покупка новых жестких дисков для хранения фотографий.

Таким образом, при использовании облачного хранилища данных будет возможность взять готовый для дальнейшей обработки датасет (снимки) с любого компьютера, имеющего доступ в интернет;

3) Объединение (расширение) программно-аппаратного комплекса, а именно возможность быстро получить 3д модель при помощи методов фотограмметрии в нужном расширении. Таким образом, это является эффективным решением для получения 3D-моделей в нужном расширении. В результате повышается точность, скорость, производительность и качество системы, а также становится возможным создание более доступных и гибких систем.

4) Возможность дополнить программно-аппаратный комплекс дополнительными модулями, такими как лидар или другими модулями для улучшения съемки;

5) Использование хромакея и дополнительного освещения для улучшения дальнейшей обработки создаваемой 3д модели. Данное дополнение имеет следующие преимущества:

- о Лучшее разделение объектов и фона. При использовании хромакея, однотонный фон значительно облегчает процесс отделения объектов от фона. Дополнительное освещение обеспечивает

лучшую видимость объектов и повышает контрастность;

- о Более точная коррекция цвета. Использование хромакея позволяет сделать более точную коррекцию цвета, поскольку фон является однотонным;

- о Уменьшение мерцания и шума. Дополнительное освещение и хромакей обеспечивают более стабильное освещение, что может уменьшить мерцание и шум на изображении;

- о Увеличение точности и скорости. Отделение объекта от фона и более точная коррекция цвета позволяют увеличить точность при создании 3D-модели. Это также может сократить время, затрачиваемое на обработку изображения и моделирование;

- о Создание качественного фона. Хромакей и дополнительное освещение позволяют создавать качественный фон для 3D-модели, который может быть использован в дальнейшем для различных целей.

Таким образом, использование хромакея и дополнительного освещения при создании 3D-моделей может значительно улучшить качество и точность создаваемой модели, а также сократить время на обработку. Также это позволяет создавать качественный фон для моделей, который может быть использован в дальнейшем при различных проектах;

В результате, данный разработанный программно-аппаратный комплекс имеет огромные перспективы развития и будет в дальнейшем дополняться и улучшаться по мере возможностей.

VII. СРАВНИТЕЛЬНЫЙ АНАЛИЗ ТОЧНОСТИ АЛГОРИТМОВ 3D ОБРАБОТКИ ИЗОБРАЖЕНИЙ

Авторы статьи «Application of Areal Change Detection Methods Using Point Clouds Data» исследовали метод многократного лазерного сканирования, который позволяет определять деформации объектов, и применили его к дамбе в Болгарии [30]. В этой же статье авторы выделяют следующие методы для сравнения облаков точек:

- Прямое (cloud-to-cloud) сравнение облаков точек с помощью метода ближайшей точки
 - о 7-параметрическое 3D преобразование
 - о Алгоритм Iterative Closest Point
 - о Точки регулярной сетки
- Cloud-to-mesh расстояние or cloud-to-model расстояние
 - о Использование плоской аппроксимации
 - о Оценка вектора нормали
 - о Метод базовой линии
- Mesh-to-mesh (model-to-model)
- Цифровая модель поверхности (DSM) - не может справиться с нависающими частями (утесами, обрушениями банка, большими блоками)
- Расчет объема
- Точки регулярной сетки

При этом авторы выделяют:

- о Distances between two point clouds (cloud-cloud distances),
- о Distances between point cloud and surface (cloud-mesh distances)

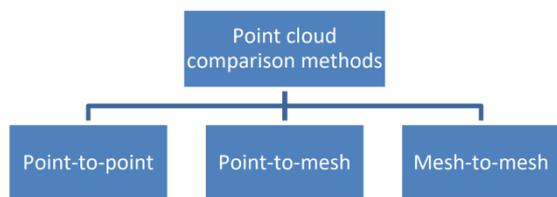


Рис. 26 - Point cloud comparison methods

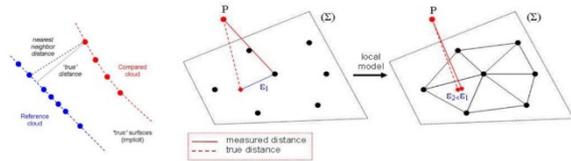


Рис. 27 - Поверхностная модель для расстояний

В статье Three Dimensional Change Detection Using Point Clouds: A Review [31] авторами рассматриваются следующие методы для сравнения облаков точек:

o Cloud-to-cloud comparison (C2C) – использует Евклидово расстояние между конкретными точками и применяет метод ближайших соседей на финальной итерации. Процесс проиллюстрирован на рис. 28.

o Cloud-to-Mesh comparison (C2M) - использует Евклидово расстояние между конкретными точками и ближайшую грань (или к ребру, если ортогональная проекция точки не попадает ни на одну грань) в целевой сетке триангулированного облака точек.

o Multiscale Model to Model Cloud Comparison (M3C2) - Этот метод оценивает перемещение на основе найденных интересных точек в каждом облаке точек (например, подвыборка точек облака или даже всего исходного облака точек). Он состоит из двух основных шагов: (1) оценки нормали поверхности и ориентации в 3D-пространстве в масштабе, согласующемся с местной шероховатостью поверхности; (2) измерения среднего изменения поверхности в направлении нормали с явным расчетом локального доверительного интервала (σ). Главным преимуществом этого метода является то, что он работает напрямую с облаками точек, без необходимости сетчатого или сеточного представления. Он оценивает доверительный интервал для каждого расстояния m

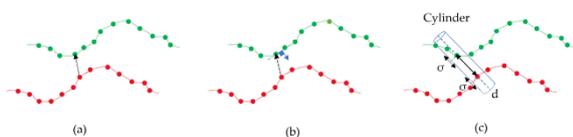


Рис. 28 – (a) - Cloud-to-cloud comparison (C2C), (b) - Cloud-to-Mesh comparison (C2M), (c) - Multiscale Model to Model Cloud Comparison (M3C2)

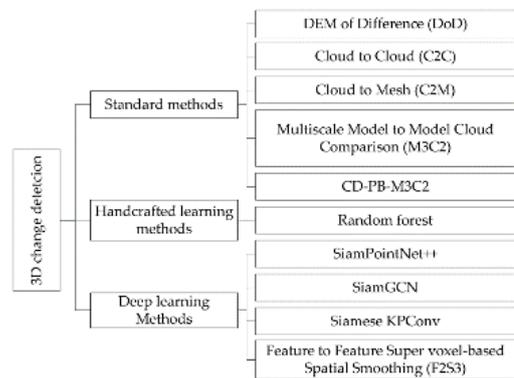


Рис. 29 – методы сравнения облаков точек
Все объекты, используемые в качестве датасетов, представлены в таблице 3.

Таблица 3 – Объекты используемые для датасетов

Номер набора данных	Наименование датасета	Тип объекта	Описание объекта	Сложность обработки
1	Cube	Куб	Простой объект	Легкая
2	Conus	Конус	Простой объект	Легкая
3	Cilindr	Цилиндр	Простой объект	Легкая
4	Sphere	Сфера	Простой объект	Легкая
5	Trapeciya	Тетраэдр	Простой объект	Легкая
6	Blue Sponge	Квадрат	Цветной Текстурированный объект	Легкая
7	Color Tea	Параллелепипед	Цветной Текстурированный объект	Средняя
8	Karich	Сфера	Цветной Текстурированный объект	Средняя
9	Glass Mag	Сложный элемент	Сочетание стекла и прозрачной текстуры	Высокая

На рис. 30 представлен пример простейших геометрических объектов с монотонной текстурой



Рис. 30 – Простейшие геометрические фигуры
Обработка данных для пяти шести наборов объектов является легкой, так как они являются простыми

геометрическими фигурами. Обработка шестого, седьмого и восьмого наборов данных немного сложнее, так как эти объекты имеют более сложную форму и текстуру разного цвета. Работа с девятым набором данных является наиболее сложной, так как он представляет из себя более сложный элемент, сочетающий в себе стеклянную поверхность более прозрачной текстуры.

Для автоматизации процесса фотографирования объектов был использован программно-аппаратный комплекс получения снимков объекта.

В центре полусферы в основании устанавливался конкретный объект и далее осуществлялся процесс съемки 50 фотографий объекта с разных сторон. Камера автоматически перемещалась по вертикали и горизонтали вокруг объекта.



Рис. 31 – Программно-аппаратный комплекс

Процесс отправки полученных снимков на удаленный компьютер производился с помощью визуального интерфейса, позволяющего задать количество остановок камеры по вертикали и горизонтали, а также наименование объекта съемки для его однозначной идентификации в дальнейшем.

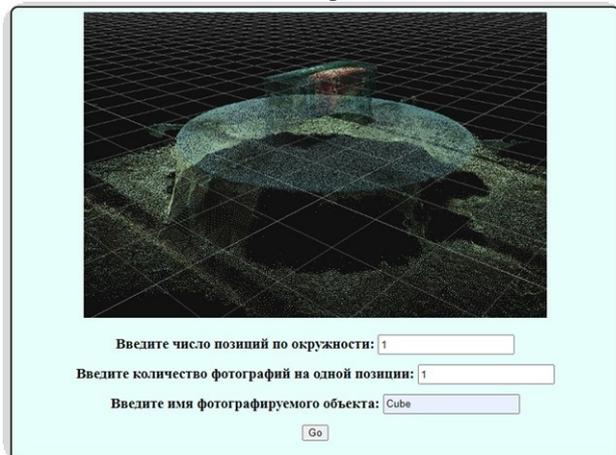


Рис. 32 – Интерфейс программно-аппаратного комплекса

Далее 50 фотоснимков каждого объекта были готовы для обработки в программах с использованием различных алгоритмов фотограмметрии.



Рис. 33 – Пример хранения датасетов для одного объекта съемки

В качестве тестируемых в исследовании программ были выбраны следующее программное обеспечение:

- 3D Zephyr
- Agisoft Metashape
- MeshRoom
- RealityCapture
- Pix4D
- COLMAP

В исследовании представлены шесть программ для обработки данных с помощью фотограмметрии. Некоторые из этих программ могут выполнять предварительную подготовку данных перед обработкой.

В качестве сравнения были использованы:

- Разные варианты освещения
- Простейшие геометрические фигуры
- Фигуры с более сложной геометрией
- Цветные фигуры

Процесс обработки представлен на рис. 34.

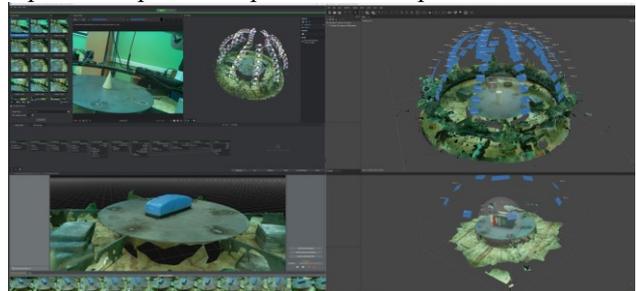


Рис. 34 – Процесс обработки

После обработки всех 9-ти датасетов в каждой из 6 программ в качестве эталонных моделей были выбраны результаты обработки в программе 3D Zephyr. В качестве сравнения было решено произвести анализ облаков точек в форматах .ply, которые были предварительно экспортированы из каждой программы. В качестве сравнения использовались следующие метрики, используемые для сравнения результатов облаков точек (.ply):

- Евклидово расстояние
- Расстояние Хаусдорфа
- L1-норма (Минковского)
- Коэффициент корреляции

Для автоматизации расчета написан программное обеспечение на языке Python.

Для сравнения PLY файлов могут быть использованы различные математические метрики, в зависимости от потребностей конкретной задачи. Некоторые из наиболее распространенных метрик включают в себя:

1. Евклидово расстояние: Это расстояние между двумя точками в трехмерном пространстве, которое

определяется как корень квадратный суммы квадратов разницы координат по каждой оси. Оно может быть использовано для сравнения координат вершин моделей.

2. Расстояние Хаусдорфа: Это метрика, которая измеряет расстояние между двумя наборами точек в трехмерном пространстве. Оно может быть использовано для измерения характеристик формы моделей.

3. L1-норма: Эта метрика, также известная как расстояние Манхэттенское расстояние, используется для измерения расстояния между двумя наборами точек, которое определяется как сумма абсолютных разностей между соответствующими координатами точек.

4. Расстояние Вассерштейна: Эта метрика используется для оценки расстояния между двумя распределениями вершин моделей. Она может использоваться для сравнения цветowych гистограмм PLY файлов [32].

5. Коэффициент корреляции: Эта метрика используется для выявления связей между двумя наборами данных. Она может быть использована для сравнения свойств вершин моделей.

Есть и другие метрики, которые могут быть использованы для сравнения PLY файлов в зависимости от целей и заданий.

Общий алгоритм работы для такой написанной программы:

1. Загрузить два PLY файла в программу.
2. Использовать библиотеки для чтения PLY файлов.
3. Преобразовать координаты вершин каждого файла в соответствующие наборы точек в трехмерном пространстве.
4. Применить выбранную метрику (например, евклидово расстояние) для сравнения двух наборов точек.
5. Определить, насколько большее значение метрики сравнения показывает, насколько сильно различаются два PLY файла, с учётом порогового значения для данной метрики.

Код для реализации этого алгоритма будет зависеть от выбранных библиотек для чтения PLY файлов и математической библиотеки для вычисления метрик. 3D объекты после обработки представлены на рис. 35.



Рис. 35 – 3D объекты после обработки

Сравнительный анализ простых геометрических объектов представлен на рис. 36.

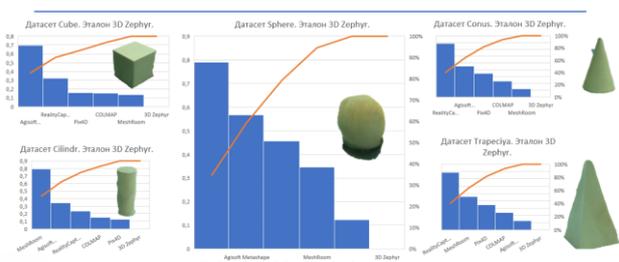


Рис. 36 – Сравнительный анализ простейших геометрических объектов

Сравнительный анализ сложных геометрических объектов представлен на рис. 37.

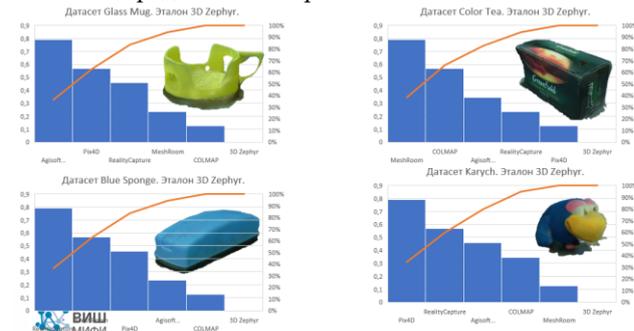


Рис. 37 – Сравнительный анализ сложных геометрических объектов

VIII. ЗАКЛЮЧЕНИЕ

В результате работы был создан программно-аппаратный комплекс, осуществляющий фотосъемку и достигнуто повышение точности генерации 3D модели за счет комбинирования алгоритмов обработки для различных типов объектов.

Процесс съемки объекта был осуществлен с использованием одной фотокамеры, позиционирование которой осуществлялось с помощью шаговых двигателей. Управление шаговыми двигателями контролировалось с помощью драйверов, которые, в свою очередь, управлялись с использованием платы Arduino Uno WiFi, замененной впоследствии на Raspberry Pi. Переход на более производительное оборудование был совершен для обеспечения передачи серий фотографий на удаленный компьютер для последующей обработки. Перед каждой серией съемки выполнялась установка количества снимков по вертикали и горизонтали и идентификатора для всех фотоснимков одного объекта. Позиционирование камеры в пространстве производилось по шагам двигателя в зависимости от количества фотографий. От позиционирования с использованием акселерометра и компаса было принято решение отказаться ввиду значительной погрешности (+20 градусов).

Полученные в ходе анализа результаты сравнения алгоритмов обработки позволят оптимизировать подбор программного обеспечения для преобразования реальных объектов в 3D-модель методом фотограмметрии и повысить точность за счет комбинирования наиболее подходящих методов для различных типов объектов от простейших геометрических фигур до реальных машиностроительных деталей.

В работе были исследованы существующие методы генерации облаков точек, наложения текстуры, такие

как SfM (Structure From Motion), MVS, Стереофотограмметрия, TLS (Terrestrial Laser Scanning), SfM и TLS, а также были исследованы существующие методы сравнения двух облаков точек и выполнен сравнительный анализ методов фотограмметрической обработки с целью выявить наиболее качественный метод обработки при сравнении двух облаков точек. Идеальным облаком точек считалось то, которое имело наименьшую разницу в сравнении с эталоном. В качестве эталонного 3D объекта были выбраны результаты программы 3D Zephyr.

В качестве тестируемых в исследовании программ было выбрано следующее программное обеспечение: 3D Zephyr, Agisoft Metashape, MeshRoom, RealityCapture, Pix4D, COLMAP.

В конечном итоге, приведены результаты обработки, полученные с помощью различных методов фотограмметрии. В исследовании отражены особенности, присущие трехмерной реконструкции, такие как, освещение, трудности доступа к отдельным частям объекта, разрешение снимков и стеклянные поверхности.

БЛАГОДАРНОСТИ

Авторы выражают благодарность преподавателю ВИШ МИФИ Жабицкому М.Г., за помощь в разработке технологии, а также выражают благодарность Высшей инженеринговой школе НИЯУ МИФИ, в частности, Бойко О.В, за помощь в возможности опубликовать результаты выполненной работы.

Данное исследование использует результаты научно-исследовательских работ Мочкина К.А, в частности результаты исследования «Разработка ПО трехмерного сканирования на базе 2D лидара для работы с облачной средой по технологии IoT, интегрированного с системой BIM-проектирования Revit», Замчалова А.Б. в ходе работы над исследованием «Разработка на основе технологии IoT программно-аппаратного комплекса получения, передачи и накопления серий фотоизображений для генерации 3D моделей для систем автоматизированного проектирования на основе принципа фотограмметрии», а также Конькова В.В. в исследовании «Анализ точности генерации 3D моделей для систем автоматизированного проектирования на основе принципа фотограмметрии для различных алгоритмов цифровой обработки и режимов».

БИБЛИОГРАФИЯ

1. Краснопецев Б.В. Фотограмметрия. - М.: УПП "Репрография" МИИГАиК, 2008. - 160 с.
2. В.А. Клячин, А.А. Широкий Триангуляция Делоне многомерных поверхностей Вестник СамГУ — Естественнонаучная серия. 2010.
3. Building a Photogrammetry Library. — Текст: электронный // Artstation [электронный ресурс]. URL: <https://www.artstation.com/blogs/olegroenbaek/L1by/building-a-photogrammetry-library> (дата обращения: 24.04.2023).
4. Testing the Artec Ray 3D Scanner. - Текст: электронный // 3Dnatives [электронный ресурс]. — URL: <https://www.3dnatives.com/en/3dnatives-lab-testing-the-artec-ray-3d-scanner-290620214/#> (дата обращения: 03.05.2023).
5. Аврунев Е. Опритова О. ОЦЕНКА ТОЧНОСТИ 3D-МОДЕЛЕЙ, ПОСТРОЕННЫХ С ИСПОЛЬЗОВАНИЕМ БЕСПИЛОТНЫХ АВИАЦИОННЫХ СИСТЕМ Вестник СГУГиТ, Том 23, № 3, 2018
6. Михеева А.А., Ялтыхов В.В., Парадня П.Ф. СОЗДАНИЕ ОРТОФОТОПЛАНА В ПРОГРАММЕ AGISOFT PHOTOSCAN 2021
7. Photogrammetry: Step-by-Step Guide and Software Comparison [Электронный ресурс]. — URL: <https://formlabs.com/blog/photogrammetry-guide-and-software-comparison/> (дата обращения 01.05.2023)
8. AutoDesk Recap Tutorial [Электронный ресурс]. — URL: <https://www.autodesk.com/support/technical/product/recap> (дата обращения 01.05.2023)
9. RealityCapture Tutorial [Электронный ресурс]. — URL: <https://www.capturingreality.com/RealityCapture-Tutorials> (дата обращения 01.05.2023)
10. Agisoft Metashape Tutorial [Электронный ресурс]. — URL: <https://www.agisoft.com/support/tutorials/> (дата обращения 01.05.2023)
11. AliceVision Meshroom Tutorial [Электронный ресурс]. — URL: <https://meshroom-manual.readthedocs.io/en/latest/tutorials/sketchfab/sketchfab.html> (дата обращения 01.05.2023)
12. 3DF Zephyr Tutorial [Электронный ресурс]. — URL: <https://www.3dflow.net/technology/documents/3df-zephyr-tutorials/> (дата обращения 01.05.2023)
13. Colmap Tutorial [Электронный ресурс]. — URL: <https://colmap.github.io/tutorial.html> (дата обращения 01.05.2023)
14. Regard3D Tutorial [Электронный ресурс]. — URL: <https://www.regard3d.org/index.php/documentation/tutorial> (дата обращения 01.05.2023)
15. Jacob A. Morgan & Daniel J. Brogan How to VisualSFM Department of Civil & Environmental Engineering Colorado State University Fort Collins, Colorado January 2016 [Электронный ресурс]. — URL: https://d32ogoqmya1dw8.cloudfront.net/files/getsi/teaching_materials/high-rez-topo/visual_sfm_tutorial.pdf (дата обращения 01.05.2023)
16. iWitness Tutorial [Электронный ресурс]. — URL: <https://iwitness.usc.edu/features> (дата обращения 01.05.2023)
17. Photomodeler Tutorial [Электронный ресурс]. — URL: <https://www.photomodeler.com/tutorial-videos/> (дата обращения 01.05.2023)
18. Qlone Tutorial [Электронный ресурс]. — URL: <https://www.qlone.pro/faq> (дата обращения 01.05.2023)
19. Image-Based Localization in 3D Point Clouds [Электронный ресурс]. — URL: <https://mediatum.ub.tum.de/doc/1624887/1624887.pdf> (дата обращения 01.05.23)
20. Косяков А., Свит У. и др. Системная инженерия. Принципы и практика. Пер. с англ. под ред. В.К. Батоврина. — М.: ДМК Пресс, 2014. — 624 с.: ил.
21. Макаров С. Л. Arduino Uno и Raspberry Pi 3: от схемотехники к интернету вещей. - М.: ДМК Пресс, 2018. - 204 с.
22. Камера для Raspberry Pi «Модель D». — Текст: электронный // Amperka: [электронный ресурс]. — URL: <http://wiki.amperka.ru/products/camera-raspberry-pi-model-d> (дата обращения: 15.04.2023).
23. Python 3 Документация [Электронный ресурс]. — URL: <https://docs.python.org/3/tutorial/> (дата обращения: 01.05.2023)
24. Raspberry Pi GPIO. — Текст: электронный // RPi Labs [электронный ресурс]. — URL: <https://rpi.science.uoit.ca/lab/gpio/> (дата обращения: 04.04.2023).
25. Flask: Quickstart. — Текст: электронный // flask-docs [электронный ресурс]. — URL: <https://flask-docs.readthedocs.io/en/latest/quickstart/> (дата обращения: 06.04.2023).
26. Модуль subprocess — Работаем с процессами. - Текст: электронный // python-scripts [электронный ресурс]. — URL: <https://python-scripts.com/subprocess> (дата обращения: 10.04.2023).

27. Определение угла наклона акселерометром. — Текст: электронный // Bitaks: [электронный ресурс]. — URL: <http://bitaks.com/resources/inclinometer/content.html> (дата обращения: 10.03.2023).
28. How to use HMC5883L Compass Sensor. — Текст: электронный // miliohm: [электронный ресурс]. — URL: <https://miliohm.com/how-to-use-hmc5883l-compass-sensor/> (дата обращения: 10.03.2023).
29. Raspberry Pi High Quality Camera — Текст: электронный // Amperka: [электронный ресурс]. — URL: <http://wiki.amperka.ru/products/raspberry-pi-high-quality-camera> (дата обращения: 03.05.2023).
30. Gergana Antova Application of Areal Change Detection Methods Using Point Clouds Data IOP Conference Series Earth and Environmental Science 2019
31. Citation: Kharroubi, A.; Poux, F.; Ballouch, Z.; Hajji, R.; Billen, R. Three Dimensional Change Detection Using Point Clouds: A Review. *Geomatics* 2022, 2, 457–485. <https://doi.org/10.3390/geomatics2040025>
32. K. Kawano, S. Koide and T. Kutsuna, "Learning Wasserstein Isometric Embedding for Point Clouds," 2020 International Conference on 3D Vision (3DV), Fukuoka, Japan, 2020, pp. 473-482, doi: 10.1109/3DV50981.2020.00057.

Статья получена 17 июля 2023

Работа выполнена в рамках научно-исследовательских и опытно-конструкторских работ (НИОКР) по темам: «Разработка на базе технологии IoT АПК получения, передачи и накопления серий фотоизображений для генерации 3D моделей для систем автоматизированного проектирования с применением фотограмметрии» и «Анализ точности генерации 3D моделей для систем автоматизированного проектирования на основе принципа фотограмметрии для различных алгоритмов цифровой обработки»

Коньков В.В. – инженер Высшей инженеринговой школы Института ядерного физики и технологий Национального исследовательского института «МИФИ» (e-mail: vlad.konkov.7145@gmail.com)

Замчалов А.Б. – инженер Высшей инженеринговой школы Института ядерного физики и технологий Национального исследовательского института «МИФИ». (e-mail: andreizam@yandex.ru)

Жабицкий М.Г. – заместитель директора Высшей инженеринговой школы Института ядерного физики и технологий Национального исследовательского института «МИФИ». (e-mail: jabitsky@mail.ru)

Software and hardware complex for obtaining photographic images based on IIOT technology and accuracy analysis of various algorithms for digital generation of 3D models based on the principle of photogrammetry

V.V. Konkov, A.B. Zamchalov, M.G. Zhabitsky

Abstract - The authors considered the methods of obtaining, transmitting and accumulating photographs based on the Internet of Things technology and analyzed the accuracy of various algorithms for digital generation of 3D models based on the principle of photogrammetry. These methods were used to develop a software and hardware complex that allows shooting objects from different sides for the subsequent conversion of the obtained photographs into three-dimensional objects for further comparative analysis of various algorithms for digital generation of 3D images. The features of using various methods of positioning stepper motors to move the camera around the subject were studied. In the course of the work, a software and hardware complex was developed for automated measurements of relatively small objects outside and inside the room using both photofixation and lidar scanning.

The resulting photographs of geometric shapes of varying complexity were transmitted to a remote computer and processed by various algorithms.

The aim of the study was to automate the end-to-end process of obtaining a series of photographs of objects from different angles and subsequent generation of 3D models of the desired quality by combining different methods and algorithms for processing a series of photographic images.

Keywords: Internet of things, stepper motors, photogrammetry

REFERENCES

1. Krasnopevtsev B.V. Photogrammetry. - M.: UPP "Reprography" MIIGAIK, 2008. - 160 p.
2. V.A. Klyachin, A.A. Shirokiy Delaunay triangulation of multidimensional surfaces Bulletin of the Samara State University — Natural Science Series. 2010.
3. Building a Photogrammetry Library. — Text: electronic // Artstation [electronic resource]. URL: <https://www.artstation.com/blogs/olegroenbaek/L1by/building-a-photogrammetry-library> (accessed: 04/24/2023).
4. Testing the Artec Ray 3D Scanner. - Text: electronic // 3Dnatives [electronic resource]. — URL: <https://www.3dnatives.com/en/3dnatives-lab-testing-the-artec-ray-3d-scanner-290620214/#> (date of reference: 03.05.2023).
5. Avrunev E. Opritova O. EVALUATION OF THE ACCURACY OF 3D MODELS BUILT USING UNMANNED AIRCRAFT SYSTEMS Bulletin of the SSUGiT, Volume 23, No. 3, 2018
6. Mikheeva A.A., Yaltykhov V.V., Paradnya P.F. CREATING AN ORTHOPHOTOPLANE IN THE AGISOFT PHOTOSCAN 2021 PROGRAM
7. Photogrammetry: Step-by-Step Guide and Software Comparison [Electronic resource]. — URL: <https://formlabs.com/blog/photogrammetry-guide-and-software-comparison/> (accessed 01.05.2023)
8. AutoDesk Recap Tutorial [Electronic resource]. — URL: <https://www.autodesk.com/support/technical/product/recap> (accessed 01.05.2023)
9. RealityCapture Tutorial [Electronic resource]. — URL: <https://www.capturingreality.com/RealityCapture-Tutorials> (accessed 01.05.2023)
10. Agisoft Metashape Tutorial [Electronic resource]. — URL: <https://www.agisoft.com/support/tutorials/> (accessed 01.05.2023)
11. AliceVision Meshroom Tutorial [Electronic resource]. — URL: <https://meshroom-manual.readthedocs.io/en/latest/tutorials/sketchfab/sketchfab.html> (accessed 01.05.2023)
12. 3DF Zephyr Tutorial [Electronic resource]. — URL: <https://www.3dflow.net/technology/documents/3df-zephyr-tutorials/> (accessed 01.05.2023)
13. Colmap Tutorial [Electronic resource]. — URL: <https://colmap.github.io/tutorial.html> (accessed 01.05.2023)
14. Regard3D Tutorial [Electronic resource]. — URL: <https://www.regard3d.org/index.php/documentation/tutorial> (accessed 01.05.2023)
15. Jacob A. Morgan & Daniel J. Brogan How to VisualSFM Department of Civil & Environmental Engineering Colorado State University Fort Collins, Colorado January 2016 [Electronic resource]. — URL: https://d32ogogmya1dw8.cloudfront.net/files/getsi/teaching_materials/high-reztopo/visual_sfm_tutorial.pdf (accessed 01.05.2023)

16. iWitness Tutorial [Electronic resource]. – URL: - <https://iwitness.usc.edu/features> (accessed 01.05.2023)
17. Photomodeler Tutorial [Electronic resource]. – URL: - <https://www.photomodeler.com/tutorial-videos/>(accessed 01.05.2023)
18. Qlone Tutorial [Electronic resource]. – URL: - <https://www.qlone.pro/faq> (accessed 01.05.2023)
19. Image-Based Localization in 3D Point Clouds [Electronic resource]. – URL: - <https://mediatum.ub.tum.de/doc/1624887/1624887.pdf> (accessed 01.05.23)
20. Kosyakov A., Sweet U., etc. System engineering. Principles and practice. Translated from English. edited by V.K. Batovrin. – M.: DMK Press, 2014. – 624 p.: ill.
21. Makarov S. L. Arduino Uno and Raspberry Pi 3: from circuitry to the Internet of Things. - Moscow: DMK Press, 2018. - 204 p.
22. Camera for Raspberry Pi "Model D". — Text: electronic // Amperka: [electronic resource]. — URL: <http://wiki.amperka.ru/products:camera-raspberry-pi-model-d> (accessed: 04/15/2023).
23. Python 3 Documentation [Electronic resource]. – URL: - <https://docs.python.org/3/tutorial/> / (accessed: 01.05.2023)
24. Raspberry Pi GPIO. – Text: electronic // RPi Labs [electronic resource]. — URL: <https://rpi.science.uoit.ca/lab/gpio/> / (accessed 04.04.2023).
25. Flask: Quickstart. – Text: electronic // flask-docs [electronic resource]. — URL: <https://flask-docs.readthedocs.io/en/latest/quickstart/> / (accessed: 06.04.2023).
26. subprocess module — Working with processes. - Text: electronic // python-scripts [electronic resource]. — URL: <https://python-scripts.com/subprocess> (date of application: 04/10/2023).
27. Determination of the angle of inclination by the accelerometer. — Text: electronic // Bitaks: [electronic resource]. — URL: <http://bitaks.com/resources/inclinometer/content.html> (accessed: 03/10/2023).
28. How to use HMC5883L Compass Sensor. — Text: electronic // miliohm: [electronic resource]. — URL: <https://miliohm.com/how-to-use-hmc5883l-compass-sensor/> / (accessed: 03/10/2023).
29. Raspberry Pi High Quality Camera — Text: electronic // Amperka: [electronic resource]. — URL: <http://wiki.amperka.ru/products:raspberrypi-high-quality-camera> (accessed: 03.05.2023).
30. Gergana Antova Application of Areal Change Detection Methods Using Point Clouds Data IOP Conference Series Earth and Environmental Science 2019
31. Citation: Kharroubi, A.; Poux, F.; Ballouch, Z.; Hajji, R.; Billen, R. Three Dimensional Change Detection Using Point Clouds: A Review. *Geomatics 2022*, *2*, 457–485. <https://doi.org/10.3390/geomatics2040025>
32. K. Kawano, S. Koide and T. Kutsuna, "Learning Wasserstein Isometric Embedding for Point Clouds," 2020 International Conference on 3D Vision (3DV), Fukuoka, Japan, 2020, pp. 473-482, doi: 10.1109/3DV50981.2020.00057.