# АЛГОРИТМИЧЕСКИЙ ПОДХОД К ОБРАБОТКЕ ОБЛАКА ТОЧЕК ПРИ ПОСТРОЕНИИ ТРЁХМЕРНЫХ ИНФОРМАЦИОННЫХ МОДЕЛЕЙ ИНЖЕНЕРНЫХ ОБЪЕКТОВ

**Гунин И.И.**, магистрант,

**Шеметов И.А.**, аспирант,

ФГБОУ ВО «МГУ им. Н.П. Огарёва», г. Саранск, Россия

**Аннотация.** В статье рассматривается алгоритмическая обработка облака точек при построении трёхмерных информационных моделей инженерных объектов по данным пространственного сканирования. Предлагаемая последовательность включает предварительную фильтрацию, разбиение облака на фрагменты, анализ локальных геометрических характеристик, построение полигональной поверхности и оптимизацию модели. Подход рассчитан на большие массивы с шумами, выбросами и неоднородной плотностью точек и применим к данным наземного и воздушного лазерного сканирования. Практической основой служат программная реализация и результаты трёхмерной реконструкции объекта транспортной инфраструктуры.

**Ключевые слова:** облако точек, лазерное сканирование, трёхмерная модель, информационная модель, фильтрация данных, полигональная поверхность, алгоритм обработки, инженерные объекты

Развитие технологий трёхмерного моделирования и цифровизации инженерных объектов способствует широкому применению методов лазерного сканирования для получения пространственной информации об объектах различного назначения. Особую актуальность данные методы приобретают при обследовании зданий, мостов, путепроводов и других инженерных сооружений, для которых необходимо получать точные геометрические характеристики и контролировать техническое состояние.

Пространственные данные по инженерным сооружениям чаще всего получают лазерным сканированием LiDAR (LightDetectionandRanging). Съёмка даёт облако точек: каждая точка фиксирует положение участка поверхности в трёхмерном пространстве. Состав записи зависит от прибора и

условий съёмки. Помимо координат X, Y, Z в ней могут храниться интенсивность отражённого импульса и цвет.

Такое облако ещё нельзя сразу принимать за трёхмерную информационную модель. В измерениях встречаются шумы, изолированные точки и геометрические выбросы, плотность распределения по объекту обычно неодинакова [1]. На сложном сооружении массив получается большим, и обработка требует заметных вычислительных ресурсов. Если эти особенности не учесть, восстановленная поверхность расходится с реальной геометрией: появляются искажения и дефекты.

Учитывая перечисленные аспекты, актуальной является следующая практическая задача:для построения информационной модели объекта необходимо составить алгоритм, который проводит облако точек через последовательную обработку, выделит геометрические элементы и собиретиз них структурированное описание инженерного объекта.

В работе рассматривается такой алгоритм для облаков точек, полученных LiDAR-сканированием, применительно к построению трёхмерных информационных моделей инженерных объектов.

В статьерассматриваютсяследующие последовательные этапы: предварительная подготовка данных, фильтрация шумов и выбросов, снижение плотности облака, сегментация и восстановление геометрии отдельных элементов. Отдельно прослеживается, как эти шаги связаны между собой и как качество промежуточного результата сказывается на итоговой информационной модели.

## 1. Общая схема алгоритмической обработки облака точек

Данные лазерного сканирования на входе ещё не являются моделью объекта. Это неупорядоченный набор измерений: у каждой точки есть координаты X, Y, Z, а часто ещё интенсивность отражённого импульса и цвет. Пока массив не приведён к согласованному виду, форму сооружения по

нему восстановить трудно. В измерениях остаются шумы, сканы с разных стоянок могут лежать в разных системах координат, плотность точек от участка к участку меняется.

Поэтому обработку ведут цепочкой: следующий расчёт опирается на то, что получилось на предыдущем шаге. Сначала облако загружают и проверяют. Записи с некорректными координатами отбрасывают. Если объект снимали с нескольких положений, отдельные сканы совмещают и переводят в одну систему координат. Иначе сопоставлять точки между собой геометрически некорректно.

Дальше выполняется очистка облака точек. Изолированные точки и выбросы, которые появляются из-за погрешности измерения или сложного отражения сигнала, убирают до построения поверхности. Если их оставить в массиве, то они в дальнейшем дадут ложные фрагменты геометрии. Когда точек слишком много для следующих расчётов, плотность требуется снизить так, чтобы не потерять изломы и кромки конструкции.

Очищенный массив делят на области, соответствующие отдельным элементам сооружения. Для этого используют кластеризацию и выделение геометрических примитивов: по локальному расположению точек отделяют плоские и линейные участки от поверхности более сложной формы.

Геометрию каждой области восстанавливают по-своему. Участки правильной формы аппроксимируют примитивами. То, что примитивом не описывается, закрывают полигональной поверхностью. Способ зависит от формы элемента и от того, насколько плотно он представлен в облаке.

В конце восстановленные элементы собирают в информационную модель. Кроме геометрии в ней фиксируют то, без чего набор поверхностей ещё не описывает объект: тип элемента, размеры и взаимное положение частей.

Порядок проведения всех операций построения информационной модели показан на рисунке 1.




Рисунок 1 – Общая схема алгоритмической обработки облака точек при построении трёхмерной информационной модели инженерного объекта

Следует отметить, что качество выполнения ранних шагов оказывает сильное влияние на дальнейшие расчеты. Шумовые точки, оставшиеся в облаке, смещают границы элементов при сегментации, и в один кластер попадают чужие измерения. Если точек изначально мало, при восстановлении поверхности остаются разрывы. Подготовка и фильтрация облака точек задают пригодность итоговой модели, а не только уменьшают объём данных перед следующими расчётами.

## 2. Предварительная обработка и фильтрация облака точек

Прежде чем восстанавливать геометрию, облако нужно подготовить. На этом этапе убирают некорректные измерения, сокращают избыточное число точек и при этом оставляют те особенности формы, по которым потом узнаётся конструкция[2].

Облако точек, полученное LiDAR-съёмкой, почти всегда содержит шумы, изолированные точки и геометрические выбросы. Они появляются из-за погрешности прибора, характера отражения импульса и условий съёмки.

Плотность точек на разных частях сооружения тоже различается, иногда сильно. По такому массиву плохо считаются локальные характеристики поверхности, и ошибки переходят дальше - в сегментацию и в восстановление геометрии.

Подготовка складывается из четырёх операций: приведение пространственных данных к рабочему виду, удаление выбросов, снижение плотности и оценка локальной поверхности.

### 2.1. Подготовка исходного облака точек

Сначала облако загружают и проверяют, что в записях есть координаты X, Y и Z. Интенсивность отражённого сигнала и цвет подключают, если они есть в файле съёмки и нужны для дальнейших расчётов.

Крупное сооружение обычно снимают с нескольких стоянок, и каждый скан приходит своим облаком. Эти облака точек надо собрать в одну систему координат. Для этого выполняют регистрацию: определяют, как сканы расположены друг относительно друга. На практике часто используют ICP (IterativeClosestPoint). Метод итеративно уточняет преобразование между облаками, сопоставляя ближайшие точки.

Если всё облако уже лежит в одной системе координат, регистрацию не проводят.

Здесь же можно ограничить область интереса и выбросить точки, которые к объекту не относятся, в том числе выполнив классификацию точек окружающего рельефа и растительности [3]. Массив становится меньше, и следующие расчёты идут легче.

### 2.2. Удаление шумов и геометрических выбросов

После подготовки облако точек фильтруют. Ищут и удаляют точки, положение которых не согласуется с окружающей поверхностью.

Для этого подходят статистическая и радиусная фильтрации [1]. Статистическая фильтрация (StatisticalOutlierRemoval, SOR) смотрит на расстояния от точки до её ближайших соседей (рис. 2).

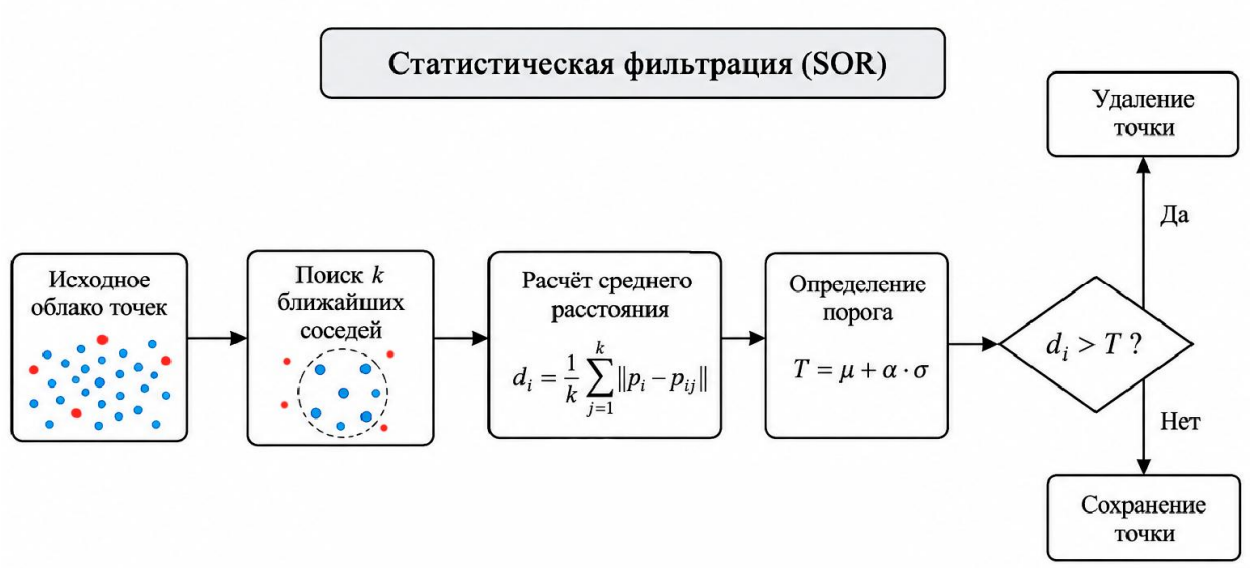


Рисунок 2 – Статистическая фильтрация

Для каждой точки берут k ближайших соседей и считают среднее расстояние до них:

$$d_i = \frac{1}{k}\sum_{j=1}^{k}\|p_i - p_{i,j}\|$$

где $d_i$- среднее расстояние до ближайших соседей рассматриваемой точки; $k$- количество анализируемых соседей; $p_i$- координаты рассматриваемой точки; $p_{i,j}$- координаты j-й соседней точки.

По всем точкам затем находят общее среднее этих расстояний$\mu$ и среднеквадратичное отклонение$\sigma$. Порог отсечения задают так:

$$T = \mu + \alpha\sigma$$

где $\alpha$- коэффициент, определяющий допустимое отклонение.

Точку относят к выбросам и удаляют, если её среднее расстояние до соседей выше этого порога.

SOR удобен тем, что изолированные точки находятся автоматически, по статистике взаимных расстояний, без ручной разметки. На облаке с сильно неравномерной плотностью классификация начинает ошибаться, поэтому k иαподбирают по конкретному объекту, а не берут «типовыми».

Другой вариант – радиусная фильтрация (RadiusOutlierRemoval, ROR, которая считает, сколько соседей попало в заданную окрестность точки (рис. 3).

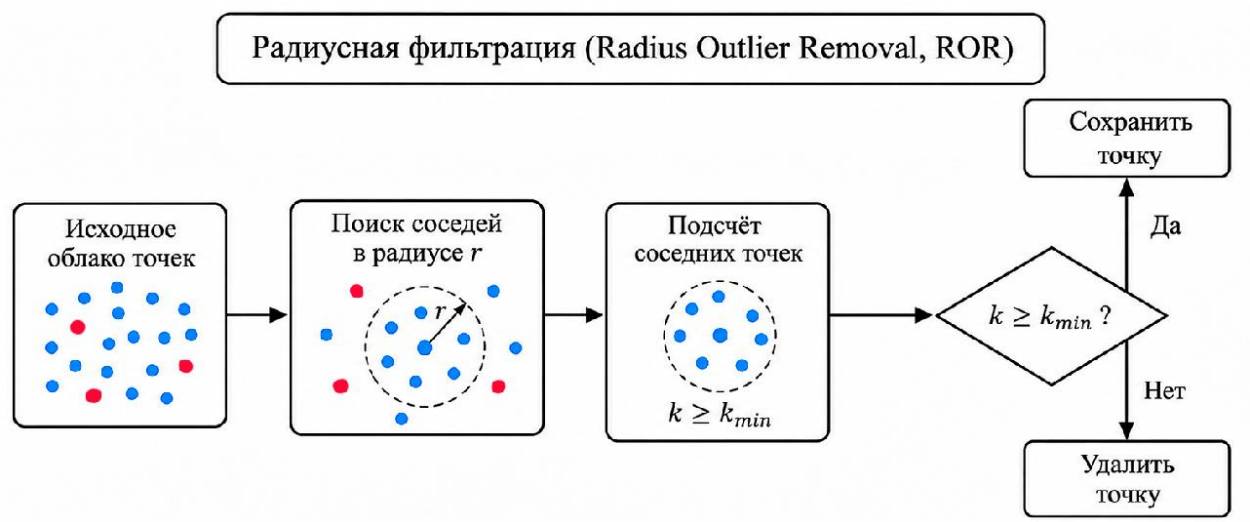


Рисунок 3 – Радиусная фильтрация

Вокруг точки строят сферу радиуса r и считают число точек внутри неё. Если соседей меньше заданного минимума, точку считают изолированной и исключают из облака.

Параметров два: радиус поиска и минимальное число соседей. Их назначают по плотности облака и по той точности, которая нужна дальше.

Оба фильтра уменьшают долю некорректных измерений, но решают разные задачи. SOR опирается на распределение расстояний между соседями. ROR - на число точек в окрестности заданного радиуса. На объекте с ровной плотностью часто хватает одного из них; при резких перепадах плотности их имеет смысл применять последовательно и сверять, какие точки отсеял каждый.

### 2.3. Уменьшение плотности облака точек

После очистки выбросов плотность можно снизить. Так сокращают объём данных и время последующих расчётов. Для крупного инженерного объекта это обычная ситуация: облако точек после съёмки содержит миллионы точек, и держать их все до сегментации незачем.

Здесь используют VoxelGrid. Пространство разбивают на элементарные объёмные ячейки –воксели.

Воксель– это прямоугольный параллелепипед, отвечающий своему участку пространства. Точки, попавшие в одну ячейку, собирают в группу. Вместо каждой исходной точки оставляют одну представительную: её координаты равны среднему арифметическому координат точек группы.

Близко лежащие измерения заменяются меньшим набором таких точек. Форма поверхности при этом сохраняется приближённо, в пределах размера ячейки.

Главный параметр – размер вокселя. Чем крупнее ячейка, тем меньше точек остаётся в облаке. Если ячейку взять слишком большой, пропадают мелкие детали: кромки, выступы, узкие элементы. Размер поэтому назначают по требуемой точности модели и по характерным размерам элементов объекта, а не по желанию максимально сжать файл.

VoxelGrid уменьшает объём пространственных данных и оставляет основные черты геометрии, если размер ячейки согласован с масштабом сооружения.

### 2.4. Подготовка облака к сегментации

Когда фильтрация и прореживание выполнены, облако готовят к выделению отдельных геометрических элементов. Ключевая операция здесь - оценка нормалей.

Нормаль – вектор, который задаёт направление локальной поверхности в окрестности точки. Её получают по ближайшим соседям: по ним строят

приближённую плоскость, а перпендикуляр к этой плоскости принимают за нормаль.

С нормалями точки сравнивают уже не только по расстоянию. Учитывается и то, куда ориентирован участок поверхности. Для инженерных объектов это существенно: в одном облаке соседствуют плоские, цилиндрические и иные элементы, и близость координат сама по себе их не разделяет.

Крупный объект дополнительно режут на пространственные фрагменты. Каждый фрагмент считают отдельно. В оперативной памяти тогда находится не всё облако, а его часть, и требования к памяти снижаются.

На выходе предварительной обработки - очищенное облако с меньшим числом шумовых и избыточных точек и с оценёнными нормалями. Его передают на сегментацию, где из массива выделяют отдельные геометрические элементы сооружения.

## 3. Сегментация и построение трёхмерной информационной модели

После предварительной обработки облако ещё остаётся единым массивом. Сегментация делит его на группы точек, каждая из которых отвечает своему элементу сооружения.

Для разбиения в работе используются три алгоритма, и каждый смотрит на облако точек по-своему. EuclideanClustering собирает в один кластер точки, близкие друг к другу в пространстве: граница кластера проходит там, где между точками появляется разрыв больше заданного. RegionGrowing идёт от затравочной точки и присоединяет соседей с близким направлением нормали, поэтому в одну область попадают участки со сходной ориентацией поверхности, даже если элемент протяжённый. RANSAC ищет в облаке геометрические примитивы, в первую очередь плоскости и цилиндры, и отделяет точки, которые на такой примитив ложатся.

На строительных конструкциях этого набора достаточно, чтобы вынуть из общего облака стены, колонны, балки и перекрытия [4]. Дальше геометрию восстанавливают уже по группам, а не по всему массиву сразу.

По выделенным точкам строят трёхмерную полигональную модель. Поверхность в ней задаётся вершинами, рёбрами и гранями. В реконструкции грань почти всегда берут треугольной: треугольник однозначно задаёт плоскость и без дополнительных условий обрабатывается в графических системах. Вопросы алгоритмического построения трёхмерных моделей реальных объектов по данным пространственных измерений рассматриваются, в частности, в работе [5].

Перед построением сетки плотное облако ещё раз просматривают и убирают изолированные точки, выбросы и участки с низкой надёжностью измерений. Ошибка, оставшаяся в плотном облаке, на модели проявляется уже как лишняя поверхность, отверстие или местное искажение, и править её на сетке труднее, чем отсечь точку заранее.

Для точек плотного облака при необходимости заново оценивают нормали. Нормаль показывает, куда ориентирована локальная поверхность рядом с точкой, и по ней алгоритм понимает, как вести поверхность через облако и какая её сторона внешняя.

Когда треугольная грань уже построена, единичный вектор нормалипсчитают по координатам её вершин:

$$n = \frac{(v_2 - v_1) \times (v_3 - v_1)}{\|(v_2 - v_1) \times (v_3 - v_1)\|},$$

где $v_1, v_2, v_3$- координаты вершин грани. Векторное произведение задаёт направление, деление на норму приводит его к единичной длине.

Саму поверхность получают одним из двух способов. При *триангуляции* точки плотного облака напрямую соединяют в систему треугольников, и вершина сетки совпадает с измерением. *Метод*

*Пуассона*устроен иначе: по точкам и нормалям сначала восстанавливают непрерывную поверхность, а уже её переводят в полигональную сетку. Первый путь сохраняет измеренные точки, второй лучше закрывает поверхность там, где облако неровное по плотности, но сильнее зависит от того, насколько верно оценены нормали.

Качество сетки повторяет качество плотного облака. Если точки покрывают объект без пропусков и без заметного шума, поверхность выходит гладкой и непрерывной. Пропуски и шумы дают отверстия, лишние фрагменты и неровности. На объектах транспортной инфраструктуры пропуски возникают там, куда луч не доходит или откуда импульс почти не возвращается: за балками и ограждениями, на нижней грани пролётного строения, на мокром и тёмном покрытии.

Готовую сетку очищают. Удаляют мелкие изолированные треугольники, не лежащие на поверхности объекта. Отверстия, которые остались из-за нехватки точек в плотном облаке, при необходимости закрывают.

Если вершин и граней слишком много для хранения, передачи и просмотра, сетку упрощают. Упрощение не должно съедать то, ради чего модель строится: границы элементов, кромки, трещины и другие особенности конструкции оставляют на том уровне детализации, который нужен для информационной модели.

## 4. Практические результаты

Предложенный подход реализован программно и проверен на данных лазерного сканирования многопролётного моста (центральный пролёт около 46 м, боковые - по 27 м). Проезжая часть и опоры выполнены из бетона, продольные балки - из стали. Съёмка выполнялась наземным и воздушным лазерным сканированием.

Исходное объединённое облако после регистрации содержало более 320 млн точек и характеризовалось шумом, изолированными выбросами и неоднородной плотностью. После статистической и радиусной фильтрации, а также прореживания методом Voxel Grid облако было разбито на 27 пространственных фрагментов. Суммарно в них сохранилось 308 066 811 точек. Распределение точек по фрагментам приведено в таблице 1.

Таблица 1 – Число точек по подмоделям после фильтрации

| **№** | **Число точек** |
|---|---|
| 0 | 41 693 |
| 1 | 1 051 |
| 2 | 16 865 561 |
| 3 | 5 489 161 |
| 4 | 10 264 725 |
| 5 | 18 329 086 |
| 6 | 5 289 068 |
| 7 | 123 705 |
| 8 | 27 466 012 |
| 9 | 24 616 515 |
| 10 | 1 490 161 |
| 11 | 36 080 152 |
| 12 | 15 301 900 |
| 13 | 63 004 534 |
| 14 | 87 117 |
| 15 | 160 398 |
| 16 | 845 |
| 17 | 40 125 122 |
| 18 | 33 684 821 |
| 19 | 1 179 467 |
| 20 | 3 090 311 |
| 21 | 355 636 |
| 22 | 323 029 |
| 23 | 1 264 247 |
| 24 | 931 821 |
| 25 | 1 724 028 |
| 26 | 776 645 |

Наиболее плотные фрагменты (№ 11, 13, 17, 18) соответствуют участкам с хорошим покрытием и высокой плотностью исходных измерений. Фрагменты с малым числом точек относятся к зонам сильного затенения или краям области интереса.

По фрагментам построена полигональная поверхность методом Пуассона с последующей очисткой и упрощением. Суммарно модель содержит 45 463 328 вершин и 90 897 655 граней. Результаты по отдельным подмоделям приведены в таблице 2.

Таблица 2 – Результаты построения полигональной модели по подмоделям

| № | Число вершин | Число граней |
|---|---|---|
| 0 | 10 752 | 21 436 |
| 1 | 0 | 0 |
| 2 | 3 772 269 | 7 542 792 |
| 3 | 968 756 | 1 936 238 |
| 4 | 1 726 843 | 3 452 635 |
| 5 | 2 538 335 | 5 075 307 |
| 6 | 1 039 512 | 2 078 138 |
| 7 | 35 873 | 71 661 |
| 8 | 4 407 075 | 8 812 084 |
| 9 | 3 906 589 | 7 811 997 |
| 10 | 293 466 | 586 719 |
| 11 | 4 386 909 | 8 772 637 |
| 12 | 2 359 220 | 4 718 193 |
| 13 | 9 198 227 | 18 385 881 |
| 14 | 24 755 | 49 444 |
| 15 | 54 664 | 109 269 |
| 16 | 0 | 0 |
| 17 | 3 906 587 | 7 810 369 |
| 18 | 4 488 351 | 8 975 435 |
| 19 | 297 923 | 595 237 |
| 20 | 730 851 | 1 461 020 |
| 21 | 105 644 | 211 069 |
| 22 | 30 817 | 61 442 |
| 23 | 345 514 | 690 867 |
| 24 | 287 176 | 574 228 |
| 25 | 318 315 | 635 853 |
| 26 | 228 905 | 457 704 |

Наиболее детализированные сетки получены для подмоделей № 13, 18, 8, 11, 9 и 17, что связано с высокой плотностью исходного облака на соответствующих участках. Для подмоделей № 1 и № 16 поверхность построить не удалось: после фильтрации в этих фрагментах оставалось недостаточно точек для формирования устойчивой сетки.

Визуализация построенной поверхности подмодели № 5 приведена на рисунке 4.

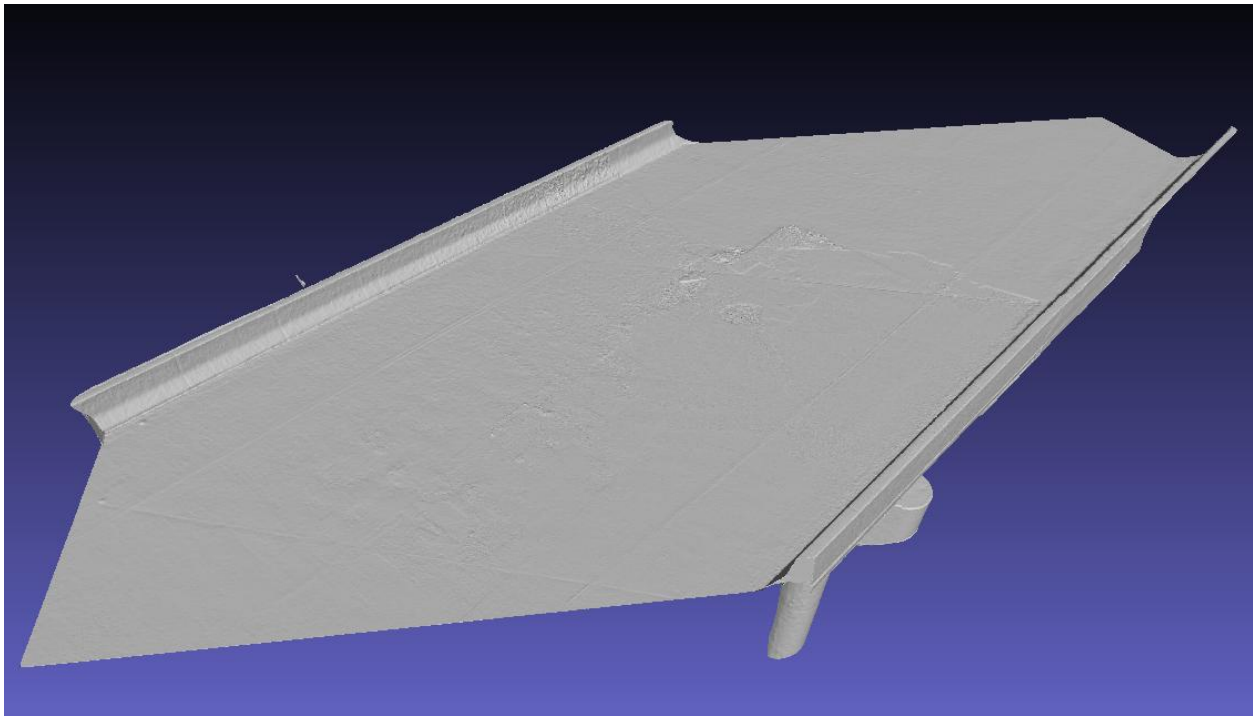

Рисунок 4 – Визуализация поверхности подмодели № 5

По рисунку видно, что построенная поверхность является связной и корректно отражает форму конструктивных элементов моста.

Для оценки качества восстановленной геометрии дополнительно рассматривались увеличенные фрагменты поверхности, демонстрирующие детализацию боковых ограждений и опорных элементов моста.На рисунке 5 видно, что форма барьерного ограждения и опорных элементов восстановлена корректно и достаточно детализировано. При этом на отдельных участках поверхности наблюдаются мелкие неровности и

шероховатости. Это связано с тем, что данные зоны имели пониженную плотность точек из-за затенения или неблагоприятных условий отражения лазерного импульса.

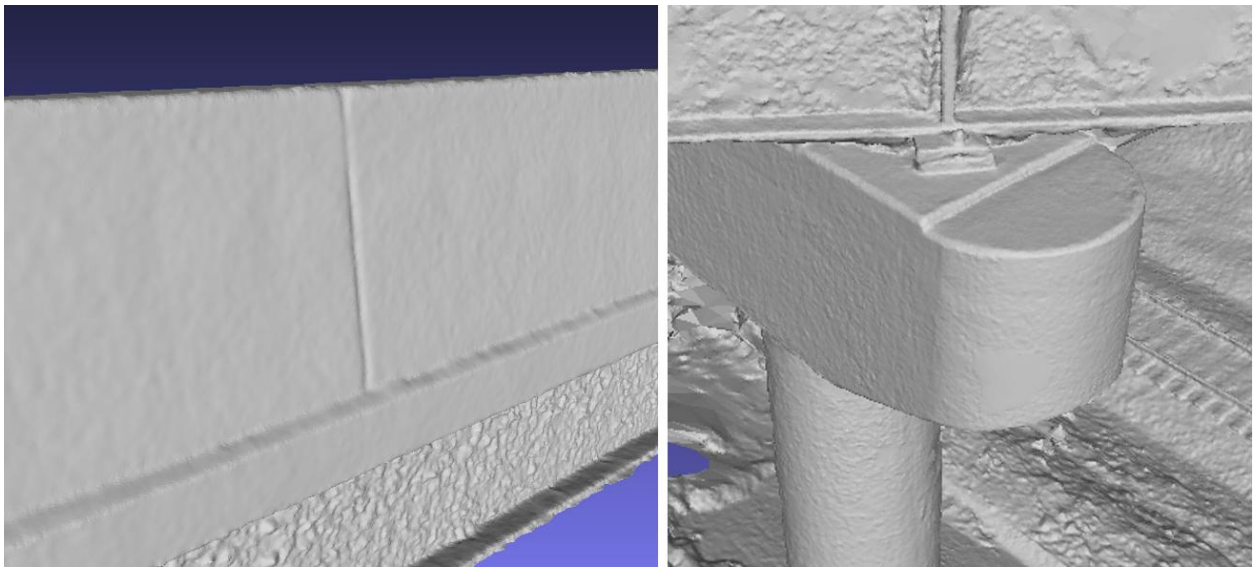

Рисунок 5 – Фрагменты поверхности, демонстрирующие детализацию боковых ограждений и опорного элемента моста

Геометрическая точность оценивалась по отклонениям между точками облака и восстановленной поверхностью. Для основных несущих элементов отклонения находились в пределах точности исходного лазерного сканирования. На участках с низкой плотностью точек локальные отклонения были выше.

Суммарное время выполнения всех этапов обработки составило около 11 часов. Проверка выполнялась на ноутбуке с процессором Intel Core i7-9750H (12 логических потоков), 16 ГБ оперативной памяти, графическим процессором NVIDIA GeForce GTX 1650 с 4 ГБ видеопамяти и SSD-накопителем с 200 ГБ свободного пространства. Полученное время показывает, что предложенный подход позволяет выполнять полный цикл обработки крупного облака точек на персональном компьютере потребительского класса, без использования специализированной

вычислительной станции. При этом качество полученной модели оказалось достаточным для визуального анализа объекта транспортной инфраструктуры. Разбиение на фрагменты позволило выполнять ресурсоёмкие этапы без превышения объёма оперативной памяти.

## Заключение

В работе рассмотрен алгоритмический подход к обработке облаков точек лазерного сканирования для построения трёхмерных информационных моделей инженерных объектов. Последовательность включает фильтрацию выбросов, снижение плотности, оценку нормалей, сегментацию и построение полигональной поверхности.

Проверка на данных сканирования моста подтвердила работоспособность подхода. Облако объёмом более 300 млн точек после фильтрации и разбиения на 27 фрагментов успешно обработано; итоговая модель содержит около 45 млн вершин и 91 млн граней.

Предложенный алгоритм применим к данным наземного и воздушного лазерного сканирования крупных инженерных сооружений и может использоваться при построении информационных моделей для обследования и мониторинга объектов транспортной инфраструктуры.

## Литература

1. Wang Q., Tan Y., Mei Z. Computational Methods of Acquisition and Processing of 3D Point Cloud Data for Construction Applications // Archives of Computational Methods in Engineering. 2020. — Vol. 27, № 2. — P. 479–499.

2 Петросян А.В., Бурков П.В. Препроцессинг облаков точек в задачах анализа результатов лазерного сканирования объектов капитального строительства // Русский инженер. Международный конгресс. Т. 3.

Комплексная автоматизация проектирования и производства (КАПП-2025). М.: МГТУ им. Н.Э. Баумана, 2026. С. 107–114.

3. Данилова Л.О., Рашевский Н.М., Рекунов С.С., Трудов Я.А., Гуртяков А.С. Классификация плотного облака точек при моделировании рельефа // Academia. Архитектура и строительство. 2024. № 3. С. 76–81. URL: https://www.aac.raasn.ru/index.php/aac/article/view/561 (дата обращения: 20.10.2026). DOI: 10.22337/2077-9038-2024-3-76-81.

4 Xu Y., Tong X., Stilla U. Toward Building and Civil Infrastructure Reconstruction from Point Clouds: A Review on Data and Key Techniques // IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing. 2021. Vol. 14. P. 2857–2885. DOI: 10.1109/JSTARS.2021.3060568.

5. Михайлова С.С. Исследование и разработка алгоритмов и методов построения трехмерных компьютерных моделей реальных объектов // ComputationalNanotechnology. 2024. Т. 11. № 1. С. 13–24. URL: https://journals.rcsi.science/2313–223X/article/view/256192 (дата обращения: 22.10.2026). DOI: 10.33693/2313-223X-2024-11-1-13-24. EDN DDIBVK