

**Отчет по практическим заданиям
1b и 2b**

Филак Александр

Долгопрудный
2025

1 Введение

В рамках двух практических заданий были изучены и реализованы методы обработки облаков точек с использованием Python и библиотеки NumPy. Задание 1b было посвящено методам субдискретизации (downsampling) облаков точек, а задание 2b — методам пространственной фильтрации и выделения областей по геометрическим признакам. Оба задания направлены на освоение базовых операций предобработки точечных данных, которые являются критически важными для последующего анализа в компьютерном зрении, робототехнике и 3D-моделировании.

2 Задание 1b: Субдискретизация облаков точек

Целью данного задания было освоение различных методов уменьшения плотности облаков точек при сохранении их ключевых геометрических характеристик.

2.1. Методы реализации

Были реализованы три метода субдискретизации:

1. **Случайная выборка (Random subsampling)** — равномерный случайный выбор N точек из исходного облака. Наиболее быстрый, но может терять важные детали.
2. **Воксельная сетка (Voxel grid subsampling)** — пространство делится на равные ячейки (воксели), в каждой из которых остаётся только одна точка. Эффективно уменьшает плотность в равномерно заполненных областях.
3. **Выбор наиболее удалённых точек (Farthest Point Sampling)** — итеративный выбор точек, максимально удалённых от уже выбранных. Наиболее эффективно сохраняет геометрию, но требует больше вычислений.

2.2. Визуализация результатов

2.3. Анализ результатов

1. **Влияние размера выборки на визуальное качество:** При уменьшении выборки до 50% случайнм методом заметна потеря мелких деталей и появление "дыр" в равномерных областях.
2. **Влияние размера вокселя:** При размере вокселя 0.001 сохраняется больше деталей по сравнению с большими размерами. Мелкий воксель лучше сохраняет геометрию, но даёт меньшую степень сжатия.

Original Point Cloud

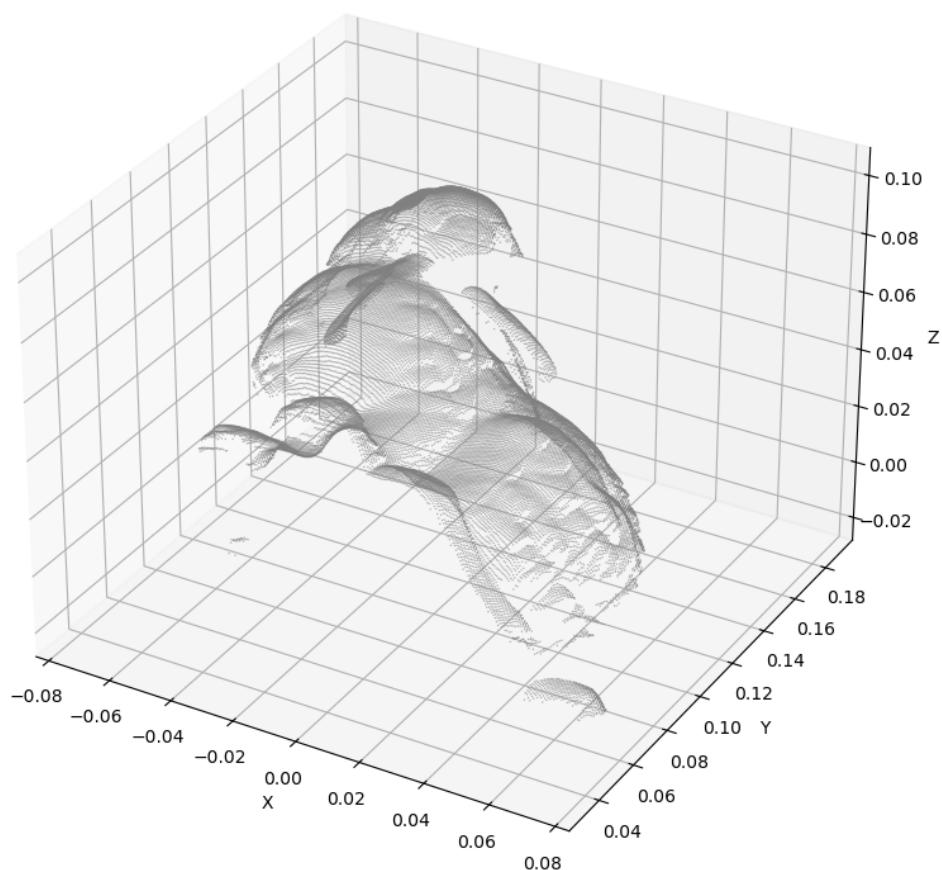


Рис. 1: Исходное облако точек

Random subsampling (50%)

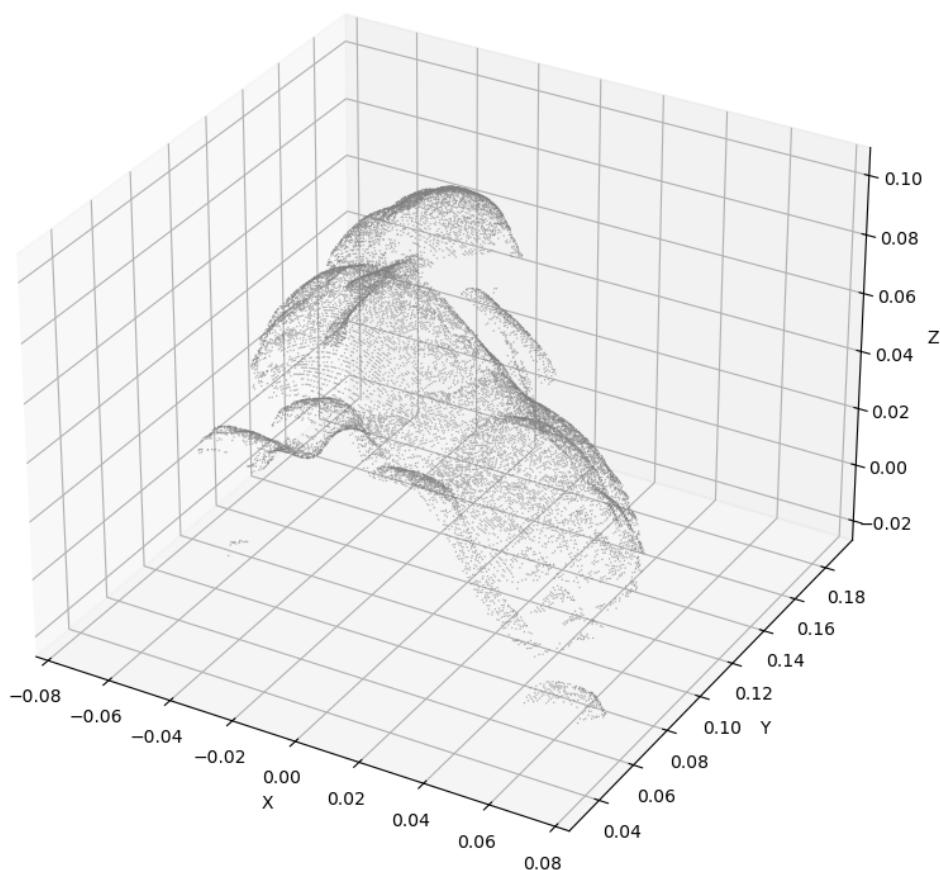


Рис. 2: Случайная субдискретизация (50%)

Voxel subsampling (voxel size 0.001)

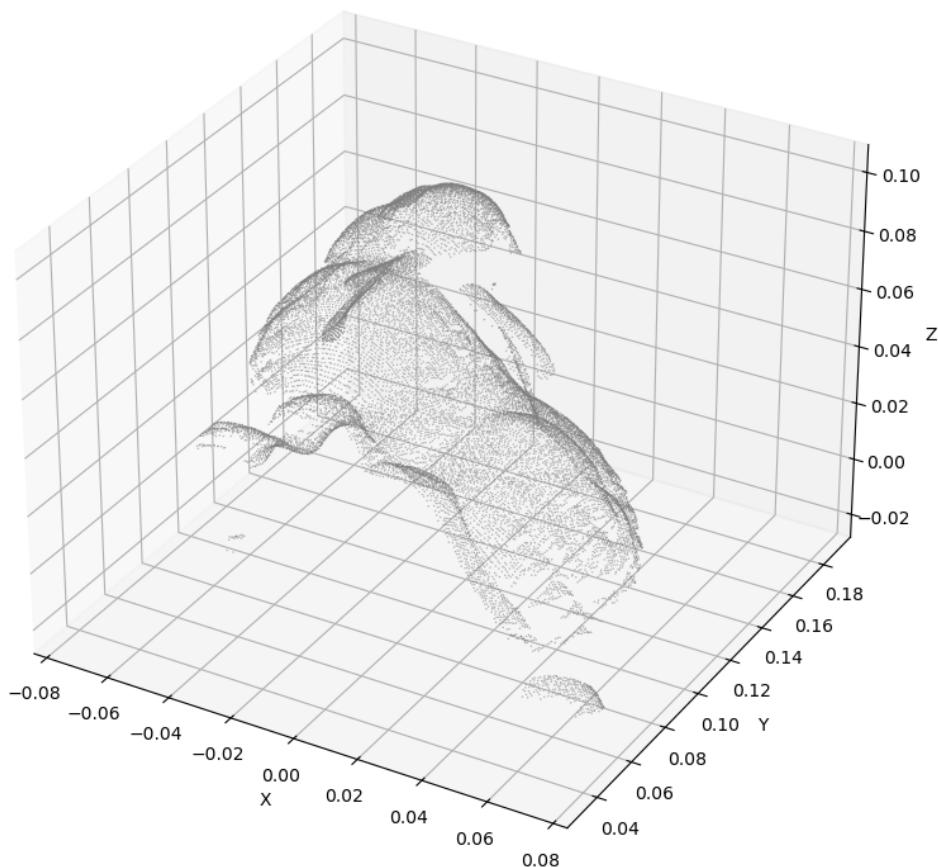


Рис. 3: Воксельная субдискретизация (размер voxеля 0.001)

FPS subsampling

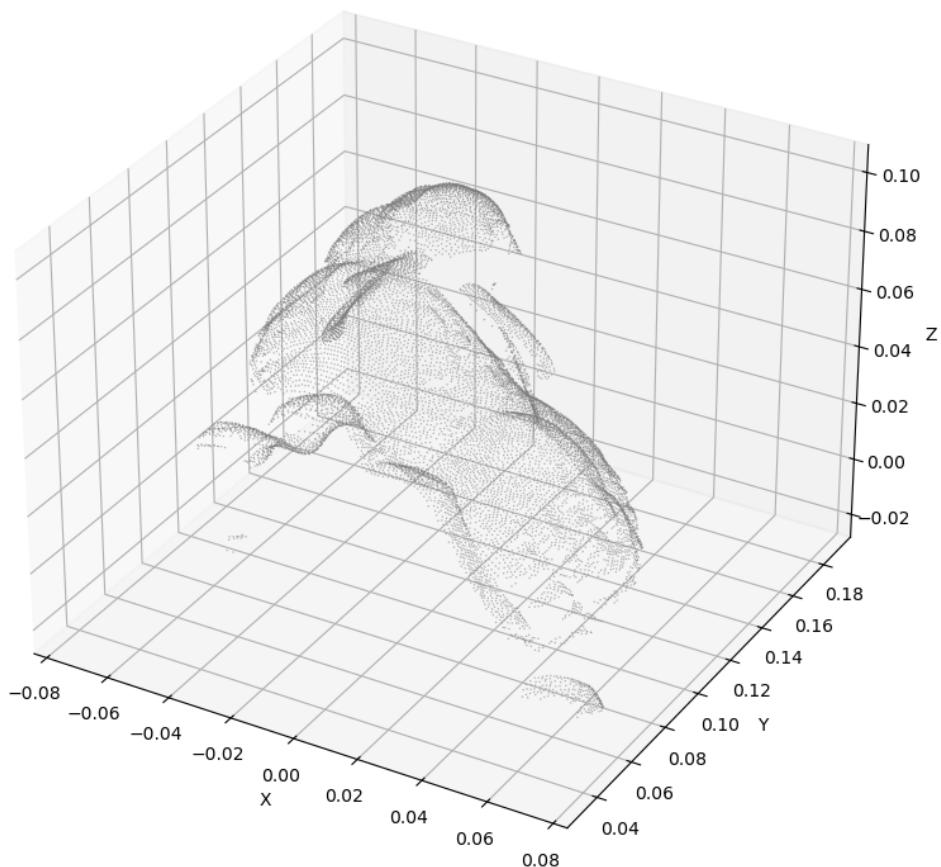


Рис. 4: Выбор наиболее удалённых точек (FPS)

3. **Эффективность по времени:** Случайная выборка — самый быстрый метод (~ 0.001 с), воксельная сетка (~ 1.0 с) и FPS (~ 1.0 с) — существенно более медленные.
4. **Сохранение геометрии:** FPS лучше всего сохраняет геометрические особенности объекта, воксельная сетка хорошо сохраняет равномерное покрытие, случайная выборка может терять важные детали.

3 Задание 2b: Выделение областей в облаках точек

Целью данного задания было освоение методов пространственной фильтрации точек по различным геометрическим критериям.

3.1. Методы реализации

Были реализованы три метода фильтрации:

1. **Ограничивающий параллелепипед (Bounding Box)** — выделение точек, попадающих в заданный диапазон координат.
2. **Фильтр по высоте** — выделение точек выше определённого уровня ($Z >$ порогового значения).
3. **Фильтр по радиусу** — выделение точек в пределах заданного радиуса от центральной точки.

3.2. Визуализация результатов

4 Выводы

1. Для быстрого уменьшения объёма данных без сохранения точной геометрии достаточно случайной выборки.
2. Для сохранения геометрических особенностей при уменьшении плотности оптimalен метод FPS, несмотря на его вычислительную сложность.
3. Воксельная сетка представляет собой хороший компромисс между скоростью и сохранением структуры.
4. Пространственная фильтрация позволяет эффективно выделять значимые области для последующего анализа.
5. Сочетание методов субдискретизации и фильтрации позволяет создавать гибкие конвейеры предобработки 3D-данных для различных прикладных задач.

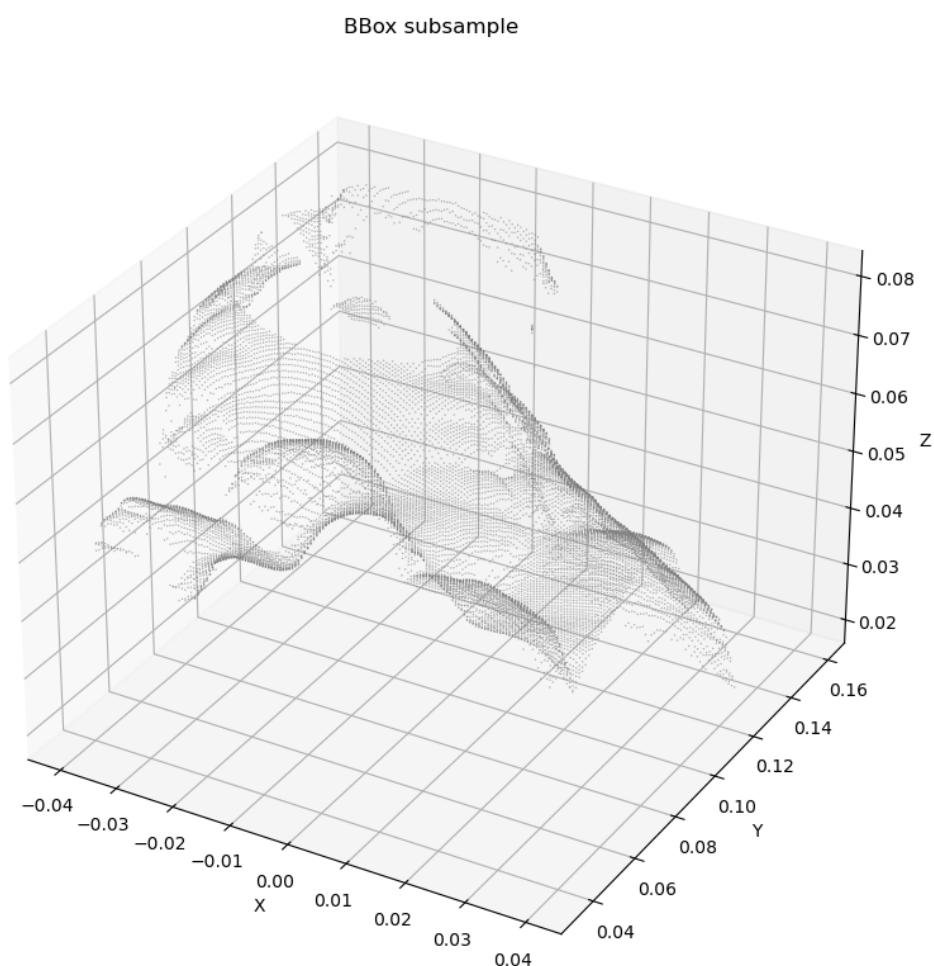


Рис. 5: Выделение области ограничивающим параллелепипедом

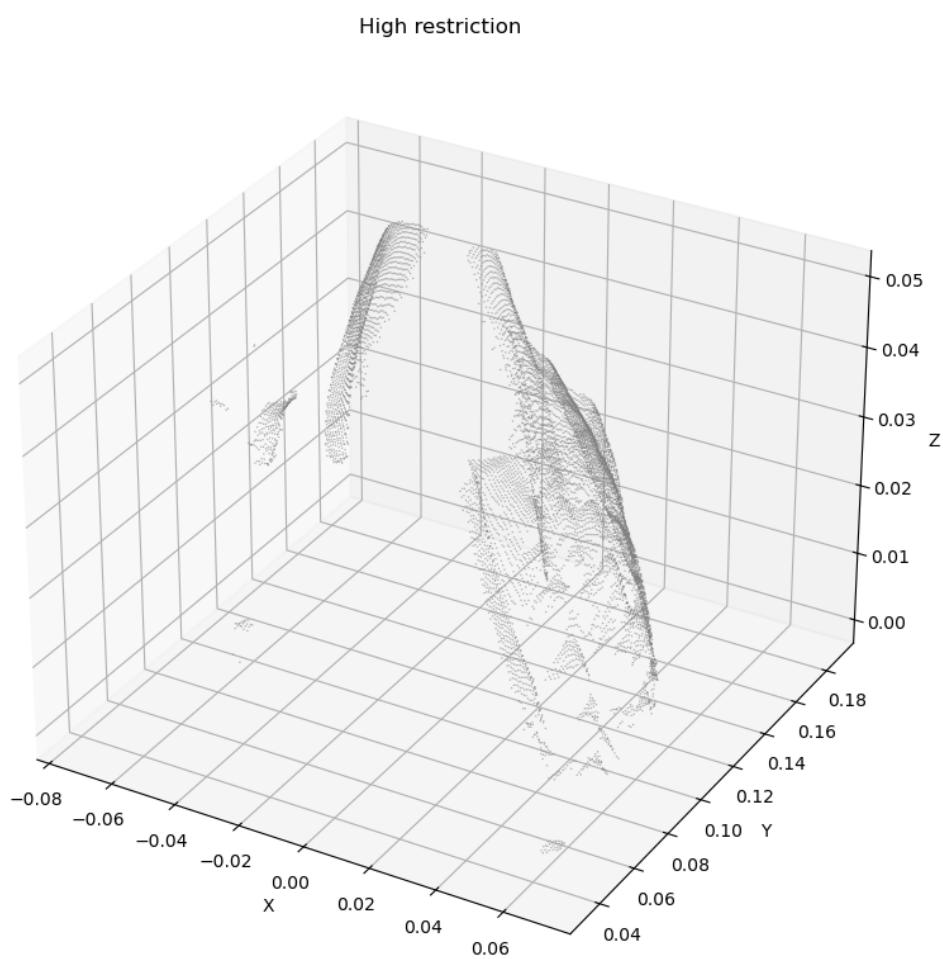


Рис. 6: Фильтрация по высоте (верхние точки)

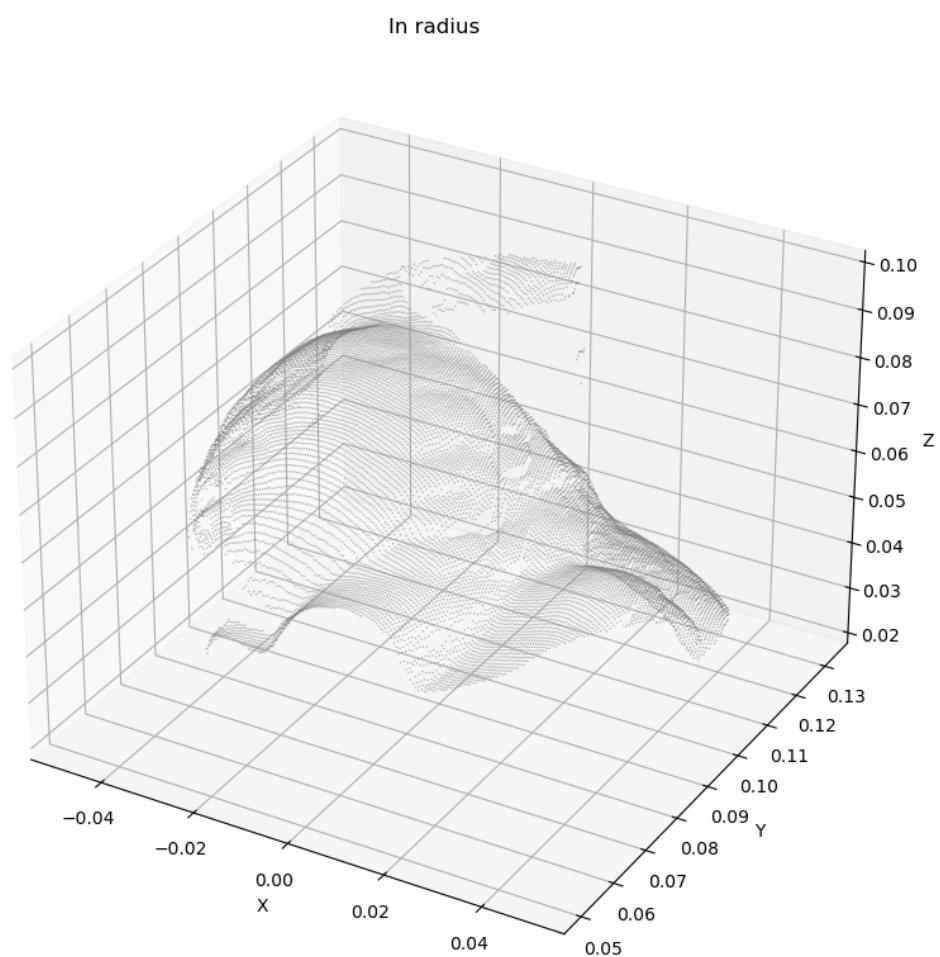


Рис. 7: Выделение точек в заданном радиусе

5 Заключение

В ходе выполнения заданий были успешно освоены базовые методы обработки облаков точек с использованием Python и NumPy. Полученные навыки субдискретизации и пространственной фильтрации являются фундаментальными для работы с 3D-данными в таких областях, как компьютерное зрение, робототехника, автономные системы и 3D-моделирование. Реализованные алгоритмы могут быть использованы как самостоятельные инструменты, так и в качестве этапов предобработки в более сложных конвейерах обработки точечных данных.