

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

**Навчально-науковий фізико-технічний інститут
Кафедра математичного моделювання та аналізу даних**

«На правах рукопису»

УДК _____ 51.7 _____

До захисту допущено:

Виконувач обов'язки завідувача
кафедри

_____ Іван ТЕРЕЩЕНКО

« 16 » _____ травня 2023 р.

Магістерська дисертація

на здобуття ступеня магістра

**за освітньо-науковою програмою «Математичні методи моделювання,
розпізнавання образів та комп'ютерного зору»**

зі спеціальності 113 «Прикладна математика»

на тему: « 3D-реконструкція приміщення за сферичними панорамами »

Виконав:

студент II курсу, групи ФІ-11мн

Крохальов Іван Данилович _____

Науковий керівник:

доцент кафедри ММАД, к. т. н.

Орехов Олександр Арсенійович _____

Консультант:

професор кафедри ММАД, д. т. н.

Шелестов Андрій Юрійович _____

Рецензент: доцент, к.т.н.

старший науковий співробітник ІКД ДКА і НАН України

Силантьєв Сергій Олександрович _____

Засвідчую, що у цій магістерській дисертації
немає запозичень з праць інших авторів без
відповідних посилань.

Студент _____

Київ – 2023 року

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Навчально-науковий фізико-технічний інститут
Кафедра математичних методів аналізу даних

Рівень вищої освіти – другий (магістерський)

Спеціальність – 113 «Прикладна математика»

Освітньо-наукова програма «Математичні методи моделювання,
розпізнавання образів та комп'ютерного зору»

ЗАТВЕРДЖУЮ

в.о. завідуючого кафедри

_____ Іван ТЕРЕЩЕНКО _____

«_16_» _____ травня _____ 2023 р.

ЗАВДАННЯ
на магістерську дисертацію студенту

Крохальов Іван Данилович

1. Тема дисертації «3D-реконструкція приміщень за сферичними панорамами», науковий керівник дисертації Орехов Олександр Арсенійович, доцент, к.т.н. затверджені наказом по університету від «___» _____ 20__ р. № _____
2. Термін подання студентом дисертації _____
3. Об'єкт дослідження — панорамні фотознімки приміщень;
4. Предмет дослідження - методи 3D-реконструкції сцени та їх робота на панорамних фотознімках приміщень;
5. Перелік завдань, які потрібно розробити
 - Вивчити форми представлення 3D-об'єктів;
 - Дослідити основні програмні пакети, що використовуються для роботи із 3D-об'єктами (MeshLab, Blender, Open3D);

- Дослідити концепцію 360-градусної камери та приклади зображень, отриманих за допомогою неї;
- Провести аналіз літератури, що стосується Structure-from-Motion;
- Дослідити кожен із етапів Structure-from-Motion, візуалізувати та пояснити результати;
- Розробити програмне забезпечення, що дозволяє отримувати 3D-реконструкцію приміщення за вхідними знімками.

6. Орієнтовний перелік графічного (ілюстративного) матеріалу

7. Консультанти розділів дисертації[†]

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1	Шелестов А. Ю., проф. кафедри ММАД (НН ФТІ)		
2	Шелестов А. Ю., проф. кафедри ММАД (НН ФТІ)		
3	Шелестов А.Ю, проф. кафедри ММАД (НН ФТІ)		

9. Дата видачі завдання _____

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
1	Ознайомлення із постановкою задачі, форматом вхідних даних	1 вересня 2022р. - 30 вересня 2022р.	Виконано
2	Збір різних прикладів вхідних даних	1 жовтня 2022р. — 30 жовтня 2022р.	Виконано
3	Підготовка програмних засобів для	1 листопада — 30	Виконано

	дослідження (конфігурації середовища, встановлення необхідних бібліотек та залежностей)	листопада 2022 р.	
4	Проведення експериментів на зібраних даних, отримання перших 3D-моделей, аналіз їх якості	1 грудня 2022р. - 30 січня 2023р.	Виконано
5	Аналіз вразливостей і недоліків сформованого підходу, кроки для покращення якості 3D-реконструкції	1 лютого 2023р. - 15 березня 2023р.	Виконано
6	Оформлення магістерської дисертації	16 березня 2023р. -- 16 травня 2023р.	Виконано
7	Отримання допуску до захисту	16 травня 2023р.	Виконано
8	Захист магістерської дисертації	24 травня 2023р.	Виконано

Студент

Крохальов Іван Данилович

Науковий керівник

Олександр ОРЕХОВ

РЕФЕРАТ

Магістерська дисертація: 59 с., 35 рис., 2 табл., 15 джерел.

Об'єкт дослідження – панорамні знімки приміщень, 3D-моделі приміщень, отримані внаслідок реконструкції; усі матеріали, що є результатом розв'язання проміжних задач реконструкції;

Предмет дослідження – методи 3D-реконструкції, підхід Structure-from-Motion, його поведінка при різних прикладах вхідних даних, можливі кроки для покращення якості реконструкції;

Мета роботи — побудувати алгоритм 3D-реконструкції на основі підходу Structure-from-Motion, для використання на різних прикладах вхідних даних.

В роботі наведено програмну реалізацію математичної моделі, що дозволяє проводити дослідження у відповідності до мети роботи. Проведений порівняльний аналіз різних конфігурацій 3D-реконструкції. В роботі також наведено порівняння запропонованого метода із альтернативними підходами до 3D-реконструкції.

3D-РЕКОНСТРУКЦІЯ, СФЕРИЧНІ ПАНОРАМИ, КОМП'ЮТЕРНИЙ ЗІР, STRUCTURE-FROM-MOTION, ТРИВИМІРНА ГЕОМЕТРІЯ, ФОТОГРАММЕТРІЯ, НЕЙРОННІ МЕРЕЖІ, ГЛИБОКЕ НАВЧАННЯ, MULTI-VIEW-GEOMETRY, MULTI-VIEW STEREO.

ABSTRACT

Master's thesis: 59 p., 35 figures, 2 tables, 15 sources.

The object of research – spherical panoramas of indoor scenes, 3D-models of indoor spaces, all materials that are received from all stages of 3D-reconstruction;

Subject of research – 3D-reconstruction methods, Structure-from-Motion approach, behavior of this approach (different types of input data), possible options to improve quality of 3D-reconstruction;

The purpose of the work – to build a good model for 3D indoor scene reconstruction, based on Structure-from-motion technique and neural networks methods.

The master thesis implements a program that allows research to be carried out in accordance with the purpose of the work. A comparative analysis of the effectiveness of various model configurations has been carried out. The paper also includes comparative analysis between proposed approach and alternatives.

3D-RECONSTRUCTION, SPHERICAL PANORAMAS, COMPUTER VISION, STRUCTURE-FROM-MOTION, STEREO GEOMETRY, PHOTOGRAMMETRY, NEURAL NETWORKS, DEEP LEARNING, MULTI-VIEW GEOMETRY, MULTI-VIEW STEREO.

ЗМІСТ

Вступ.....	8
1 Аналіз існуючих підходів до задачі 3d-реконструкції приміщення.....	9
1.1 Опис задачі.....	9
1.2 Вхідні дані.....	10
1.3 Результат 3D - реконструкції.....	17
Висновки до розділу 1.....	19
2 Математичні методи 3d-реконструкції.....	21
2.1 Форми представлення 3D-об'єкта.....	21
2.2 Structure-from-Motion.....	28
2.3 Глобальний та інкрементальний підходи.....	38
Висновки до розділу 2.....	41
3 Аналіз практичних результатів.....	42
3.1 Опис зібраних даних.....	42
3.2 Noised point cloud (зашумлена точкова хмара).....	43
3.3 Dense point cloud (щільна точкова хмара).....	47
3.4 Текстурований меш.....	49
3.5 Порівняння реконструкції Пуассона та Делоне.....	52
Висновки до розділу 3.....	54
Висновки.....	55
Перелік літературних посилань.....	56

ВСТУП

Проблема 3D-реконструкції сцени представляє собою інтерес у сфері комп'ютерного зору та фотограмметрії уже протягом кількох десятиліть. Процес конвертування знімків у 3D-реконструкцію може бути складною задачею, враховуючи спотворення, викривлення, різну якість освітлення.

3D-реконструкція сцени є складною багатостадійною задачею, існує велика кількість різних можливих підходів до задачі (наприклад, група підходів що базуються на Structure-from-Motion; підходи що базуються на LiDAR-скануванні; нейромережева реконструкція). Вибір алгоритму може залежати від багатьох чинників — зокрема від якості та кількості вхідних даних, від природи вхідних даних (дані можуть бути отримані із камери смартфона, можуть використовуватись RGB-D камери, зокрема, в даній роботі робиться акцент на роботу із 360-degree камерою).

Для дослідження було обрано саме дані приміщень, оскільки це може мати широкий спектр застосувань (наприклад, використовуватись у AR/VR додатках — в якості віртуальних моделей приміщень; також результати дослідження можуть бути корисним доробком до платформ, що займаються продажем та орендою нерухомості, зокрема LUN.ua, OLX, dom.ria).

Серед можливих підходів найбільшу кількість часу приділяється саме Structure-from-Motion, оскільки даний підхід фактично є unsupervised і не вимагає великої кількості даних для навчання. Збір датасетів для 3D-реконструкції є дуже важкою та затратною задачею. До того ж, датасети практично завжди покривають лише певну область (наприклад, в даному випадку — лише приміщення).

Водночас, стисло розглянуті і альтернативні підходи до реконструкції, зокрема нейромережева реконструкція. Перевагами нейромережевих методів є здатність працювати навіть за одним зображенням кімнати, однак при

збільшенні кількості вхідних знімків, за допомогою класичних алгоритмів можна отримати кращу якість ніж у нейромережевій реконструкції.

РОЗДІЛ 1 АНАЛІЗ ІСНУЮЧИХ ПІДХОДІВ ДО ЗАДАЧІ 3D-РЕКОНСТРУКЦІЇ ПРИМІЩЕННЯ

1.1 Опис задачі

Станом на зараз дуже активно розвивається сфера Computer Vision, яка вирішує величезну кількість нетривіальних задач, таких як доповнена та віртуальна реальність, розпізнавання об'єктів, створення їх 3D-моделей тощо. Для вирішення цих проблем уже розроблено багато алгоритмів (деякі з них залишаються актуальними вже понад 10 років, як, наприклад дескриптори зображень FAST, SIFT, ORB). Попри це, вони постійно вдосконалюються, комбінуються та допомагають вирішити нові задачі комп'ютерного зору та фотограмметрії.

Ми живемо в тривимірному світі, де порожній і зайнятий простір визначається фізичною присутністю об'єктів. Щоб успішно орієнтуватися у тривимірному світі та взаємодіяти з ним, важливо розуміти семантику навколишнього середовища та 3D-геометрію. Людина сприймає ці речі вже на інтуїтивному рівні, наприклад, коли вона бачить передню частину будинку, то вона не замислюється, що в нього є і задня частина. Наразі люди працюють над вирішенням багатьох питань, щоб навчити комп'ютер розпізнавати тривимірний світ, а також створювати його віртуальні 3D-моделі, які можна модифікувати. Вирішення таких задач дозволяє виділяти семантичне значення предметів у сцені та виконувати завдання високого рівня. В основі задачі 3D-реконструкції лежить ідея відтворення усіх об'єктів у визначеному просторі та самого простору у віртуальній тривимірній формі. Дану реконструкцію можна проводити як із зображень, так і з потоку відео.

Задача 3D-реконструкції сцени є досить добре вивчена, але якщо проглянути всі дослідження, то вони вельми специфічні. Кожне з них заточене під конкретні критерії, які вимагаються для деякої сфери

(наприклад, реконструкції на вулиці або в приміщенні, з кількох зображень або з відеопотоку, дрібних деталей або великих об'єктів тощо).

До моделі, що виконує реконструкцію приміщення, можна поставити ряд інтуїтивно зрозумілих вимог:

- модель має використовувати невелике число знімків для реконструкції (чим менше, тим краще);
- модель має точно відновлювати 3D-поверхню приміщень (точність — у сенсі метрик для співставлення двох 3D-об'єктів, наприклад 3D Intersection over union);
- модель має якісно відновлювати текстури приміщення (без спотворень);
- модель має враховувати обмеження, що стосуються саме приміщень (наприклад, обов'язкова наявність великих площин, таких як стеля, як несучі стіни чи підлога);

Список обмежень не є вичерпним, однак представляє основні вимоги, які ми висуваємо до реалізованої моделі. У даній роботі в якості вхідних даних обрано саме 360-degree панорами (наявність такого типу вхідних даних має певні переваги перед звичайними знімками телефона, а тому — дозволяє отримати більш якісну реконструкція алгоритму).

1.2 Вхідні дані

Вхідні дані для задачі 3D-реконструкції можуть бути різними. Зокрема, типовими прикладами вихідних даних можуть бути:

- 360-градусні панорами;
- фото, отримані зі звичайного смартфона;

У контексті 3D-реконструкції обрано саме 360-градусні панорами, оскільки це дозволяє отримати реконструкцію високої якості. Також у цьому

розділі буде обговорення переваг і недоліків сферичних панорам (у порівнянні із звичайними знімками)

360-градусна камера — це спеціальний тип камери, яка може зняти зображення із усіх ракурсів навколо неї, створюючи панораму, яка охоплює повний кут огляду 360 градусів.

Особливості 360-градусної камери є те, що вона складається із двох напівсферичних лінз, розташованих одна напроти одної. Це дозволяє охоплювати обидва напрямки із оглядом по 180 градусів, тобто у сумі 360 градусів.



Рис. 1.1 — 360-градусна камера

Оригінально знімки камери нагадують очі риби — дві пари круглих фото (англійською мовою їх називають “fisheye pairs”). Пара “риб’ячих очей” знята із двох протилежних боків камери. Кожен знімок є широкоформатним, охоплює більшу частину сцени:



Рис. 1.2 — початкова форма 360-градусних знімків (fisheye pair)

Перехід від fisheye-пар до сферичної панорами вимагає додаткових обчислень та обробки зображень для створення іммерсивного зображення, яке охоплює повний 360-градусний кут.

Для створення сферичної панорами з fisheye-пар необхідно зробити наступні кроки:

1) Калібрування: Перш за все, необхідно визначити параметри фішай-об'єктива або fisheye-камери, такі як величина спотворення зображення, горизонтальний та вертикальний кут огляду. Це допоможе у виправленні спотворення та коректному приведенні до одного формату.

2) Вирівнювання: Fisheye-пари потрібно вирівняти, щоб їхні геометричні осі збігалися. Це допомагає забезпечити правильне наложення двох півсферичних зображень.

3) Проекція: Після вирівнювання фішай-пари перетворюються на сферичну проекцію. Існує кілька способів проекції, таких як еквіректангулярна, еквіробічна, кубічна або інші, які дозволяють привести зображення до формату сферичної панорами.

4) Зшивання: За допомогою спеціального програмного забезпечення або алгоритмів зшивання зображень, дві сферичні половини поєднуються для створення повної сферичної панорами. Це включає належне вирівнювання, збігання країв та згладжування переходів між двома половинами.

5) Корекція кольору та налаштування: Після зшивання зображення можуть потребувати додаткової обробки, такої як корекція кольору, налаштування контрастності та яскравості, щоб забезпечити оптимальну якість та візуальний вигляд сферичної панорами.

За допомогою описаної вище процедури стітчингу (детально описана у роботі [8]), отримують сферичні панорами із fisheye-пар.



Рис. 1.3 — 360-градусна панорама.

Сферична панорама - це високоякісне зображення, яке охоплює повний кут огляду 360 градусів горизонтально та 180 градусів вертикально. Вона створює ілюзію, ніби ви перебуваєте в самому середині зображення, дозволяючи вам досліджувати його в будь-якому напрямку. За геометричною формою воно нагадує розгортку циліндра, в середині якого перебуває людина разом із камерою.

Маючи сферичну панораму, можна виконати дисторцію і відновити ракурси в шести напрямках (top, bottom, left, right, up, down):



Рис. 1.4 Процедура андисторції

На рисунку 1.4 можна бачити приклад, як виконується відновлення різних ракурсів із одної сферичної панорами. У цьому її перевага порівняно із звичайними знімками зі смартфона — один 360-градусний знімок зберігає інформацію одразу про всі 6 ракурсів, в той час як інформація звичайного знімку є більш обмеженою в плані просторової геометрії.

Для виконання дипломної роботи мною було зроблено багато знімків різних приміщень, зокрема у вигляді 360-градусних панорам:



Рис. 1.5 — Приклад сферичної панорами кімнати



Рис. 1.6 — Приклад сферичної панорами сцени зі сходами

На рисунках 1.5 та 1.6 можна бачити типові приклади сферичних панорам із набору даних, зібраних для дослідження.

У контексті 3D-реконструкції приміщення сферичні панорами мають ряд переваг над звичайними знімками зі смартфона:

1) 360-градусний огляд. Звичайні знімки зі смартфона, незалежно від кількості знімків, можуть не забезпечити такого повністю покриваючого огляду, адже фіксують лише певний кут або певні області приміщення.

Водночас, 360-градусні сферичні панорами надають повністю покриваючий огляд приміщення, оскільки вони фіксують зображення у всіх напрямках одночасно.

2) Інтерактивність: 360-градусні сферичні панорами дозволяють глядачам взаємодіяти з вмістом, обертаючи зображення та досліджуючи простір у всіх напрямках. Звичайні знімки зі смартфона зазвичай не надають такої інтерактивності, оскільки їх можна лише переглядати в 2D-форматі без можливості взаємодії з простором.

3) Глибина та просторова інформація: 360-градусні сферичні панорами можуть містити додаткову інформацію про глибину та простір, що важливо для 3D-реконструкції приміщення. За допомогою спеціальних камер або датчиків глибини, можна створити точніші 3D-моделі приміщення на основі 360-градусних сферичних панорам. Звичайні знімки зі смартфона не мають такої просторової інформації та глибини (якщо це не спеціальна RGB-D камера із LiDAR-датчиком).

4) Якість зображення: Хоча сучасні смартфони можуть мати високу якість камер і здатність до зйомки високої роздільної здатності, 360-градусні сферичні панорами можуть мати ще вищу якість зображення. Завдяки спеціалізованим камерам та оптиці, що використовуються для створення 360-градусних панорам, можна досягти більшої чіткості, деталізації та кольорової точності.

Враховуючи всі вище перераховані переваги сферичних панорам, варто також зазначити і недоліки сферичних панорам перед звичайними знімками зі смартфона:

1) Вимоги до обладнання: Для створення якісних 360-градусних сферичних панорам може знадобитися спеціальне обладнання, таке як високоякісна

камера з можливістю фіксації зображення у всіх напрямках. Це може бути витратно і не зручно для звичайного користувача.

2) Обробка зображень: Створення повноцінної сферичної панорами вимагає обробки і зшивання кількох зображень. Це може бути часо- та ресурсозатратним процесом, особливо якщо використовуються великі обсяги даних.

3) Відсутність глибини: Хоча сферичні панорами можуть передавати широкий кут огляду, вони не забезпечують точної інформації про глибину об'єктів у приміщенні. Це може бути обмеженням у деяких випадках, коли точна просторова інформація є важливою для 3D-реконструкції або аналізу приміщення.

4) Залежність від освітлення: Як і в будь-якій фотографії, якість 360-градусних сферичних панорам може залежати від умов освітлення. Недостатня або неправильна освітленість може призвести до втрати деталей або зниження якості зображення.

Користувачі повинні зважати на свої потреби, обмеження та ресурси для прийняття найкращого рішення. Для персонального використання, певно буде достатньо звичайної камери смартфона. Однак, для широкого проекту і великої кількості даних буде доцільно закупити 360-градусну камеру.

1.3 Результат 3D - реконструкції

Результат 3D-реконструкції приміщення полягає у створенні точної та деталізованої 3D-моделі приміщення з вірними пропорціями, геометрією та внутрішнім простором.

На рисунку 1.3.1 можемо бачити приклад бажаного результату — 3D-модель приміщення, яку можна обертати, переміщатись по приміщенню, крутити і досліджувати із різних боків.



Рис. 1.7 — Приклад 3D-моделі приміщення

Основні вимоги до отриманої 3D-моделі приміщення включають:

1) Точність: 3D-модель повинна відповідати реальним розмірам, формі та геометрії приміщення. Точність є важливою для правильного відображення простору та об'єктів у ньому.

2) Деталізація: 3D-модель повинна включати достатню кількість деталей, щоб передати різні елементи приміщення, такі як стіни, підлога, стеля, вікна, двері, меблі та інші предмети, що знаходяться у ньому. Деталізація допомагає створити реалістичну та іммерсивну 3D-модель.

3) Кольори та текстури: 3D-модель повинна мати правильні кольори та текстури, які відповідають реальному приміщенню. Кольори та текстури допомагають створити візуально привабливу та реалістичну реконструкцію.

4) Глибина та просторова інформація: 3D-модель повинна передавати глибину та просторову інформацію приміщення. Це може включати визначення відстаней між об'єктами, правильне положення предметів у просторі та зв'язки між різними елементами приміщення.

Звичайно, є деякі нерозглянуті чинники, такі як час роботи моделі, залежність від якості вхідних даних. Однак, час роботи моделі в даному випадку не є пріоритетним (модель приміщення будується один раз, і далі буде використовуватись як збережена модель у програмних пакетах MeshLab/Blender/Open3D).

Висновки до розділу 1

3D-реконструкція сцени є масштабною і багатостадійною задачею, яка представляє великий інтерес у сферах комп'ютерного зору і фотограмметрії. Водночас, саме 3D-реконструкція приміщень може мати багато прикладних застосувань, зокрема у сферах AR/VR, у платформах що займаються орендою і продажем нерухомості.

В якості вхідних даних можна використовувати як звичайні знімки зі смартфона, так і сферичні панорами, отримані за допомогою спеціального обладнання. Однак, 360-градусні знімки мають ряд переваг у контексті 3D-реконструкції, тому основний фокус у роботі буде саме на роботі із сферичними панорамами.

Керуючись інтуїцією, можна побудувати ряд вимог до 3D-моделі приміщення, серед яких зокрема точність, якість текстури, збереження геометрії та пропорцій. Час роботи моделі є другорядним чинником, оскільки в даній задачі не вимагається робота застосунку в режимі реального часу.

РОЗДІЛ 2 МАТЕМАТИЧНІ МЕТОДИ 3D-РЕКОНСТРУКЦІЇ

2.1 Форми представлення 3D-об'єкта

Існує багато форм представлення тривимірних об'єктів. Зокрема, у цій роботі будуть розглядатись три основні із них:

- Point Cloud (точкова хмара);
- Triangular Mesh (трикутний меш);
- Voxel Grid (воксельна сітка);

2.1.1 Point Cloud (точкова хмара)

Point Cloud (хмара точок) є найпростішою формою представлення трьохвимірних об'єктів. Це дискретний набір точок у тривимірному просторі (в форматі координат (x, y, z)), до якого часто також включають кодування кольору точки (наприклад, (r, g, b) по шкалі $[0, 1]$ або $[0, 255]$)



Рис. 2.1 Приклади point cloud, що представляють різні 3D-об'єкти

На рисунку 2.1 можемо бачити, як візуально виглядають point cloud для кулі, людини, крісла та стільця. Кожен набір точок наближено представляє поверхню свого об'єкту.

За своїми властивостями хмара точок може бути:

- щільною або розрідженою (dense point cloud, sparse point cloud);
- зашумленою (noisy point cloud)
- кольоровою або мати виключно координати точок (формати (xyz) або (xyz+rgb));

Точкові хмари можна отримати із стерео-камер, LiDAR-сенсорів, камер зі структурованим освітленням. Point Cloud, що вже згенерований, вже можна далі обробляти, застосовувати до нього семплінг, конволюції, та інші математичні операції що поширені у машинному навчанні.

Точкові хмари можуть описувати досить складну геометрію об'єкту і його деталі. Однак, через велику кількість точок використання точкових хмар може бути обчислювально затратним, що стає проблемою у вирішенні великої кількості задач.

2.1.2 Triangular Mesh (трикутний меш)

Mesh — це сукупність вершин (vertices), ребер (edges) і граней(faces), що з'єдані між собою у сітку (мережу). Подібно як і точки в point cloud, можна сказати що елементарним елементом меша є трикутні грані. Зазвичай за основу грані вибирають опуклу геометричну фігуру (найбільш поширений випадок — трикутний меш).

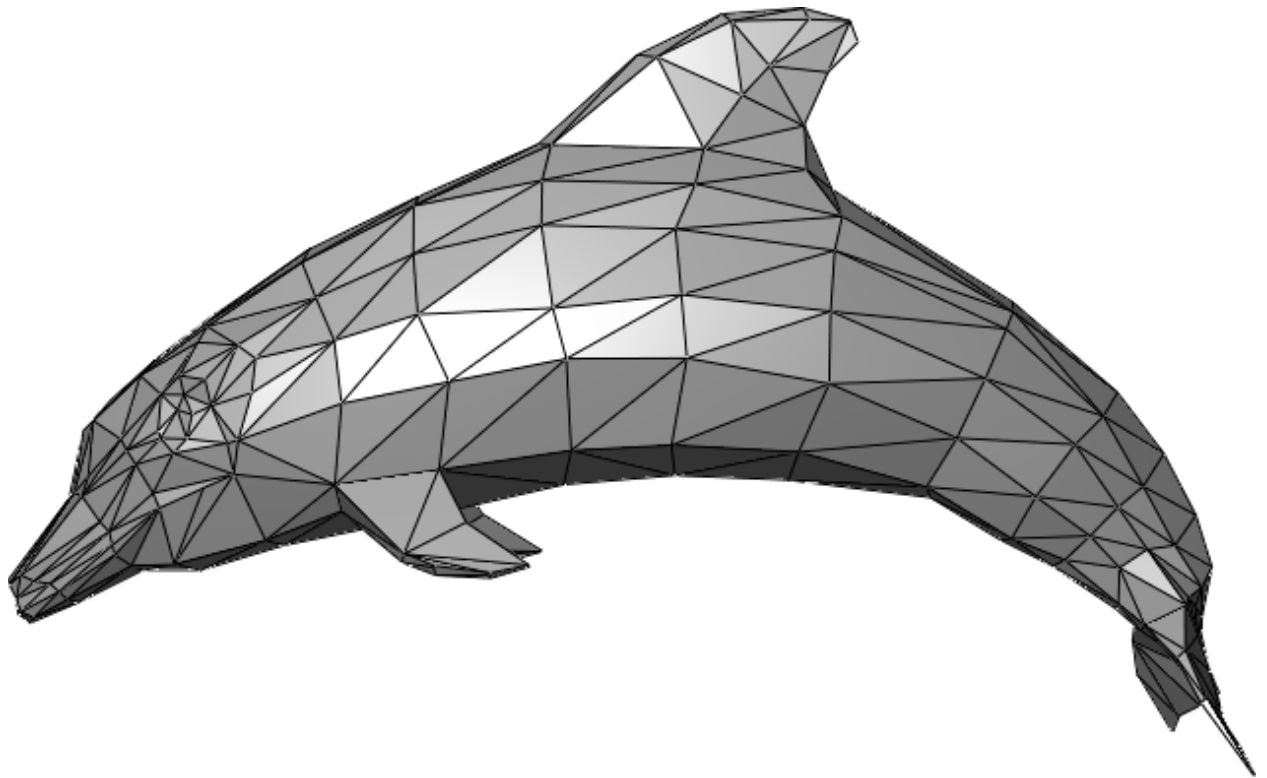


Рис. 2.2 — Приклад мешу, що зображує модель дельфіна

На рисунку 2.2 можемо бачити приклад трикутного меша, що є об'ємним представленням дельфіна. Меш складається із вершин, деякі вершини з'єдані між собою і утворюють ребра, в свою чергу, ребра з'єдані у суміжні трикутники і утворюють грані.

Як і point cloud, mesh може налічувати різну кількість структурних елементів. Залежно від деталізації об'єкта та складності поверхні, ця кількість може варіюватись і змінюватись.

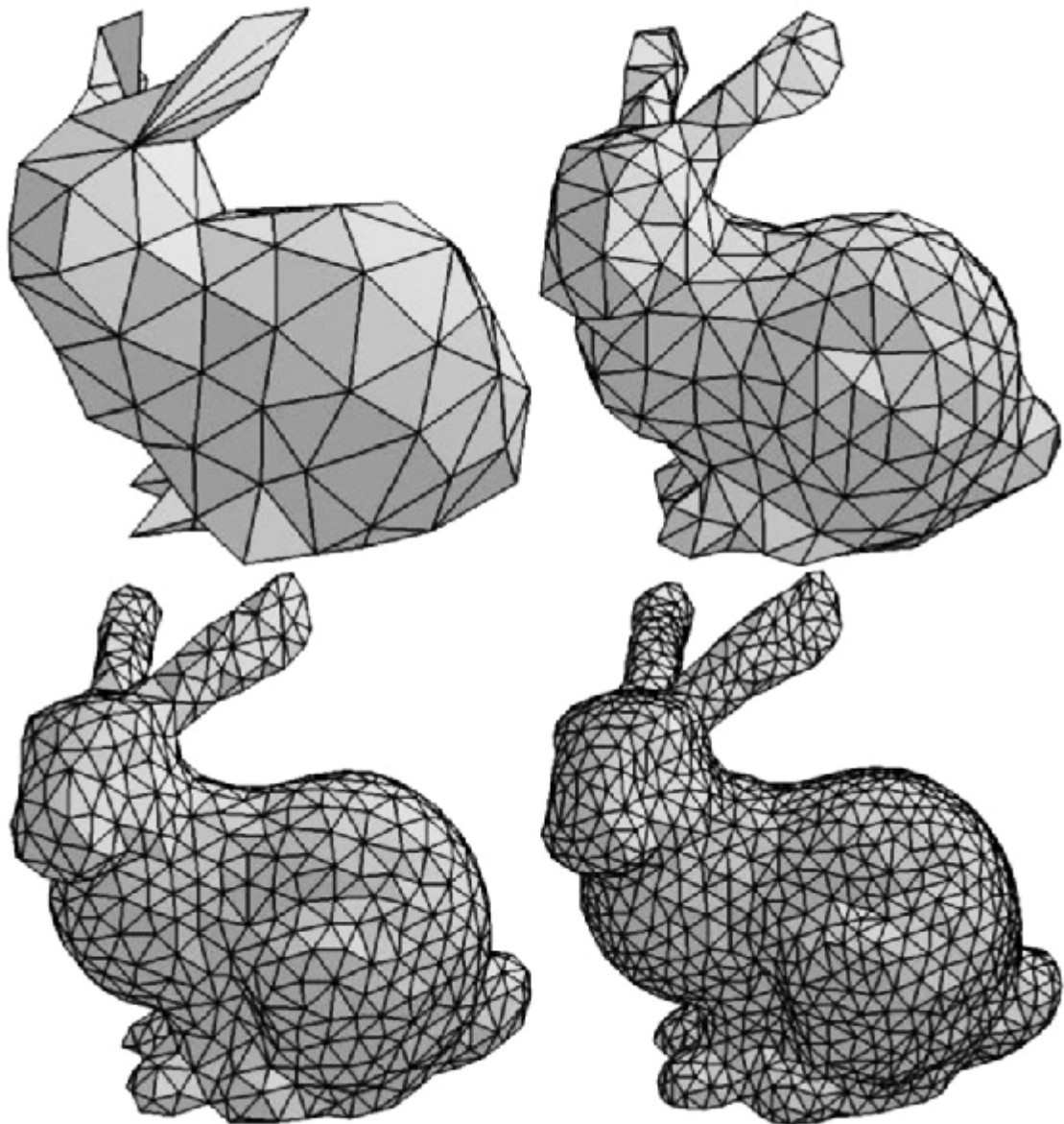


Рис. 2.3 Різний рівень деталізації об'єкта у mesh-і

На рисунку 2.3 можемо наглядно бачити, як може змінюватись структура меша в залежності від рівня деталізації об'єкта на прикладі 3D-моделі кролика. Чим менше розмір елементарного трикутника у сітці, відповідно тим більше їх кількість, відповідно і зростає рівень деталізації об'єкта.

При порівнянні меша із точковою хмарою можна виділити деякі переваги і недоліки меша. До переваг можна віднести більшу ефективність

по пам'яті — трикутниками можна досить детально покрити поверхню і при цьому використати менше точок, ніж потрібно було б для аналогічної задачі із точковою хмарою. Також меш дає повністю представлення поверхні, об'єкти у формі мешу можна використовувати у додатках CAD (computer aided design) та для 3D-друку.

Недоліки — менша гнучкість у плані захоплення складної геометрії. Іноді поверхня може бути досить гладкою і добре описуватись деяким рівнянням поверхні (наприклад, поверхня сфери), а коли конструюється меш, ми її намагаємось апроксимувати сукупністю площин (кожен трикутник можна вважати площиною). Відповідно, доводиться або будувати дуже детальний меш, що може займати багато пам'яті; або менш детальний, тоді доводиться миритись із “угловатістю” поверхні.

2.1.3 Voxel grid (воксельна сітка)

Третім із основних типів представлення 3D-об'єктів є Voxel Grid (воксельна сітка). Це структура даних, що представляє собою тривимірну сітку, розбиту на регулярні об'ємні елементи, що називаються вокселями. Кожен воксель є елементарним тривимірним кубиком мінімального розміру, аналогічно як піксель є елементарним квадратиком найменшого розміру на двохвимірному зображенні. Воксель містить інформацію про властивості об'єкта саме у даній точці простору (наприклад, колір, прозорість, щільність і т.д.). Вокселі також використовуються для представлення геометричних даних, таких як форма і положення.

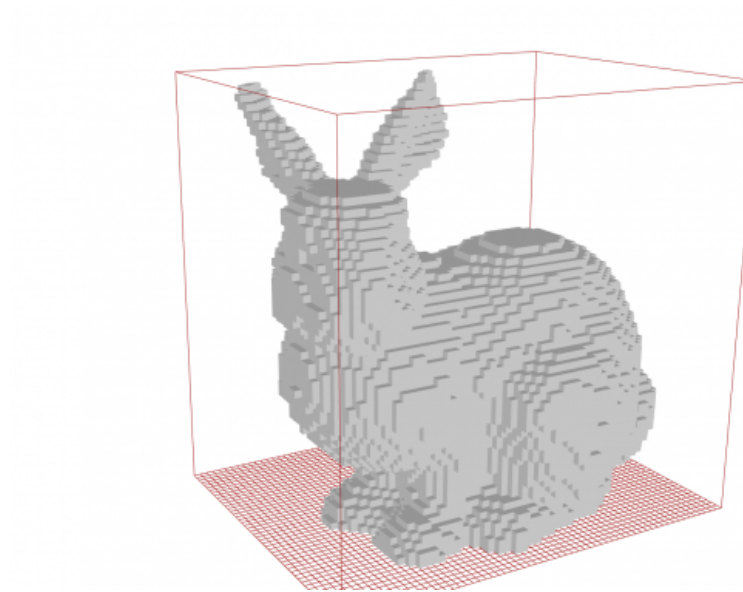


Рис. 2.4 — воксельна сітка для кролика

На рисунку 2.4 можемо бачити воксельну сітку тривимірної моделі кролика. Аналогічно як у меші можна змінювати розмір елементарного трикутника, у воксельних сітках можна змінювати розмір вокселя. Чим менший розмір вокселя, тим більш деталізованою є відновлена поверхня яку він репрезентує:

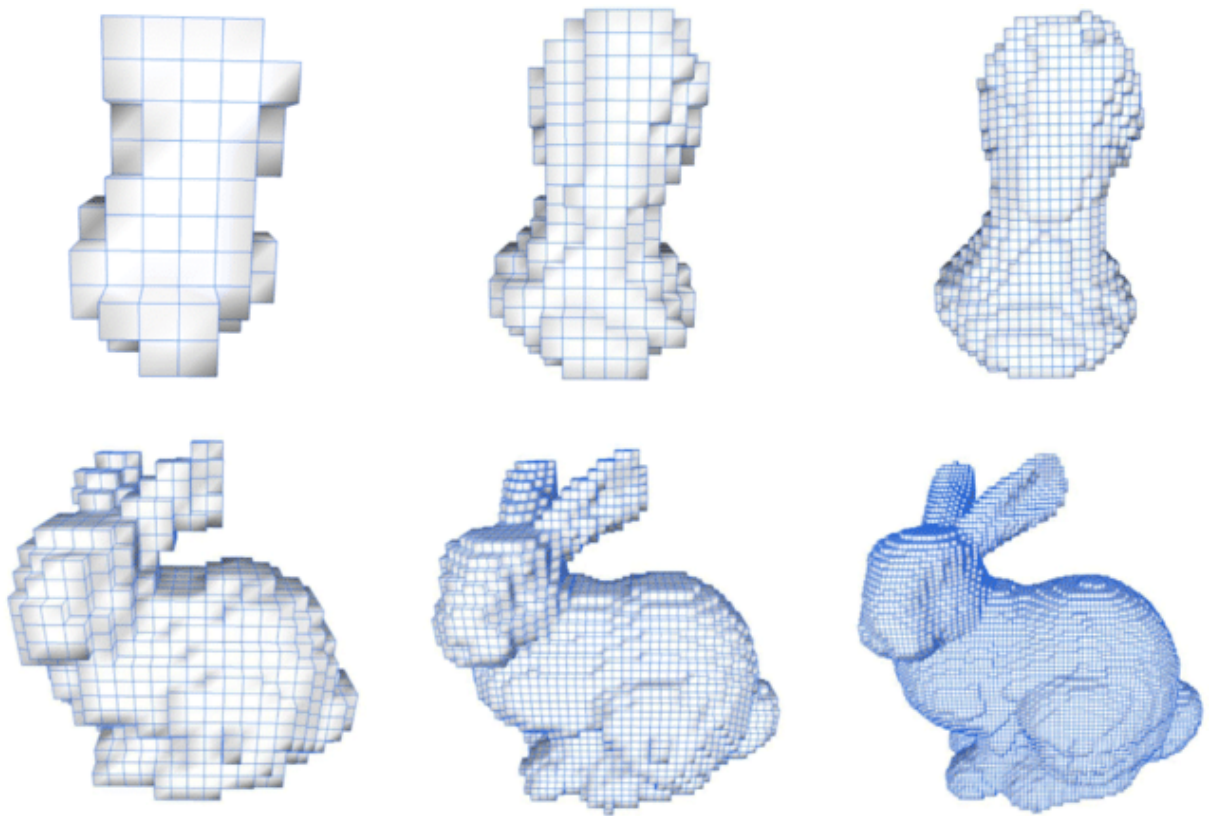


Рис. 2.5 — Деталізація об'єкта при різних розмірах вокселя

Розмір вокселя та роздільна здатність сітки може змінюватись в залежності від конкретного застосування. Чим менше розмір вокселя та більша роздільна здатність сітки, тим більш деталізованим буде об'єкт, однак і вимоги до обчислювальних ресурсів також будуть вищими.

До переваг воксельних сіток можна віднести здатність захоплювати складну геометрію об'єкта, врахування додаткових властивостей (таких як колір, щільність). Багато алгоритмів обробки об'ємних даних легше всього застосовувати саме на воксельних сітках. Також перевагою є простота - воксель є простою та інтуїтивно зрозумілою структурою даних. Однак є і очевидні недоліки — великі вимоги до пам'яті та обчислювальних ресурсів (зберігати воксельну структуру у пам'яті дорожче, ніж аналогічний меш чи точкову хмару).

2.2 Structure-from-Motion

Техніка Structure-from-Motion (SfM) є одним із ключових підходів у комп'ютерному зорі для отримання тривимірної інформації з двовимірних зображень. SfM дозволяє відтворити тривимірну сцену та реконструювати 3D-модель об'єкта або приміщення на основі набору зображень, отриманих під час послідовних рухів камери у просторі (камера або крутиться навколо оператора, або переміщається разом із ним відносно приміщення).

Основні етапи підходу Structure-from-Motion:

- feature detection (детекція ознак);
- feature matching (співставлення ознак);
- відновлення позицій камери та 3D-координат точок об'єкта;
- refinement (подальша робота із точковою хмарою — denoising, densification, meshing, texturing).

2.2.1 Feature detection

На етапі детекції ознак знаходимо важливі ознаки на кожному зображенні, такі як ключові точки, кути, ребра. Ключові точки є унікальними та стабільними областями на зображенні, які можна використовувати для подальшого відтворення та знаходження відповідностей між зображеннями.

Знаходження ключових ознак може виконуватись за допомогою дескрипторів зображень, таких як SIFT (Scale-invariant feature transform),

SURF (Speeded up robust features), ORB (Oriented fast and Rotated BRIEF), HOG (Histogram of gradients).

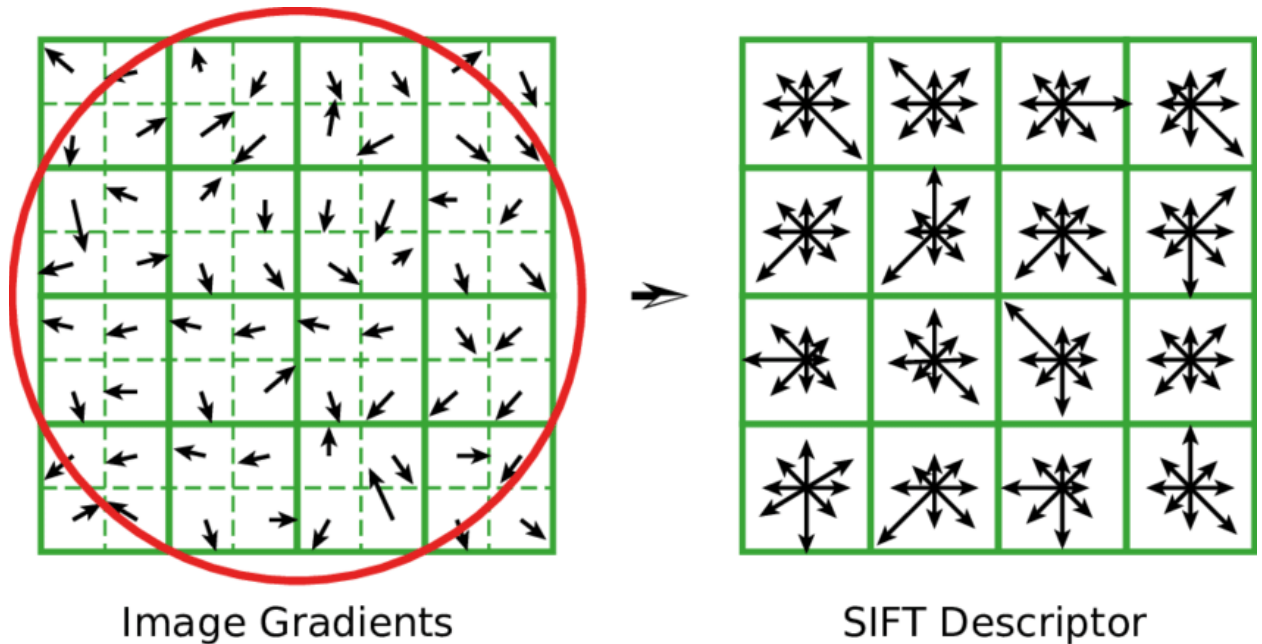


Рис. 2.5 — Дескриптор точки на двовимірному зображенні

Дескриптор точки на зображенні є числовим вектором, який представляє характеристики або особливості цієї точки. Зокрема, він включає інформацію про локальне оточення точки, таку як градієнти яскравості або орієнтація пікселів у малому вікні навколо точки. Ці характеристики використовуються для створення унікального опису точки, який може бути використаний для виявлення та відповідання точок на різних зображеннях.

Вектори, що описують ключові точки зображень, далі будуть використовуватись для співставлення точок на різних зображеннях. Кількість ключових точок є гіперпараметром алгоритму, який обирає користувач, розмірність же цього вектору визначається особливостями алгоритму дескриптора (наприклад, SIFT розраховує 128-вимірний вектор для кожної ключової точки, ORB — розмірності 64).



Рис. 2.6 - Ключові точки на двох панорамах (позначено червоним)

На рисунку показано візуалізацію процесу детекції ключових ознак — відображено ключові точки фото червоними кругами. Таким чином, наприкінці етапу детекції ознак, ми маємо для кожної ключової точки:

- двовимірні координати (x, y) на зображенні;
- числовий вектор, що описує найближчий окіл навколо точки (градієнти кольору навколо неї, орієнтація пікселів).

Кількість ключових точок та вибір алгоритму дескриптора є предметом для дискусій, оскільки це все підбирається із урахуванням особливостей вхідних даних.

2.2.2 Feature matching

На етапі feature matching виконується співставлення ключових ознак, що отримані із різних зображень кімнати. Тривимірна точка матиме різні координати (x, y) на кожному із зображень, оскільки між знімками приміщень відбувався рух камери у тривимірному просторі. Відповідно, зсуваються і координати проєкцій тривимірної точки на двовимірне зображення.

Процедура матчіну виконується із використанням дескрипторів ключових точок, що були розраховані на попередньому етапі.

Наведемо приклад, як відбувається матчінг ключових точок на зображеннях.

1) Припустимо, що ми маємо точку на першому зображенні і три точки на другому зображенні. Для спрощення, нехай наш дескриптор - розмірності 8:

descriptor1_1: [25 207 66 172 148 147 207 245]

descriptor2_1: [57 207 66 174 148 147 207 245]

descriptor2_2: [57 207 66 174 148 147 207 246]

descriptor2_3: [57 207 66 178 148 147 207 246]

Схожість двох дескрипторів можна вимірювати типовими у лінійній алгебрі метриками, зокрема L2-відстанню (евклідова відстань), L1-відстанню (манхеттенська відстань), косинусною схожістю, відстанню Хеммінга тощо.

Порахуємо манхеттенську відстань між трьома парами точок:

$$\text{delta_11} = (57-25) + 0 + 0 \dots = 32$$

$$\text{delta_12} = (57-25) + 0 + 0 \dots + (246-245) = 33$$

$$\text{delta_13} = (57-25) + 0 + 0 \dots + (178-174) + (246-245) = 37$$

Розрахувавши відстань між векторами, можемо сказати, що за манхеттенською відстанню найближчою до точки 1_1 є точка 2_1. Це і є процедура матчіngu — співставити дескриптори ключових точок і знайти найкращі збіги.

2) Піднімаємось на рівень абстракції вище (порівняння двох зображень). У контексті порівняння двох зображень — ми маємо тензор розмірності [n_kp1, n_dim] для першого зображення та [n_kp2, n_dim] для другого зображення.

Фактично, тепер треба порахувати усі попарні відстані між дескрипторами перших зображень, і дескрипторами других зображень, виявити пари найкращих збігів.

Задача пошуку найкращого збігу називається англійською ANN (approximate nearest neighbors search). Обчислення точних найближчих сусідів векторів може бути обчислювально затратним і неефективним, тому використовують

ряд алгоритмів для швидкого наближеного пошуку сусідів у великому наборі даних. У випадку зображень, розмірність тензорів вимірюється у сотнях або тисячах, тому тут доцільно використовувати наближені методи пошуку сусідів.

Результатом співставлення ключових точок є тензор розмірності $[n_matches, (x1, y1, x2, y2)]$ — пари точок, що відповідають найкращим збігам. Для фільтрації збігів можна використовувати пороги по відстані (наприклад, максимально можлива відстань — 10), або сортувати збіги по відстані між дескрипторами і вибирати певний відсоток від найкращих збігів (поширена практика — обирати 10-20% від загального числа збігів).



Рис. 2.7 — Співставлення отриманих ключових точок

3) Припустимо, що ми виконали співставлення між кожною парою зображень у наборі (типові скани приміщень налічують 10-20 фотознімків). Далі виконується пошук спільних точок, що наявні на усіх зображеннях набору даних про кімнату (наприклад точка $A1(x1, y1)$ відповідає точкам $A2(x2, y2)$, $A3(x3, y3)$, $A4(x4, y4)$ ) Складається матриця спостережень, яка далі буде активно використовуватись для відновлення тривимірної структури об'єкта.

2.2.3 Camera Pose Estimation + Reconstruction.

Коли ми визначили ключові точки між різними зображеннями, Structure-from-motion оцінює позицію та орієнтацію камери (для кожного із зображень — відносно сцени). Один із способів відновити позицію камери — використовуючи матричну декомпозицію (Tomasi-Kanade factorization):

$$\begin{aligned}
 \tilde{u}_{f,p} &= \mathbf{i}_f^T \mathbf{P}_p \\
 \tilde{v}_{f,p} &= \mathbf{j}_f^T \mathbf{P}_p
 \end{aligned}
 \quad \Rightarrow \quad
 \begin{bmatrix} \tilde{u}_{f,p} \\ \tilde{v}_{f,p} \end{bmatrix} = \begin{bmatrix} \mathbf{i}_f^T \\ \mathbf{j}_f^T \end{bmatrix} \mathbf{P}_p$$

	Point 1	Point 2	...	Point N			Point 1	Point 2	...	Point N
Image 1	$\tilde{u}_{1,1}$	$\tilde{u}_{1,2}$	...	$\tilde{u}_{1,N}$	=	$\begin{bmatrix} \mathbf{i}_1^T \\ \mathbf{i}_2^T \\ \vdots \\ \mathbf{i}_F^T \\ \mathbf{j}_1^T \\ \mathbf{j}_2^T \\ \vdots \\ \mathbf{j}_F^T \end{bmatrix}$	$\mathbf{P}_1$	$\mathbf{P}_2$	...	$\mathbf{P}_N$
Image 2	$\tilde{u}_{2,1}$	$\tilde{u}_{2,2}$	...	$\tilde{u}_{2,N}$						
	$\vdots$	$\vdots$	$\vdots$	$\vdots$						
Image F	$\tilde{u}_{F,1}$	$\tilde{u}_{F,2}$	...	$\tilde{u}_{F,N}$						
Image 1	$\tilde{v}_{1,1}$	$\tilde{v}_{1,2}$	...	$\tilde{v}_{1,N}$						
Image 2	$\tilde{u}_{2,1}$	$\tilde{u}_{2,2}$	...	$\tilde{v}_{2,N}$						
	$\vdots$	$\vdots$	$\vdots$	$\vdots$						
Image F	$\tilde{v}_{F,1}$	$\tilde{v}_{F,2}$	...	$\tilde{v}_{F,N}$						

$W_{2F \times N}$
**Centroid-Subtracted
Feature Points (Known)**

$M_{2F \times 3}$
**Camera Motion
(Unknown)**

$S_{3 \times N}$
**Scene Structure
(Unknown)**

Рис. 2.8 Матричне рівняння відносно позиції камери

На рисунку 2.8 зображено приклад матричного розкладу, що використовується в пайплайні Structure-from-Motion. Маючи матрицю спостережень із кроків 1 та 2 (ключові точки на зображеннях та їх взаємна відповідність на різних зображеннях), можемо використати низькорангове матричне розкладання і таким чином відновити позиції камери та тривимірну структуру сцени.

Далі Structure-from-motion відновлює point cloud розміром $3 \times N$ (отриманий із матричного розкладу, або іншого способу відновлення положення камери).

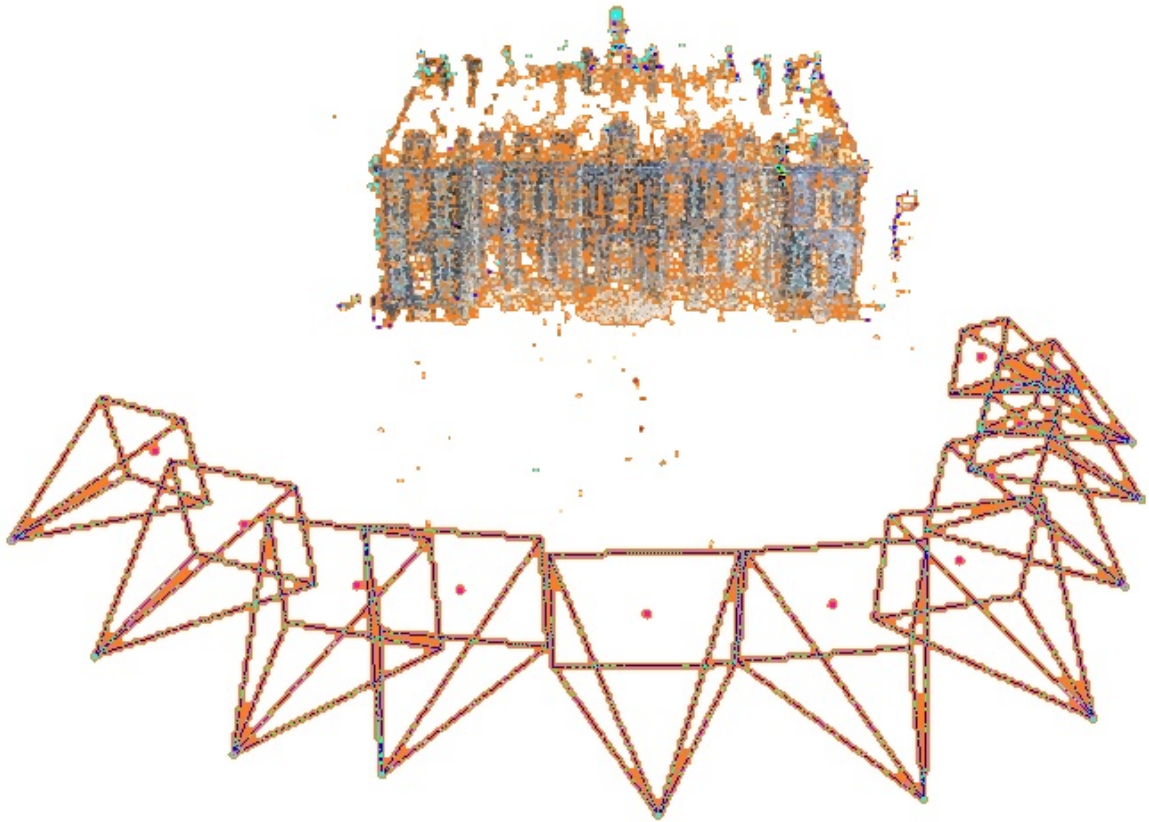


Рис. 2.9 — Візуалізація позицій камери відносно деякого об'єкта

Одним із чисельних методів, що дозволяє виконати відновлення двох невідомих матриць обчислювально ефективно — SVD (сингулярну декомпозицію матриці спостережень). Процедуру обчислення матриць M та S можна описати трьома основними кроками:

- обчислення сингулярної декомпозиції матриці W , із встановленням обмеження на ранг матриці ($\text{rank } W \leq 3$):

$$W = U_1 \Sigma V_1^T$$

- припускаємо, що матриці M та S дорівнюють

$$\begin{aligned} M &= U_1 (\Sigma_1)^{1/2} Q \\ S &= Q^{-1} (\Sigma_1)^{1/2} V_1^T \end{aligned}$$

де Q — ортонормована матриця 3×3 ;

- знаходимо матрицю Q із системи рівнянь (ортонормована система векторів):

$$\left. \begin{aligned} \mathbf{f}_i^T Q Q^T \mathbf{f}_i &= 1 \\ \mathbf{f}_i^T Q Q^T \mathbf{f}_j &= 1 \\ \mathbf{f}_i^T Q Q^T \mathbf{f}_j &= 0 \end{aligned} \right\}$$

Оскільки матрична декомпозиція SVD існує для будь-якої прямокутної матриці, цей метод можна використовувати і для матриці спостережень. Розклад не є єдиним, тому додатково у системи рівнянь включається ортогональна матриця Q і додаткові обмеження на неї.

Фундаментальним фактом є спостереження Томаса і Канаде — матриця спостережень, складена саме таким чином, має ранг не більше 3, оскільки отримана вона шляхом переміщення камери у тривимірному просторі та виконанням проєкцій на 2D-зображення.

2.2.4 Point Cloud densification (ущільнення точкової хмари)

Point Cloud, отриманий одразу після відновлення 3D-координат точок об'єкта, зазвичай містить багато шумових точок та є розрідженим. Наступним кроком у 3D-реконструкції є покращення сукупності точок, за рахунок видалення шумових точок та нарощення щільності точкової хмари.

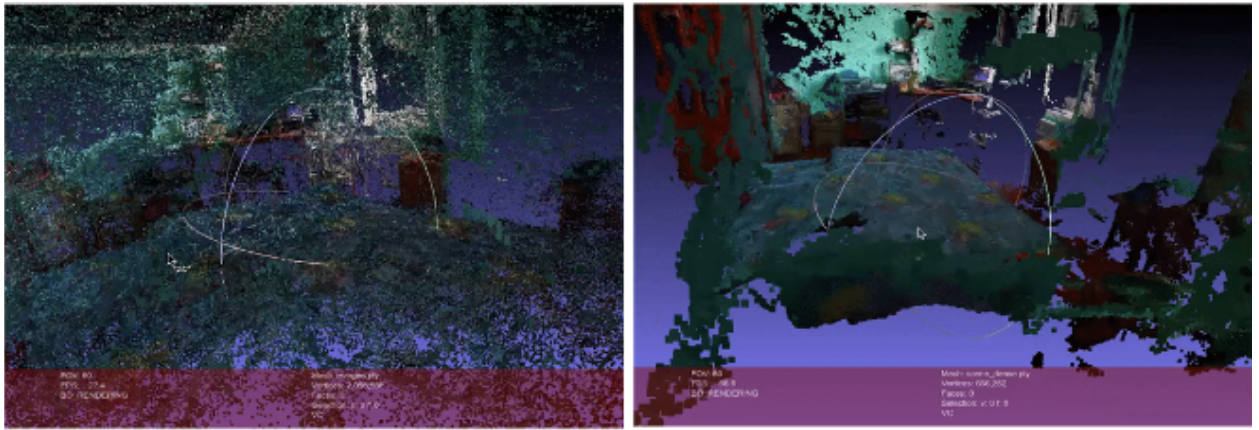


Рис. 2.10 Різниця між розрідженою та щільною точковою хмарою

Існують два основних алгоритми, які використовуються для ущільнення точкової хмари — триангуляція Пуассона та алгоритм Делоне.

За своєю природою ці два алгоритми різні — алгоритм Пуассона є статистичним і при відновленні поверхні припускає, що 3D-поверхню можна описати теоретичною щільністю багатовимірного нормального розподілу. Як наслідок, поверхні отримані реконструкцією Пуассона часто є гладкими, із дзвоноподібними опуклостями.

Підхід Делоне є більш структурним - базується на утворенні трикутників з точок набору даних таким чином, щоб жодні трикутники не містили точок усередині їх вписаних колів. Це структурний підхід, який покриває поверхню об'єкта трикутниками без їх перетину.

2.2.5 Текстуризація отриманого мешу

Фінальний крок у задачі 3D-реконструкції — текстурування мешу, тобто накладення двохвимірних текстур на отриману тривимірну поверхню. Процедура текстурування мешу, або накладання текстури на поверхню моделі, включає кілька кроків, які дозволяють призначити зображення або

текстуру на поверхню об'єкта. Основні кроки процедури текстуровання мешу наступні:

1) Розгортання мешу (UV Unwrapping): Цей крок включає розгортання поверхні мешу на плоску просторову мапу, яка називається UV-карта. У цьому процесі кожен вершині або полігону мешу присвоюється відповідна позиція на UV-карті. Розгортання може бути виконано різними способами, включаючи автоматичне розгортання або ручне редагування UV-координат.

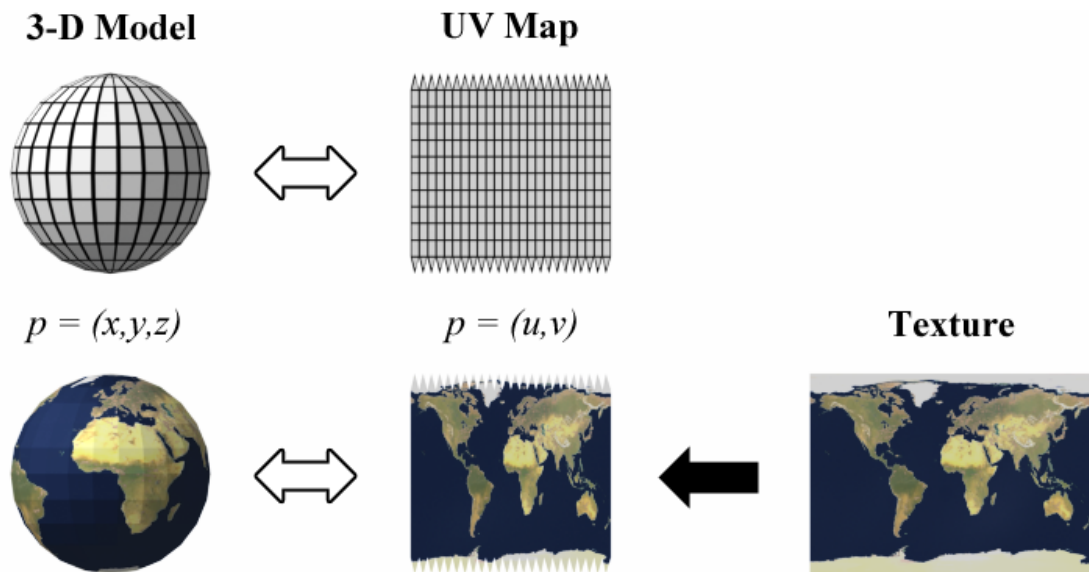


Рис. 2.11 — Приклад UV-тар для сферичної поверхні Землі

2) Вибір текстури: На цьому етапі вибирається сама текстура, яка буде накладена на поверхню мешу. Текстура може бути створена самостійно або використана з готового зображення.

В даному випадку текстури вибираються із сферичних панорам, що були надані на вхід алгоритму.

3) Налаштування UV-координат: Після розгортання мешу на UV-карту, UV-координати кожної вершини або полігона відповідають певним точкам на текстурі. У цьому кроці виконується відповідність між UV-координатами на UV-карті та пікселями на текстурі.

4) Накладання текстури: На основі встановлених UV-координат та текстури виконується саме накладання текстури на поверхню мешу. Колір або значення пікселя з відповідної позиції на текстурі передається на відповідну вершину або полігон мешу.

5) Рендеринг: Після накладання текстури на поверхню мешу може бути виконано рендеринг сцени або моделі для створення реалістичного зображення з текстурою.

2.3 Глобальний та інкрементальний підходи

Практично до кожної задачі, розглянутої у розділі 2, існують глобальний та інкрементальний підходи. Глобальний та інкрементальний підходи є двома основними стратегіями розв'язання задачі:

Глобальний підхід передбачає обробку всього набору даних або задачі в єдиному циклі. Він вимагає наявності повного набору даних на початку обробки і виконує аналіз та обчислення над цим набором даних. Основні особливості глобального підходу:

1) Обробка всього набору даних: Глобальний підхід виконує аналіз та обчислення на всьому наборі даних одночасно.

2) Більш точні результати: Оскільки глобальний підхід використовує всі дані одночасно, він може забезпечити більш точні результати, особливо в складних задачах аналізу даних.

3) Висока обчислювальна складність: Глобальний підхід може бути обчислювально витратним, особливо при великих об'ємах даних. Обробка всього набору даних одночасно може вимагати значних обчислювальних ресурсів і часу.

Інкрементальний підхід передбачає обробку даних поетапно або почастинно. Дані обробляються порціями або блоками, і результати обробки кожної порції використовуються для подальшої обробки. Основні особливості інкрементального підходу:

- 1) Послідовна обробка: Інкрементальний підхід обробляє дані поетапно, що дозволяє обробляти великі об'єми даних поступово і ефективно.
- 2) Ефективність за рахунок розподілу завдань: Інкрементальний підхід може бути більш ефективним, оскільки завдання можуть бути розподілені між декількома обчислювальними вузлами або пристроями, що прискорює обробку даних.
- 3) Можливість використання неповного набору даних: Інкрементальний підхід може працювати з неповним набором даних або з даними, які надходять у реальному часі.

У контексті 3D-реконструкції приміщення можуть використовуватись як глобальний, так і інкрементальний підходи Structure-from-Motion, кожен із них мають свої переваги і недоліки.

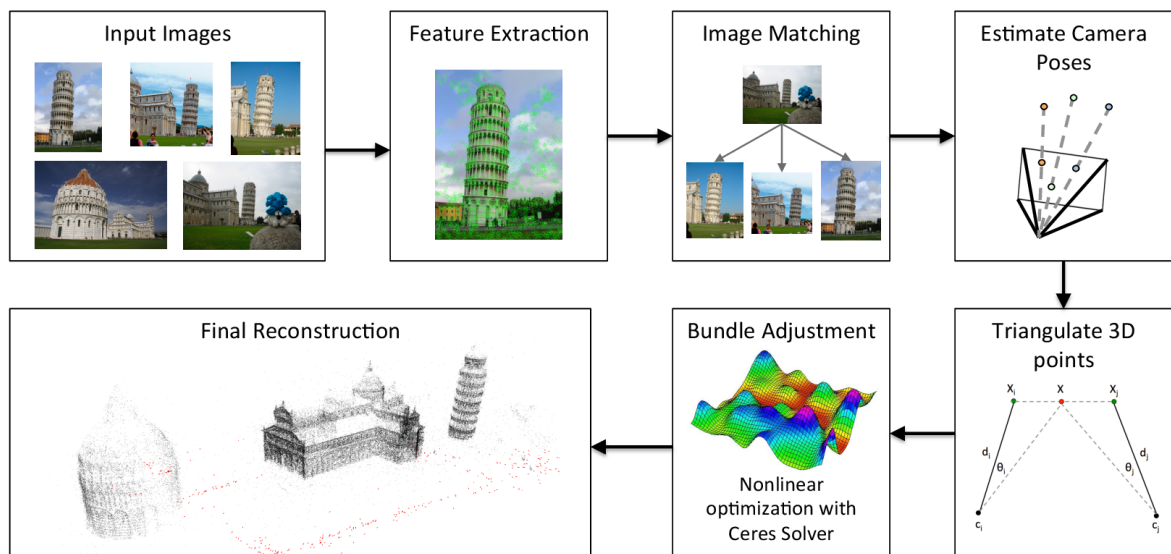


Рис. 2.11 Схема глобального Structure-from-Motion

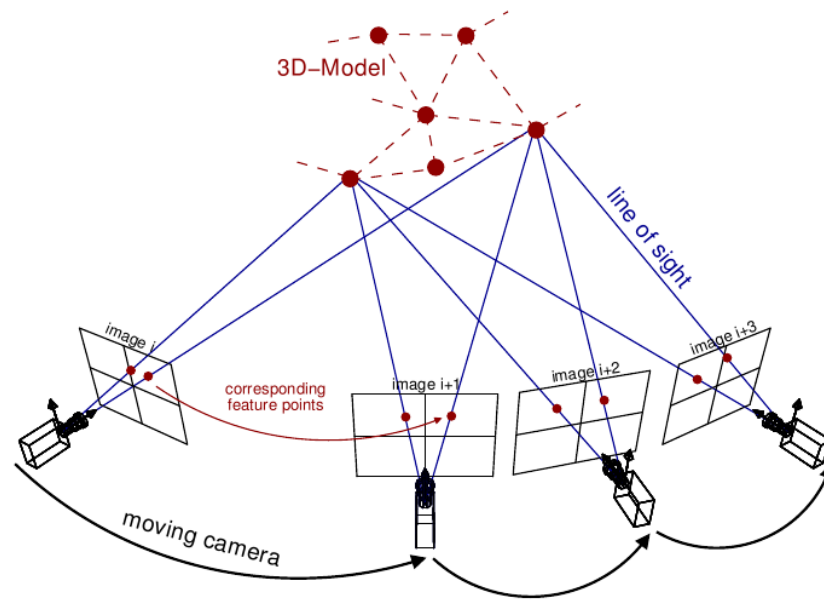


Рис. 2.12 Схема інкрементального Structure-from-Motion

Глобальний підхід зазвичай дає кращу глобальну карту приміщення, оскільки він використовує всі доступні дані одночасно. Однак, в цьому ж є його недолік - вимогливість до обчислювальних ресурсів, особливо при обробці великих наборів сферичних панорам.

У роботі в основному використовується глобальний підхід, оскільки, зважаючи на вимоги до 3D-моделі які ми висуваємо на початку роботи, пріоритетними для нас є якість 3D-реконструкції (якість текстур, пропорції, геометричні обмеження). На користь інкрементального обирають зазвичай, при наявних обмеженнях у часі або обчислювальних ресурсах. Зокрема, інкрементальний Structure-from-motion доцільно використовувати у додатках для робототехніки, якій необхідно орієнтуватися у просторі в режимі реального часу.

Висновки до розділу 2

У розділі 2 було детально проаналізовано математичні основи задачі 3D-реконструкції, такі як: форми представлення 3D-об'єктів, підхід Structure-from-Motion, фундаментальні задачі які у ньому виникають, порівняння інкрементальної та глобальної стратегії реконструкції. Сформульовано можливі алгоритми на кожному із етапів реконструкції. Проведено аналіз переваг і недоліків кожного із алгоритмів.

РОЗДІЛ 3 АНАЛІЗ ПРАКТИЧНИХ РЕЗУЛЬТАТІВ

У цьому розділі наведено кількісні та якісні результати, отримані при реконструкції приміщень. Наведено порівняльний аналіз результатів 3D-реконструкції при різних конфігураціях Structure-from-Motion.

3.1 Опис зібраних даних

Для дослідження роботи алгоритму було обрано дві основні сцени — житлова кімната та сцена зі сходами. За допомогою 360-градусної камери Insta360, яка мені була надана консультантом для проведення експериментів, було зроблено 19 панорамних знімків житлової кімнати, та 29 панорамних знімків сцени зі сходами.



Рис. 3.1.1 — Приклад панорами із набору “Житлове приміщення”



Рис. 3.1.2 — Приклад панорами із набору “Сцена зі сходами”

Кількість панорамних знімків для приміщення підбиралась із урахуванням існуючих наборів даних по приміщенням (наприклад, у датасеті ScanNet — в середньому близько 20 знімків на приміщення, однак ScanNet — це дані звичайних знімків, зроблених зі смартфона).

Інтуїтивно, сцена зі сходами має складнішу, двохповерхову структуру, плюс сам об'єкт — сходи, не є типовим паралелепіпедом, яким є приміщення житлової кімнати.

3.2 Noised point cloud (зашумлена точкова хмара)

Одним із проміжних продуктів реконструкції є зашумлена точкова хмара, яку ми отримуємо одразу після відновлення відносних позицій камери. Було проведено експерименти із різною кількістю вхідних фото, для того щоб визначити оптимальну кількість фото приміщень. Теоретично, мінімальна кількість фото для роботи алгоритму Structure-from-motion — 3. На практиці було перевірено пре-сети у 3, 8, та всіх (19) панорамних знімків кожного приміщення.

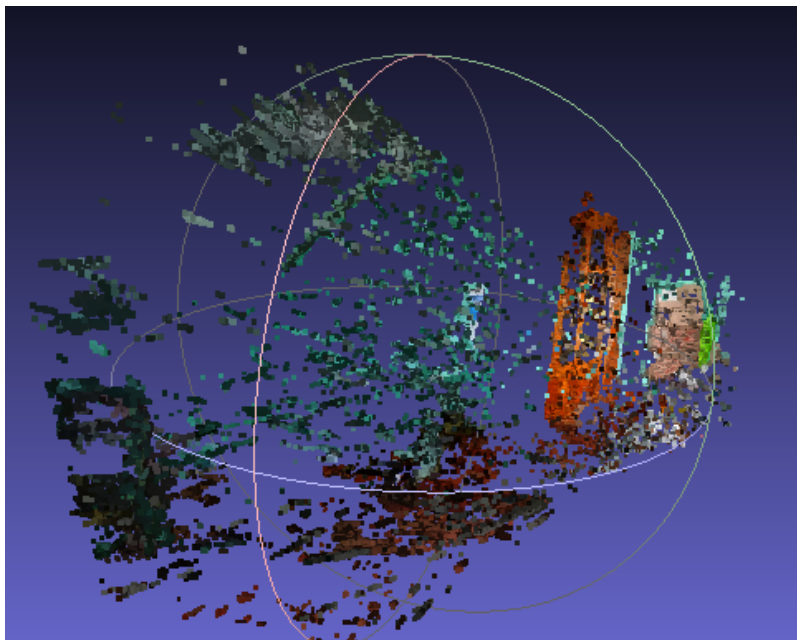


Рис. 3.1 Точкова хмара для 3 зображень (житлове приміщення)

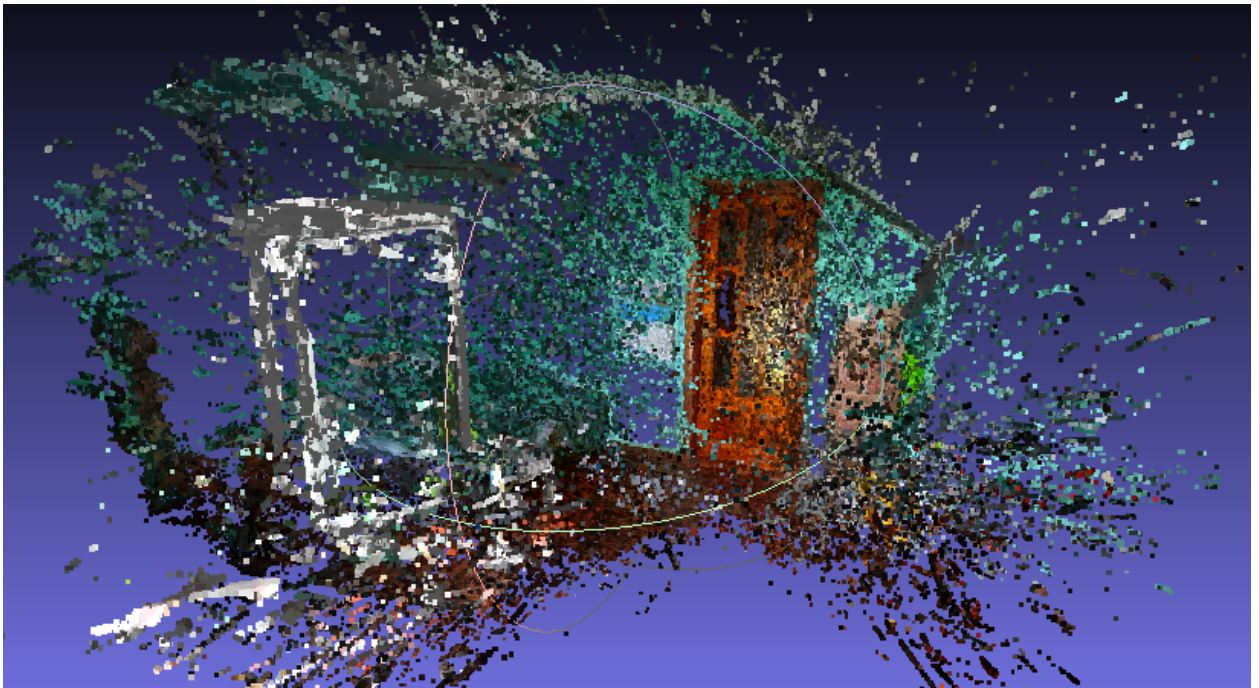


Рис. 3.2 — Точкова хмара для 8 вхідних зображень

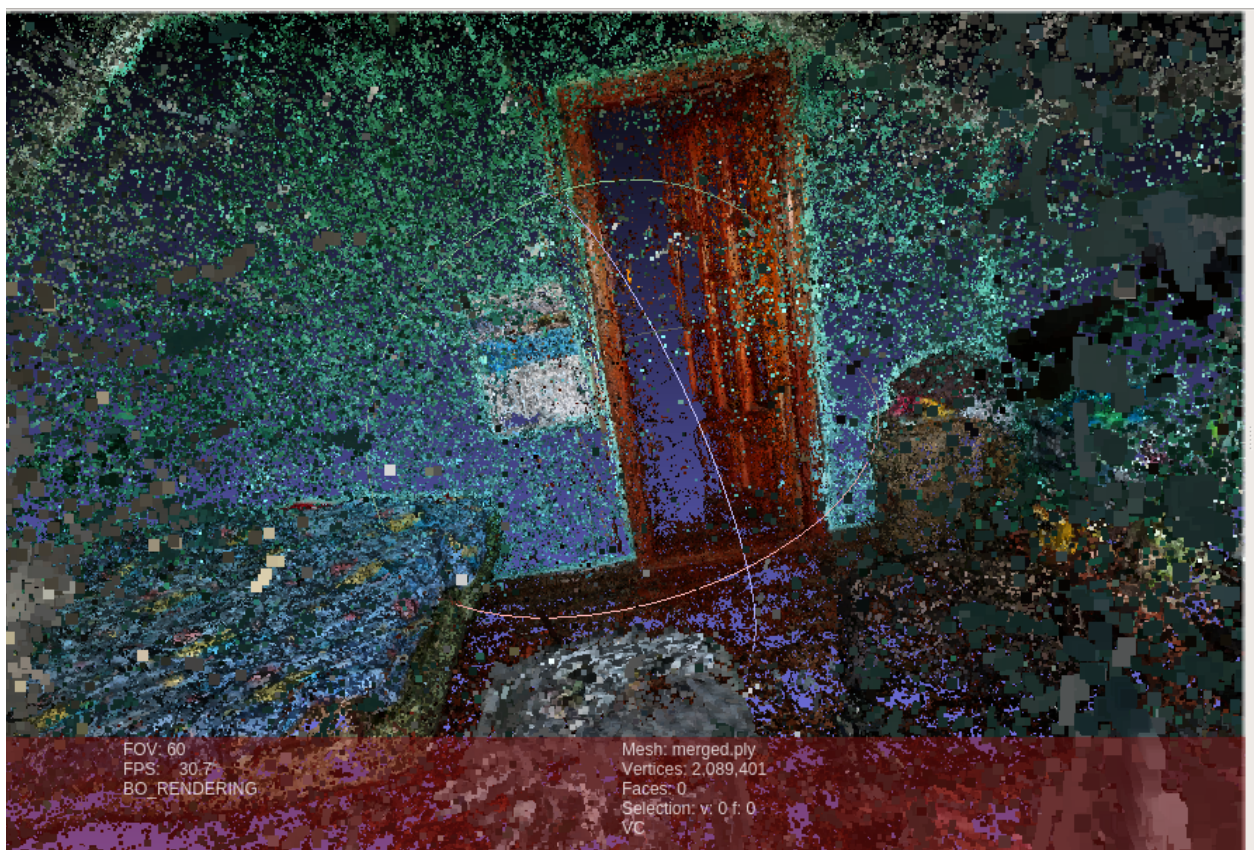


Рис. 3.3 — Точкова хмара для 19 вихідних зображень.

Порівняння показало що даних, отриманих із 3 і 8 сферичних панорам недостатньо для подальшої реконструкції приміщення. Відповідно, при 19

сферичних панорамах як вхідних даних, отримуємо точкову хмару що майже повністю описує приміщення.

Результат аналогічного експерименту для сцени зі сходами:

Для 3-х панорамних зображень алгоритм дав вироджений результат (порожню точкову хмару).

Для 8, 15 і 29 зображень отримали point cloud, кожен різної щільності і ступеня зашумленості:

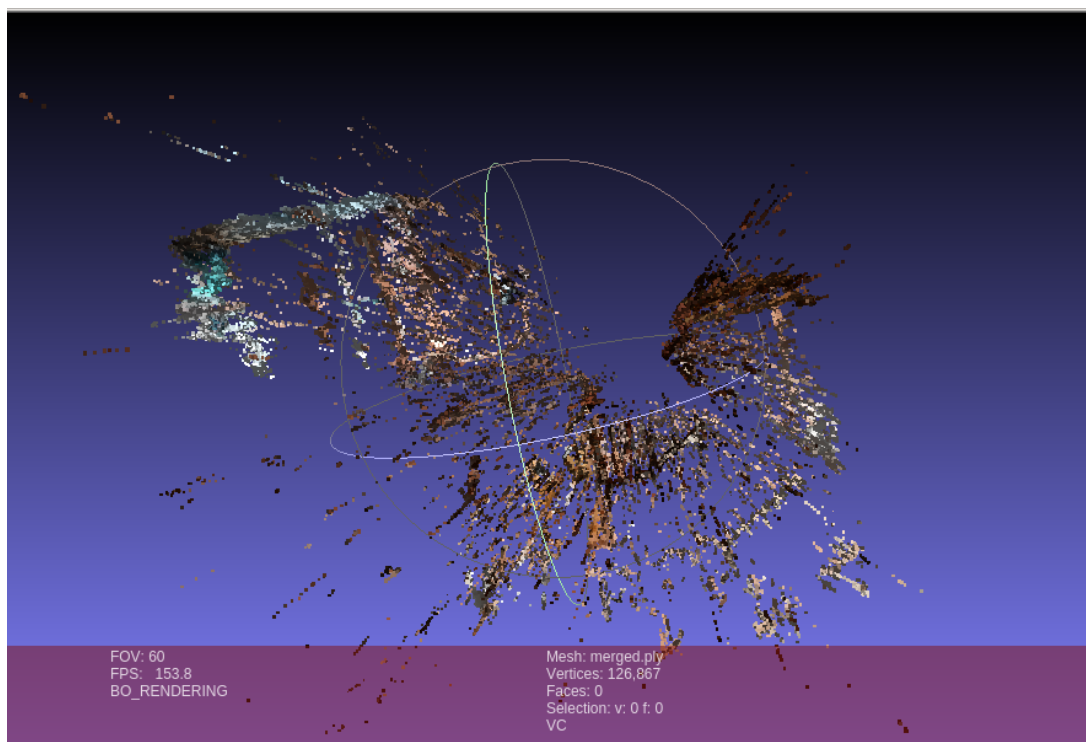


Рис. 3.4 — Точкова хмара для сцени зі сходами (8 вхідних зображень)

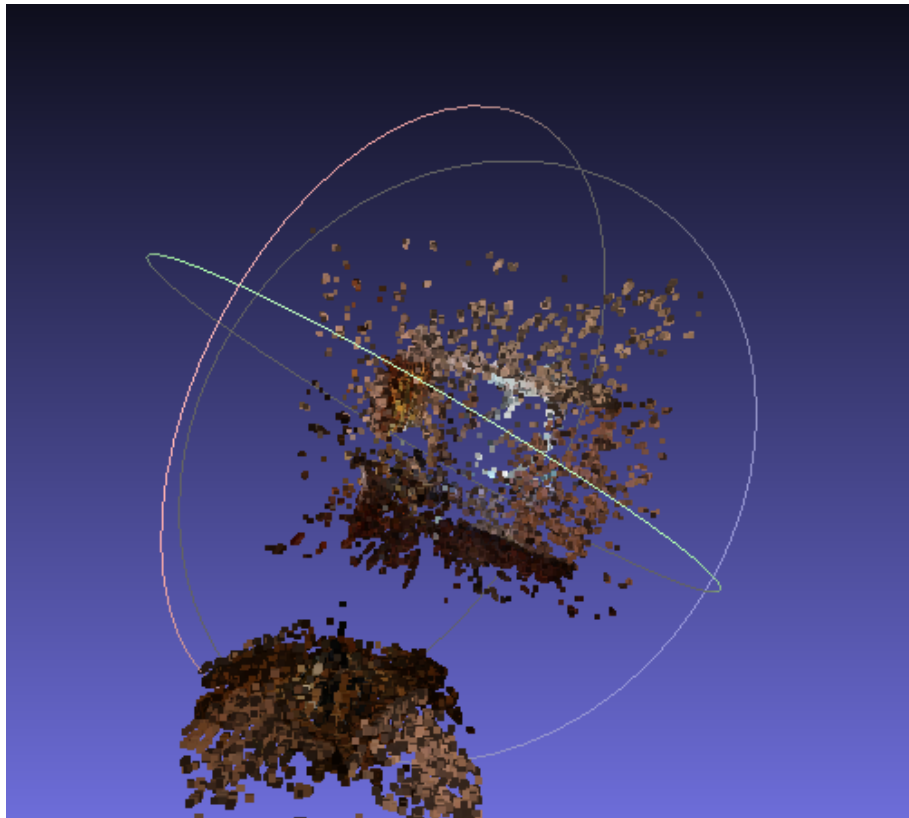


Рис. 3.5 Точкова хмара для сцени зі сходами (15 вихідних зображень)

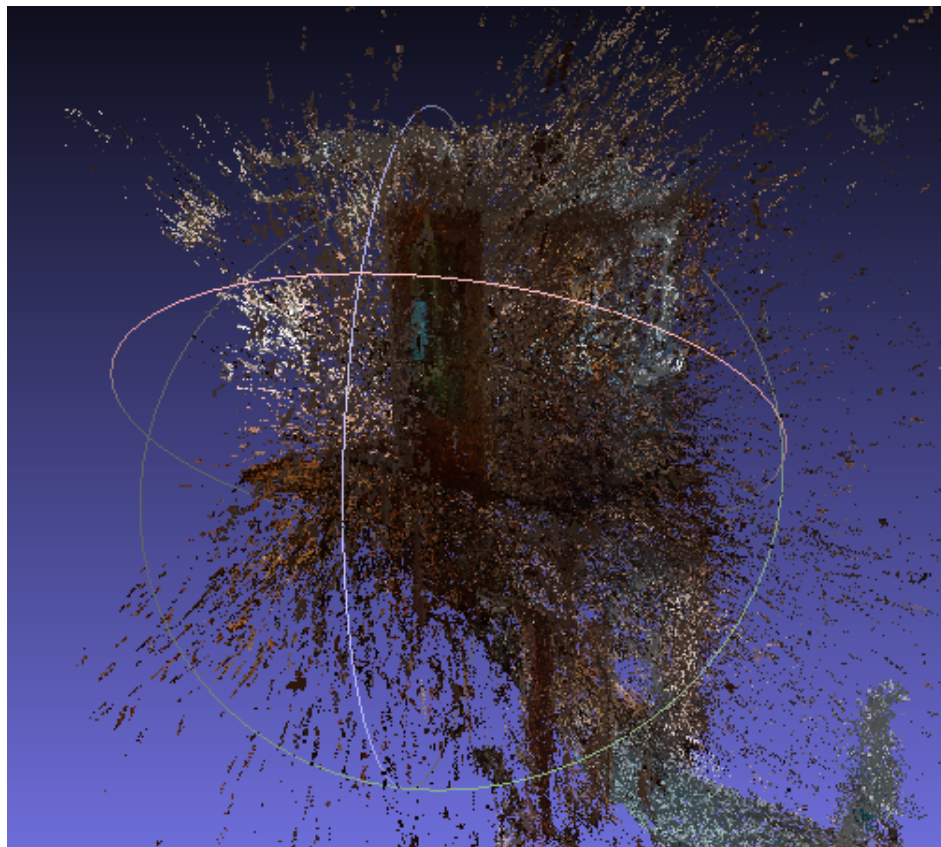


Рис. 3.6 — Точкова хмара, побудована на всіх зображеннях

Закономірне також спостереження — що із зростанням кількості даних, зростає і час роботи алгоритму:

	ladder8	ladder15	ladder29
extract_metadata	0.741949	1.367149	2.591339
detect_features	33.271515	69.591249	143.107825
match_features	21.466794	82.899558	250.414709
create_tracks	0.289441	0.583250	1.354297
reconstruct	1.360033	5.656738	25.653219
mesh	0.209770	0.461880	1.445752
undistort	4.377154	10.363047	38.272269
compute_depthmaps	96.743245	99.810437	690.264333
export_ply	0.027565	0.089989	0.286737
export_openmvs	0.026526	0.076827	0.308179

Таблиця 3.1 — час роботи алгоритму реконструкції.

Час роботи `detect_features` зростає лінійно при збільшенні набору даних, час потрібний на співставлення ознак (`match_features`) та на реконструкцію (`reconstruct`) — зростає квадратично.

3.3 Dense point cloud (щільна точкова хмара)

Наступний експеримент пов'язаний із щільною точковою хмарою та зміною її структури залежно від характеру початкових даних (зокрема, від кількості знімків змінюється щільність отриманої точкової хмари і повнота отриманої 3D-структури):

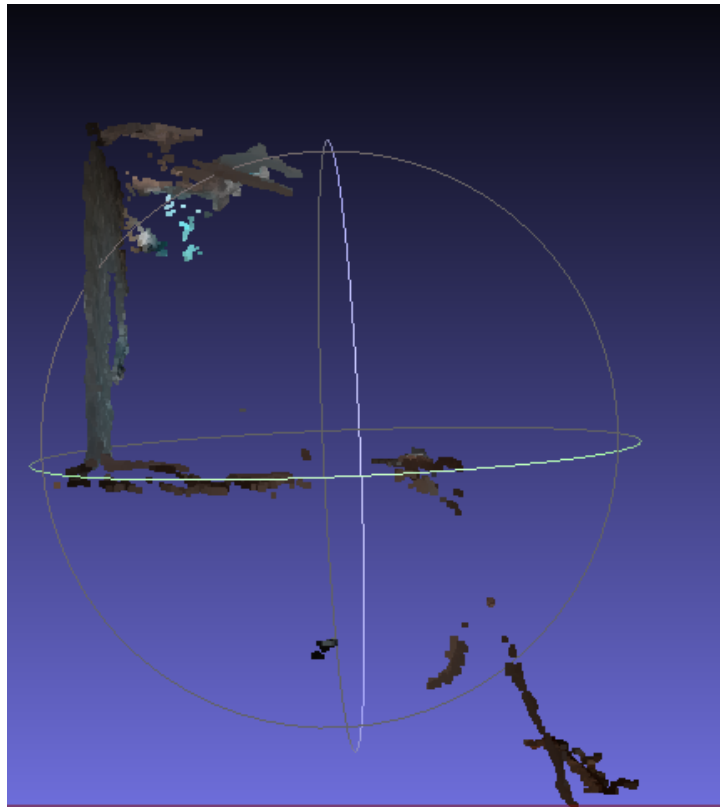


Рис. 3.7 Щільна точкова хмара (при 8 зображеннях)

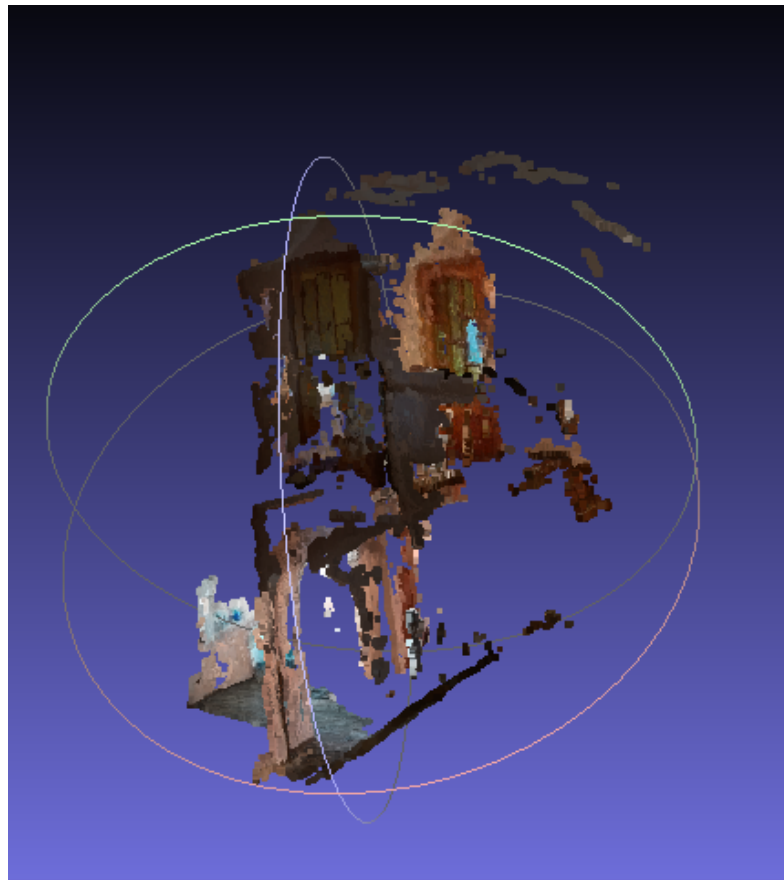


Рис. 3.8 — Щільна точкова хмара (29 зображень)

Основні спостереження у цьому експерименті:

- із зростанням кількості вхідних даних, квадратично зростає час роботи моделі;
- із зростанням кількості даних квадратично зростає роздільна здатність точкової хмари (наприклад, point cloud із зображень 3.7 та 3.8 вище налічують 15940 та 197200 точок. Співвідношення кількості вхідних даних - $29/8 = 3.62$, $197200/15940 = 12.37 \sim (3.62)^2$)

3.4 Текстурований меш

Фінальним продуктом 3D-реконструкції є кольоровий меш приміщення, на який вже нанесено текстуру. У цьому експерименті ми порівняли меші, отримані для різної кількості вхідних фото, та для двох різних сцен (житлова кімната та сцена зі сходами).

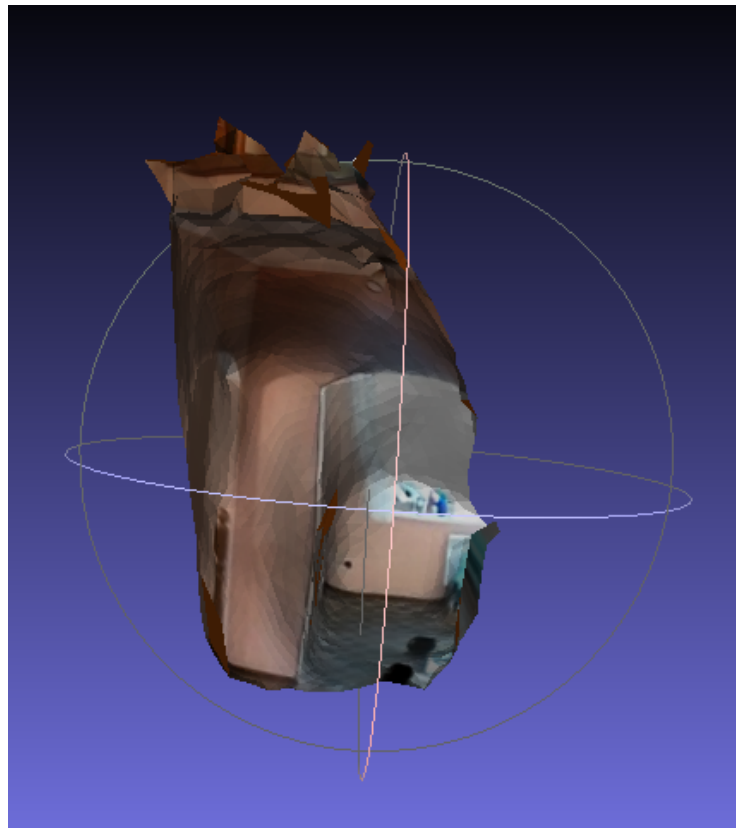


Рис. 3.9 Меш сцени зі сходами (8 вхідних зображень)

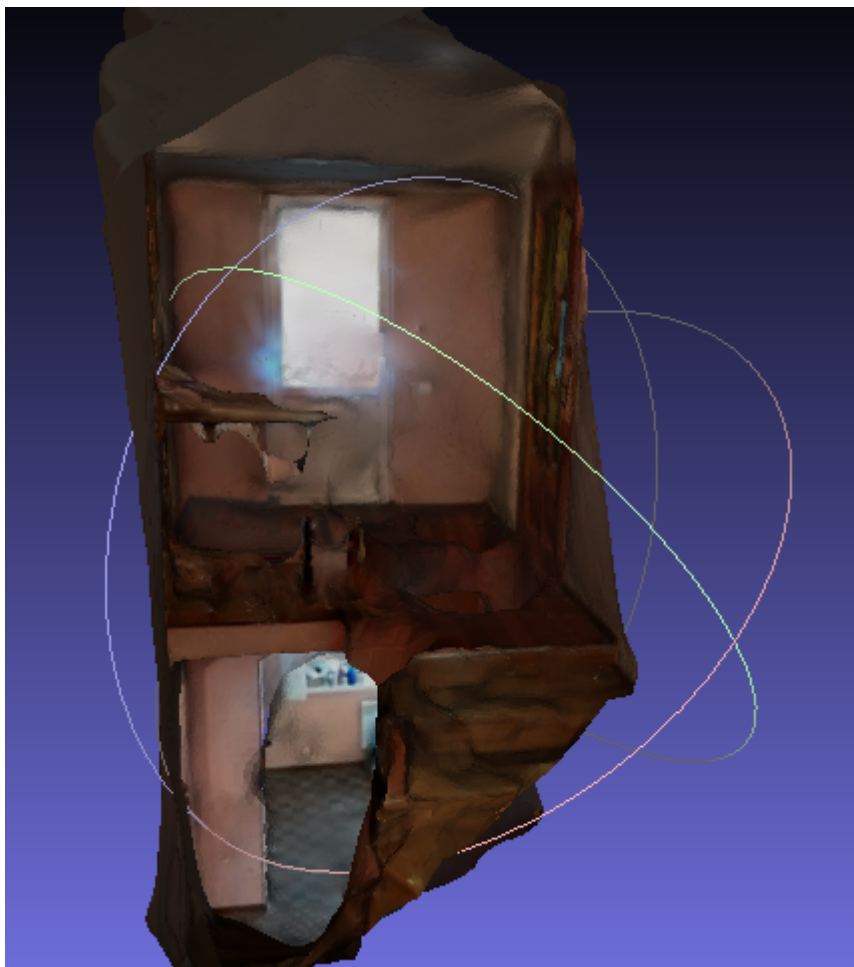


Рис. 3.10 Меш сцени зі сходами — 29 вхідних зображень

Основне спостереження — вхідні дані розміром у 8 панорам є недостатньо репрезентативними, щоб описати сцену зі сходами повністю. Водночас, при збільшенні кількості знімків спостерігаємо покращення якості поверхні, спостерігаємо більш повне представлення сцени.

Порівняємо також реконструкцію житлової кімнати, зроблену на різній кількості вхідних даних (набори кількістю 8 та 19 панорамних зображень):

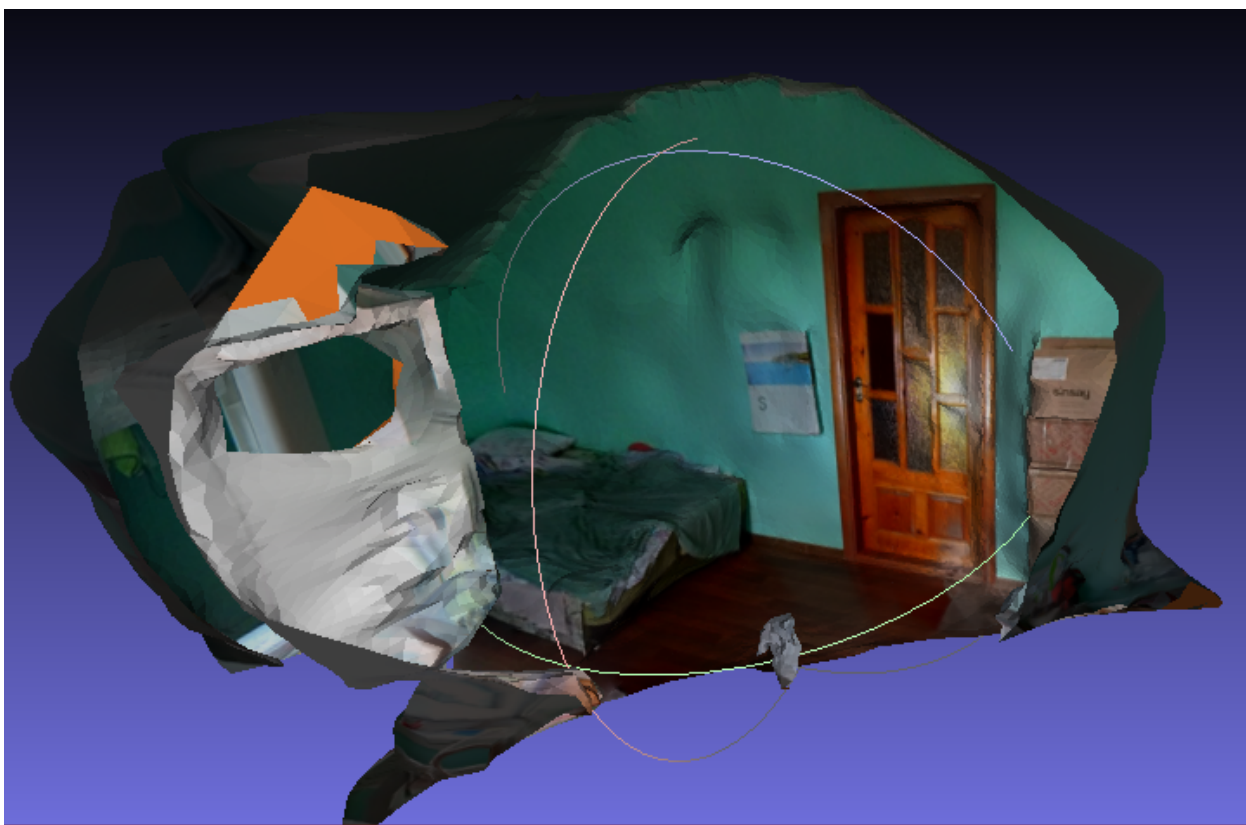


Рис. 3.11 Меш житлової кімнати (8 вхідних зображень)



Рис. 3.12 Меш житлової кімнати (19 вхідних зображень)

На фоні реконструкції житлової кімнати спостерігаємо аналогічну проблему — вхідних даних кількістю у 8 панорам недостатньо, щоб репрезентативно

описати приміщення. Тому на рисунку 3.11 ми бачимо хоча і якісний, але неповний зліпок приміщення. На рисунку 3.12 бачимо повну, деталізовану модель кімнати, отриману із 19 вхідних зображень панорам.

3.5 Порівняння реконструкції Пуассона та Делоне

В експерименті 3.5 взяли до уваги етап алгоритму, на якому відбувається перехід від щільної точкової хмари до текстурованого мешу. У розділі 2 описані два способи це зробити — триангуляція Пуассона та Делоне, дві різні процедури ущільнення точкової хмари. Тепер ми порівняємо результати обох триангуляцій на прикладі двох приміщень:



Рис. 3.13 Порівняння реконструкцій Пуассона і Делоне



Рис. 3.14 Порівняння реконструкцій Пуассона і Делоне (2)

На рисунках 3.13 і 3.14 можемо бачити відновлені поверхні кімнати під різними ракурсами. Реконструкція Делоне показала себе краще, великі поверхні вийшли більш гладкими, та трохи вища якість текстури.

Порівняємо також на прикладі сцени зі сходами:



Рис. 3.15 Порівняння реконструкцій Пуассона та Делоне (3)



Рис. 3.16 Порівняння реконструкцій Пуассона та Делоне (4)

На рисунках 3.15 та 3.16 можемо бачити приклади реконструкцій Пуассона та Делоне під різними ракурсами вже для більш складної сцени — сцени зі сходами. Спостерігається аналогічна проблема — нерівні стіни, опуклості на поверхнях у реконструкції зліва, в той час як справа поверхні більш гладкі.

Висновки до розділу 3

У розділі 3 описано два набори вхідних даних що зібрані для дослідження; описано експерименти, що показують вплив кількості та якості вхідних даних на проміжні результати та фінальний результат реконструкції. Зокрема:

- описано набір зібраних даних щодо двох приміщень різного плану (звичайна кімната, сцена зі сходами);
- досліджено якість зашумленої хмари точок;
- досліджено результат її покращення процедурою ущільнення;
- досліджено характер мешу кімнати в залежності від кількості вхідних фото.
- порівняно реконструкцію Пуассона та реконструкцію Делоне — по багатьом критеріям реконструкція Делоне показала себе краще саме у контексті нашої задачі.
- досліджено, як міняється час виконання алгоритму реконструкції, в залежності від кількості даних у наборі (експерименти показали, що час реконструкції зростає квадратично — зокрема за рахунок процедур `feature matching`, `depth_map_calculation`)

ВИСНОВКИ

В ході дослідження мною було виконано:

- Зібрано різні приклади вхідних даних, включаючи фото із датасетів ScanNet, S3DIS, також використовувались фото сферичних панорам, зроблені власноруч;
- Досліджено підхід Structure-from-Motion. Детально описано кожную складову частину, алгоритми що використовуються та їх параметри
- Досліджено частину Multi-view-stereo, розглянуто можливі способи покращення якості 3D-моделі (denoising/densification/meshing);
- Розроблено програмний продукт для обробки та візуалізації результатів
- Результати дослідження можуть бути корисним доробком у сфері AR/VR, та для платформ що займаються нерухомістю (OLX, dom.ria, LUN).

ПЕРЕЛІК ЛІТЕРАТУРНИХ ПОСИЛАНЬ

1. ScanNet: Richly-annotated 3D Reconstructions of Indoor Scenes // Dai Angela, Chang Angel X., Savva Manolis, Halber Maciej, Funkhouser Thomas, and Nießner Matthias // Proc. Computer Vision and Pattern Recognition (CVPR), IEEE. — 2017.
2. Bleyer Michael, PatchMatch Support mode: Rhemann Stereo Windows - // Christoph, Stereo Matching RotherCarsten. WithSlanted BMVC. — 2011. — January.
3. Schönberger Johannes Lutz, Frahm Jan-Michael. Structure-from-Motion Revisited // Conference on Computer Vision and Pattern Recognition (CVPR). — 2016.
4. iMAP: Implicit Mapping and Positioning in Real-Time / Sucar Edgar, Liu Shikun, Ortiz Joseph, and Davison Andrew // Proceedings of the International Conference on Computer Vision (ICCV). — 2021.
5. Zhou Qian-Yi, Park Jaesik, Koltun Vladlen. Open3D: A Modern Library for 3D Data Processing. — 2018.
6. Rublee, E., Rabaud, V., Konolige, K., Bradski, G. (2011). ORB: an efficient alternative to SIFT or SURF.
7. Aashutosh, P. (2021). Understanding the Dense Point Cloud Generation: DensifyPointCloud in OpenMVS
8. Huang, W., Zhou, T., Zhang, H., & Xia, H. (2017). Dual-fisheye Lens Stitching for 360-degree Imaging.
9. Shree Nayar, (2018). FPCV - First Principles of Computer Vision
10. Carlo Tomasi and Takeo Kanade. (November 1992). "Shape and motion from image streams under orthography: a factorization method". *International Journal of Computer Vision*. 9 (2): 137–154.
11. Lowe, D. G. (1999). Object recognition from local scale-invariant features. Proceedings of the International Conference on Computer Vision, 2, 1150-1157.

12. Bay, H., Ess, A., Tuytelaars, T., & Van Gool, L. (2008). SURF: Speeded Up Robust Features. *Computer Vision and Image Understanding*, 110(3), 346-359.
13. Schönberger, J. L., & Frahm, J. M. (2016). Structure-from-motion revisited. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 4104-4113).
14. Kazhdan, M., Bolitho, M., & Hoppe, H. (2006). Poisson surface reconstruction. *Proceedings of the Fourth Eurographics Symposium on Geometry Processing*, 61-70
15. Amenta, N., Bern, M., & Eppstein, D. (1998). The crust and the beta-skeleton: combinatorial curve reconstruction. *Graphical Models and Image Processing*, 60(2), 125-135.