Одеський Національний Університет імені І. І. Мечникова
Факультет Математики, Фізики та Інформаційних Технологій
Кафедра Оптимального Керування і Економічної Кібернетики

Кваліфікаційна робота

на здобуття ступеня вищої освіти «бакалавр»

«Алгоритми SLAM для автономних систем руху»

«SLAM algorithms for autonomous motion systems»

Виконав: здобувач денної форми навчання
спеціальності 113 Прикладна математика
Освітня програма «Прикладна математика»
Кузьмичов Олександр Євгенович
Керівник: канд. тех. наук, доц. Мороз В. В.
Рецензент: ст. викладач Платонов В. В.

Рекомендовано до захисту:

Протокол засідання кафедри

№ ___ від _____ 2025 р.

Завідувач кафедри

_____ _____

Захищено на засіданні ЕК № _____

Протокол № ___ від _____ 2025 р.

Оцінка _____ / _____ / _____

Голова ЕК

_____ _____

Одеса — 2025 р.

ЗМІСТ

Умовні позначення 3

Вступ 4

1 Аналіз існуючих алгоритмів 7

2 Моделі та методи SLAM 13
2.1 Означення SLAM 13
2.1.1 Математична модель 13
2.2 Класифікація методів SLAM 15
2.3 Класичні методи 17
2.3.1 Непрямі методи 18
2.3.2 Прямі методи 23
2.4 Методи машинного навчання 30
2.4.1 Непрямі методи 31
2.4.2 Прямі методи 38
2.5 Висновки 44

3 Практична реалізація SLAM 46
3.1 Класичні методи 46
3.2 Методи машинного навчання 51

4 Реконструкція 62

Висновки 65

Список літератури 66

Додаток А 69

Додаток Б 71

Додаток В 76

УМОВНІ ПОЗНАЧЕННЯ

Жирні малі літери ($\mathbf{x}$) позначають вектори для тверджень та формулювань. Жирні великі літери ($\mathbf{R}$) позначають матриці. Скаляри представлені світлими малими літерами (c). Функції та зображення представлені великими світлими літерами (I). Визначимо зображення I, яке містить набір пікселів. Для кожного пікселя $\mathbf{q}$ на зображенні припущено, що існує значення глибини d, яке дозволяє проектувати його відповідні 3D-координати $\mathbf{x} = (x, y, z)^\mathsf{T}$. Таким чином, пози камери представлені як матриці перетворення $\mathbf{T}_i \in SE(3)$, що перетворюють точку з реального кадру в кадр камери. $\mathbf{R}$ представляє матриці обертання, тоді як Π та Π^{-1} — це функції проекції та зворотної проекції. d^* представляє обернені значення глибини, тому $\mathbf{D}$ та $\mathbf{D}^*$ відповідають картам глибини та оберненим картам глибини.

ВСТУП

Питання високоякісного аналізу та опрацювання машиною візуальної інформації стоїть гостро вже майже шість десятиліть. На даний момент, за допомогою численних досліджень, комп'ютерний "зір"реалізований десятками методів, кожний з яких підходить під свої певні умови та частина з яких є розвитком попередніх. Сучасні завдання комп'ютерного бачення дедалі частіше потребують не лише локалізації камери, а й побудови точного тривимірного відображення навколишнього середовища. Методи SLAM (Simultaneous Localization and Mapping) стали ключовими у сфері автономної навігації, розширеної реальності та цифрової реконструкції. Водночас, зростання кількості доступних алгоритмів створює проблему вибору — не всі методи є універсальними, і ефективність кожного залежить від конкретного типу даних. Це особливо важливо для роботи з фотографічними датасетами, які можуть мати значні варіації в умовах зйомки. Отже, задача обґрунтованого підбору SLAM-методу відповідно до властивостей даних є **актуальною**.

Об'єктом дослідження є методи одночасної локалізації та побудови карти (SLAM).

Предметом дослідження виступають алгоритми SLAM для реконструкції сцени на основі фотографічного датасету.

Мета роботи полягає в аналізі ефективності методів SLAM та їх практичного застосування для трьохвимірної реконструкції.

Задачі дослідження:

1) провести класифікацію методів SLAM за архітектурними та функціональними ознаками;
2) визначити ключові характеристики, які впливають на ефективність SLAM в умовах статичного фотографічного датасету;
3) оцінити обчислювальну складність методів, стійкість до шуму та освітлення;
4) реалізувати та виконати порівняльний аналіз реконструкції за допомогою обраних методів, проаналізувати отримані результати.

Методи дослідження

У роботі розглядаються та застосовані методи 3D-проектування на основі SLAM, які поділяються на класичні методи та методи машинного навчання, які, в свою чергу, діляться на непрямі та прямі.

Результати роботи можуть бути застосовані у:

- вибору придатного методу аналізу сцени для робототехнічних систем;
- формуванні 3D-моделей для подальшого опрацювання;
- доповненій реальності на базі статичних зображень;

- підготовці моделей для симуляцій і візуалізації.

Робота складається зі вступу, чотирьох розділів, висновку і додатків.

РОЗДІЛ 1

АНАЛІЗ ІСНУЮЧИХ АЛГОРИТМІВ

Монокулярна 3D-реконструкція є погано обумовленою задачею, яку можна розв'язати шляхом поєднання різних методів та алгоритмів з таких дисциплін, як комп'ютерний зір, робототехніка та машинне навчання, наразі лише кілька робіт можуть бути пов'язані з цим оглядом. В даній роботі акцентується увага на наданні відповідної таксономії та описі кожного з найважливіших алгоритмів, щоб дати відповідний огляд, який може допомогти правильно обрати найбільш підходящий метод для проектів та досліджень.

Був проведений комплексний та послідовний огляд доступних 26 алгоритмів (включаючи машинне навчання), пропрацьована розширена класифікація, яка враховує всі монокулярні чисто візуальні системи, що сприяють проблемі монокулярної 3D-реконструкції.

В даній роботі розглянуто:

1) Таксономія, розроблена для охоплення всіх можливих існуючих підходів, побудована з урахуванням трьох класифікацій та всіх можливих комбінацій;
2) 26 монокулярних алгоритмів SLAM, що включають 10 класичних монокулярних та 16 методів, що інтегрують машинне навчання. Кожен розглянутий метод включає свою алгоритмічну систематизацію, математичні принципи для тих алгоритмів, які внесли інновації у свої формулюван-

ня, особливо щодо оцінки або оптимізації карти глибини, оскільки ця робота зосереджена на 3D-реконструкції;

3) 11 критеріїв для прийняття рішень щодо впровадження, вибору або проектування системи 3D-реконструкції, з яких дев'ять застосовні до класичних систем, а два додаткові критерії застосовні лише до підходів машинного навчання. Ця інформація була зібрана для кожного алгоритму, критерії наступні: щільність карти (щільна чи розріджена), використані пікселі (метод вилучення інформації про пікселі), метод оцінки (метод оцінки карти глибини), глобальна оптимізація, релокалізація, замикання циклу (чи включає алгоритм кроки оптимізації, релокалізації чи замикання циклу), архітектура CNN (загальновідома архітектура CNN, що використовується) та основні завдання, для яких використовувалася CNN.

Мета отримання 3D-геометричного зображення сцени – це складне завдання, яке можна виконати за допомогою камер-сенсорів.

У минулому деякі сучасні системи створювалися з використанням складних масивів камер та установок освітлення, переважно для застосування всередині приміщень. Однак сьогодні різні пристрої захоплення варіюються від дорогих багатоканальних та стереосистем до дешевих монокулярних датчиків. Далі наведено короткий огляд кількох способів введення, що використовуються для 3D-реконструкції.

1. Стерео установки.

Багатоканальні установки складаються з масиву або набору парних стереокамер, розподілених у конфігурації, яка дозволяє одночасно знімати зображення одного й того ж об'єкта. Однак ці камери, як правило, дорожчі за монокулярні сенсори та потребують більших зусиль для калібрування. Крім того, ці сенсори повинні отримувати зображення з однаковим інтервалом часу, чого можна досягти шляхом синхронізації витримки за допомогою зовнішнього сигналу запуску. Як правило, підтримка каліброваної постійної базової лінії між двома камерами вимагає набагато більше зусиль, ніж у випадку з монокуляром, і стереосистеми деградують до монокулярних, коли базова лінія набагато менша за відстань від сцени до камери. Отже, вони обмежені роботою в невеликих приміщеннях та приміщеннях.

2. Всеспрямовані камери.

   Високоякісний вибір, завдяки їх широкому полю зору (FOV). Надають більше інформації, ніж звичайні камери, а особливості, які можна знайти на зображеннях, зберігаються протягом тривалішого часу, що допомагає отримувати чіткі 3D-моделі пейзажів. Однак всеспрямовані камери є дорогими, вимагають значних зусиль для налаштування та несумісні з мобільними пристроями. Крім того, деякі з цих пристроїв поступово сканують сцену за допомогою механічного обертання, тому ці пристрої призначені для роботи в статичних умовах і можуть не працювати в динамічних середовищах. Однією з основних проблем, які мають ці пристрої для завдання 3D-реконструкції, є виникнення

спотворень у зображенні, що виникають внаслідок рівнопрямокутного представлення, що виникає, коли отримані сферичні пікселі, спроектовані на площину, значно спотворюються, що може спричинити помилки прогнозування глибини.

3. Монокулярний RGB-D.

   RGB-D-сенсори мають додатковий активний або пасивний датчик, який дозволяє системі отримувати вимірювання глибини середовища, пов'язаного з кожним пікселем зображення, в режимі реального часу. Такі вимірювання глибини допомагають вирішити неоднозначність глибини та масштабу монокулярної реконструкції, оскільки їх можна використовувати для оцінки геометрії середовища. Ці пристрої можна класифікувати як пасивні або активні. На відміну від стереосенсорів, пасивні RGB-D-камери зазвичай мають проектор замість другої камери, проектуючи візерунок на зображення, щоб знайти точки, що збігаються. RGB-D-камери, що використовують інфрачервоні проектори, відомі як активні. Слід зазначити, що активні RGB-D-камери можуть видавати помилкові вимірювання під сонячним світлом через ІЧ-випромінювання сонячного світла, яке перевантажує проектор. З цієї причини деякі RGB-D-камери мають комбінацію активних та пасивних датчиків, які активуються незалежно від того, чи знаходяться вони під сонячним світлом, чи ні, що може значно збільшити вартість цього типу пристроїв. Ще одним поширеним обмеженням цих світлових пристроїв є те, що вони не можуть

реконструювати об'єкти, менші за проектований візерунок.

4. Монокулярний RGB.

   RGB-камери фіксують інтенсивність прийнятого світла у трьох каналах: червоному, зеленому та синьому. RGB-пристрої призначені для роботи в різних конфігураціях, починаючи від CCD-сенсорів, що отримують кожен сигнал в окремому датчику, і закінчуючи датчиками на основі шаблонів Байєра, де кольорові фільтри чергуються перед одним датчиком. Монокулярні камери відомі тим, що зменшують вплив помилок калібрування. Вони найбільш розповсюджені, мають низьку вартість, прості в розгортанні та доступні в більшості портативних пристроїв, що вважається значним стимулом, який утримує увагу дослідників, які зазвичай віддають перевагу цьому типу вхідних даних для виконання завдань реконструкції, пов'язаних із SLAM. Тим не менш, ці вхідні дані пов'язані з погано проблемою, оскільки монокулярний зір страждає від невизначеності масштабу, а методи, що використовуються для досягнення мети 3D-реконструкції, зазвичай вимагають багато обчислювальних ресурсів.

   Хоча монокулярні RGB-камери мають деякі важливі проблеми через свою безсенсорну природу, що не дозволяє їм забезпечувати прямі вимірювання глибини, вони пропонують найпривабливіший набір переваг, особливо для невеликих робототехнічних застосувань. Монокулярні RGB-сенсори мають найнижчу ціну серед усіх доступних типів камер, прості в розгортанні та сумісні майже з усіма існую-

чими портативними пристроями та процесорами, такими як одноплатні комп'ютери (SBC) та програмовані польовими логічними матрицями (FPGA), що робить цей спосіб введення особливо привабливим для дослідників. Саме тому це дослідження було зосереджено на монокулярному способі введення RGB [3].

РОЗДІЛ 2

МОДЕЛІ ТА МЕТОДИ SLAM

2.1 Означення SLAM

SLAM — це тип алгоритмів, які дозволяють пристрою створювати карту свого оточення та точно орієнтуватися на цій карті в режимі реального часу [1].
Це важлива можливість для автономних роботів і самокерованих транспортних засобів, яким необхідно безпечно й ефективно орієнтуватися в навколишньому середовищі, але вони не завжди можуть мати доступ до GPS або інших зовнішніх сигналів локалізації.

2.1.1 Математична модель

Проблему SLAM можна сформулювати як задачу ймовірнісного оцінювання, яка прагне визначити траєкторію пристрою X і карту M у момент часу t на основі послідовності спостережень датчиків Z і керуючих вхідних даних U[2]:

$$P(X, M|Z, U)$$

$$U = u_{1:t} = \{u_1, u_2, u_3, ..., u_t\},$$

де U являє собою керування роботом в момент часу t.

$$Z = z_{1:t} = \{z_1, z_2, z_3, ..., z_t\},$$

де Z являє собою інформацію про навколишнє середовище, що оглядається роботом в момент часу t.

$$M,$$

$$X = x_{1:t} = \{x_1, x_2, x_3, ..., x_t\},$$

де M являє собою побудовану карту, а X - отримане розташування робота в момент часу t.

Таким чином, для розрахунку траєкторії робота, можна все зобразити в такому вигляді:

$$p(x_{1:t}, m | z_{1:t}, u_{1:t})$$

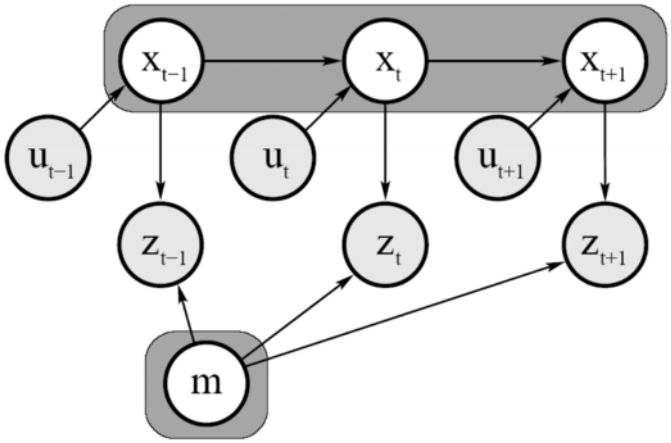

Рис 2.1. Графічна модель ймовірнісного оцінювання знаходження траєкторії автономної системи та мапи в конкретний момент часу, що залежать від керування та спостережених даних.

2.2 Класифікація методів SLAM

Враховуючи, що 3D-реконструкція є основною проблемою для методів монокулярного SLAM, класифікація підходів на основі ознак, зовнішнього вигляду або гібридних підходів не підходить для охоплення такого широкого спектру рішень для погано обумовленої монокулярної задачі. Таким чином, розглянемо такі дві основні класифікації: пряма та непряма, щільна та розріджена.

Отже, монокулярні SLAM поділяються на дві основні категорії, залежно від кількості ознак, що використовуються для завдань, необхідних для 3D-реконструкції. Розріджена та щільна класифікація є способом їх категоризації. Крім того, інша класифікація залежить від необхідності виконання попередньої обробки перед отриманням фактичних параметрів та вимірювань. Непрямі методи використовують цей крок попередньої обробки, який генерує проміжне представлення шумних вимірювань, що потребують оптимізації перед оцінкою геометрії та руху камери. На відміну від цього, прямі формулювання використовують інформацію про пікселі безпосередньо. Крім того, завдяки значним удосконаленням у машинному навчанні та вражаючим результатам, яких досягла ця нова категорія, для таксономії було введено новий компонент. Таким чином, ми встановили три класифікації, що визначають відповідну таксономію для задачі монокулярної 3D-реконструкції: пряма проти непрямої, щільна проти розрідженої, а також класичне проти машинного навчання.

- **Прямі проти непрямих (Direct vs. Indirect).** Прямі методи не використовують кроки попередньої обробки, такі як вилучення ознак або оптичного потоку, тоді як непрямі методи використовують ці кроки попередньої обробки;
- **Щільні проти розріджених (Dense vs. Sparse).** Щільні методи стосуються методів, що використовують всю або більшу частину інформації про пікселі зображення, тоді як розріджені стосуються методів, що використовують лише підмножину вибраної інформації про пікселі;
- **Класичні проти машинного навчання (Classic vs. Machine Learning).** Класичні методи, які також називають геометричними методами, базуються на геометрії, одометрії або ймовірнісних пропозиціях для запуску всього свого конвеєра. Вони не потребують жодних кроків навчання, тому їх необхідно належним чином налаштувати та відкалібрувати для функціонування. З іншого боку, методи, засновані на навчанні, продемонстрували свою потужність у виконанні завдань низького рівня (вилучення ознак, оцінка глибини, оцінка пози) та завдань високого рівня (класифікація та семантична сегментація). З цієї причини багато дослідників зосередилися на розробці нейронних мереж для прогнозування пози, прогнозування глибини, вилучення ознак та семантичної сегментації (серед інших), поєднуючи їх з класичними фреймворками для підвищення їхньої точності, можливостей узагальнення, стійкості, можливостей розуміння сцени тощо [3].

2.3 Класичні методи

Класичні методи монокулярного SLAM, також відомі як геометричний підхід, є першою категорією, що розглядається в цьому дослідженні. Ці методи називаються геометричними підходами, зазвичай спираючись на геометричну інформацію для виконання 3D-реконструкції та запуску всіх своїх конвеєрів SLAM. Однак, оскільки відновлення геометрії сцени як реконструкції є погано обумовленою задачею, велика різноманітність пропозицій також може поєднувати традиційні геометричні методи з іншими, включаючи ймовірнісні, оптимізаційні, комп'ютерний зір, евристичні методи тощо. Тим не менш, у цій класифікації не включено формулювання, що використовують машинне навчання, що буде обговорено в наступному розділі. Як зазначали Енгель та ін., геометрію сцени можна оцінити за допомогою ймовірнісної моделі, яка використовує шумні вимірювання $\mathbf{Y}$ із зображень, що генерують оцінку $\mathbf{X}$ для 3D-моделі та власного руху. Таким чином, задачу 3D-реконструкції можна вирішити за допомогою класичних підходів з двох парадигм: непрямої та прямої. У непрямому підході вимірювання камери попередньо обробляються, що дає проміжне представлення, яке вирішує частину проблеми. Ці проміжні отримані значення потім використовуються як зашумлені вхідні дані для оцінки $\mathbf{X}$. У другій групі, яка називається прямою, системи пропускають крок попередньої обробки та безпосередньо використовують вимірювання, отримані зі спостережень, як зашумлені вхідні дані для оцінки $\mathbf{X}$ в ймовірнісній моделі. Ця класифікація також

визначає, як виконується оптимізація, тому прямі методи оптимізують фотометричні похибки (різницю інтенсивності пікселів), оскільки прямі методи отримують фотометричні вимірювання. На противагу цьому, непрямі методи оптимізують геометричні похибки, оскільки попередня обробка обчислює геометричні значення [3].

2.3.1 Непрямі методи

Як згадувалося раніше, непрямі методи використовують кроки попередньої обробки для вилучення інформації про вхідну послідовність зображень у вигляді візуальних ознак, дескрипторів або оптичного потоку. Хоча, водночас, класичні непрямі системи можуть бути розроблені для відновлення розрідженого або щільного середовища 3D-реконструкції. Одна з найважливіших розробок SLAM належить до цієї категорії, система ORB-SLAM та її наступники, яка має один з найбільш вражаючих показників цитувань.

Розріджені методи

Ці методи SLAM використовують підмножину інформації про пікселі, щоб виконати свої конститутивні процеси, тому вихід також є підмножиною очікуваної реконструкції пейзажу, що в деяких застосуваннях, таких як робототехніка, достатньо для виконання таких завдань, як навігація роботів або навіть розпізнавання місць. Тому ці методи є непрямими через їхні процеси попередньої обробки, де нові змінні, такі як ознаки та

їх положення, обчислюються шляхом підстановки інформації про пікселі, тому решта процесів виконуються з використанням цих нових значень.

MonoSLAM (2007). [3] Перша повноцінна система монокулярного SLAM, представлена Девісоном у 2007 році. Працює на основі розширеного фільтра Калмана (EKF), який одночасно оцінює положення камери та 3D-карту довколишніх орієнтирів у режимі реального часу. Використовує активне відстеження високоякісних ознак (Shi–Tomasi), оцінку глибини через гіпотези на напівнескінченній прямій та поступове уточнення до гаусівського розподілу. Система ініціалізується за допомогою відомої цілі для подолання неоднозначності масштабу. Основним обмеженням є зростання складності з розміром сцени та розрідженість карти.

PTAM (2009). [3] [4] Запропонована Кляйном і Мюрреєм у 2009 році система Parallel Tracking and Mapping (PTAM) стала першою, що розділяє відстеження та картографування на паралельні потоки. Потік відстеження оцінює позу камери, шукає відповідності ознак та оновлює положення, тоді як потік картографування виконує ініціалізацію, уточнення та розширення карти через пакетну оптимізацію методом Bundle Adjustment.

ORB-SLAM (2015). [3] [5] Запропонована Мур-Аталем система ORB-SLAM базується на багатомасштабних ознаках ORB, що застосовуються для відстеження, картографування, релокалізації та замикання циклів. Метод є непрямим і виконує

попередню обробку зображення з вилученням ознак ORB у ключових точках. Як і PTAM, використовує Bundle Adjustment, але орієнтується на підмножину ключових кадрів з хорошим паралаксом і кількістю збігів, що робить глобальну оптимізацію в реальному часі можливою. ORB-SLAM демонструє високу точність при знижених обчислювальних витратах.

У цьому методі оптимізація коригування зв'язків виконується для 3D-локацій $\mathbf{x}_{\omega,j} = (x_{\omega,j}, y_{\omega,j}, z_{\omega,j})^\mathsf{T}$, а пози ключового кадру $\mathbf{T}_{i\omega}$ оптимізуються шляхом мінімізації помилки повторної проєкції для точок $\mathbf{x}_{i,j}$, тому помилка для точки j у ключовому кадрі i становить:

$$\mathbf{e}_{i,j} = \mathbf{x}_{i,j} - \pi_i(\mathbf{T}_{i\omega}, \mathbf{x}_{\omega,j}), \tag{2.1}$$

$$\pi_i(\mathbf{T}_{i\omega}, \mathbf{x}_{\omega,j}) = \begin{bmatrix} f_{i,u}\dfrac{x_{i,j}}{z_{i,j}} + c_{i,u} \\ f_{i,v}\dfrac{y_{i,j}}{z_{i,j}} + c_{i,v} \end{bmatrix}, \tag{2.2}$$

$$\begin{bmatrix} x_{i,j} \\ y_{i,j} \\ z_{i,j} \end{bmatrix} = \mathbf{R}_{i\omega}\mathbf{x}_{\omega,j} + \mathbf{t}_{i\omega}, \tag{2.3}$$

де π_i — функція проєкції, $\mathbf{R}_{i\omega}$ та $\mathbf{t}_{i\omega}$ — компоненти обертання та зсуву трансформації $\mathbf{T}_{i\omega}$. Параметри $(f_{i,u}, f_{i,v})$ та $(c_{i,u}, c_{i,v})$ — це фокусна відстань та головна точка, пов'язані з камерою i.

Таким чином, функція витрат, яку потрібно мінімізувати, має вигляд:

$$C = \sum_{i,j} \mathcal{H}_h \left(\mathbf{e}_{i,j}^{\mathsf{T}} \, \mathbf{\Omega}_{i,j}^{-1} \mathbf{e}_{i,j} \right), \tag{2.4}$$

де $\mathcal{H}_h$ — робастна функція витрат Губера, а $\mathbf{\Omega}_{i,j} = \sigma_{i,j}^2 \mathbf{I}_{2\times 2}$ — коваріаційна матриця, пов'язана зі шкалою для кожної виявленої ключової точки.

ORB-SLAM2 (2017). [3] [6] Продовжуючи розробку ORB-SLAM, Мур-Артал і Тардос представили ORB-SLAM2, який підтримує монокулярні, стерео та RGB-D сенсори. Стереопараметри дозволяють точнішу оцінку глибини через триангуляцію або глибину з RGB-D. Основною новацією стало додавання окремого потоку для повного Bundle Adjustment після замикання циклу, що виконується паралельно з основними процесами. Система зберігає DBoW2 для релокалізації та використовує граф ко-видимості, забезпечуючи масштабованість. Для оптимізації застосовується алгоритм Левенберга-Марквардта через g2o, що дозволяє налаштовувати позу камери, локальні ключові кадри та карту в різних потоках. У результаті, ORB-SLAM2 підвищує точність реконструкції та стабільність у різних конфігураціях сенсорів.

ORB-SLAM2 також інтегрує режим локалізації, де потоки локального відображення та замикання циклу деактивуються для відомих областей, якщо немає значних варіацій у пейзажі, що дозволяє довгострокову та легку локалізацію та функціональність.

Цей метод порівнювали з багатьма існуючими стерео, RGB-D та монокулярними системами того часу, перевершуючи більшість із них у наборах даних EuRoC, TUM та KITTI, доводячи свою функціональність у великій різноманітності середовищ.

ORB-SLAM3 (2021). [3] [7] ORB-SLAM3, розроблений Кампосом та ін., розширює функціональність своїх попередників, підтримуючи візуальні, візуально-інерційні та багатокартові сценарії для монокулярних, стерео- та RGB-D камер, включно з моделями «риб'яче око». Метод забезпечує чотири рівні асоціації даних: коротко-, середньо-, довгострокову та міжкартову. Основні новації включають MAP-оцінювання, вдосконалене розпізнавання місць, абстрактне представлення камери та систему ATLAS для багатокартового SLAM. Система має три основні потоки: відстеження (локалізація і вставка ключових кадрів), локальне картографування (додавання/очищення точок та локальне BA) і об'єднання (виявлення циклів і глобальне BA). ORB-SLAM3 демонструє високу точність на наборах EuRoC і TUM-VI, хоча має обмеження в умовах слабкої текстури чи домінуючих обертань.

Щільні методи

Це формулювання переважно оцінює 3D-геометрію з регуляризованого поля оптичного потоку або разом з ним, таким чином поєднуючи геометричну похибку (відхилення від поля потоку) з отриманим геометричним апріорним (що пояснюється

гладкістю поля потоку), що є поширеним явищем. Як згадувалося вище, ця категорія монокулярних методів вимагає великої кількості вхідної інформації, використовуючи більшість значень пікселів, для виконання складових процесів. Наприклад, щільні монокулярні методи не вимагають вилучення підмножини ознак, оскільки вони працюють безпосередньо над усім вхідним даними. Таким чином, ці методи не вимагають дискретних ознак, але пов'язані зі значними обчислювальними витратами через більшу кількість даних, які будуть оброблені. Крім того, ця категорія методів також вважається непрямою, оскільки більшість залежить від інформації про оптичний потік, отриманої на етапі попередньої обробки.

Valgaerts et al. (2011). [3] Не є переважно SLAM-методом, доданий як приклад. Метод Валгартса запропонував варіаційний щільний підхід для спільної оцінки епіполярної геометрії та оптичного потоку, мінімізуючи єдину енергетичну функцію. У межах порівняльного дослідження щільних і розріджених методів було встановлено, що щільна оцінка переважає у випадках недостатньої локалізації ознак або вироджених конфігурацій.

2.3.2 Прямі методи

Прямі методи створюються для відновлення геометрії сцени та руху агента за допомогою прямої інформації про інтенсивність пікселів, тому, на відміну від непрямих методів, ці формулювання не потребують етапів попередньої обробки, та-

ких як вилучення ознак (що суттєво зменшує обсяг інформації, доступної для системи SLAM, заощаджуючи обчислювальні ресурси та скорочуючи час, хоча й обмежуючи кінцеву щільність 3D-реконструкції). На відміну від цього, прямий підхід працює з кожним значенням інтенсивності пікселя на зображенні або принаймні з більшістю з них, тому кінцева якість 3D-реконструкції зазвичай вища, ніж у непрямих підходів. Однак прямі формулювання спираються на припущення про сталість яскравості, яке встановлює, що яскравість над положенням об'єкта повинна бути однаковою під різними кутами. На жаль, це не завжди так, тому відомо, що ці формулювання не працюють у сценах, які відображають розмиття руху, рухомі об'єкти або неламбертові поверхні.

Цю категорію методів також можна розділити залежно від кінцевої щільності 3D-карти, тому існують щільні та розріджені формулювання. Три найрепрезентативніші монокулярні системи всіх часів належать до категорії DTAM, LSD-SLAM та DSO. Слід зазначити, що хоча DSO була представлена нещодавно, вона привернула увагу багатьох дослідників, отримавши вражаючі бали цитування.

Розріджені методи

Цей клас методів мінімізує фотометричну похибку без геометричних апріорних припущень чи попередньої обробки. Завдяки використанню попіксельної оберненої глибини прямі методи не потребують виокремлення точок, що забезпечує детальніше

геометричне представлення сцени, включаючи краї та слабкі варіації інтенсивності. Вони стійкі до низькотекстурованих поверхонь і контурів. На відміну від методів з апріорними геометричними обмеженнями, які можуть вносити зміщення та обмежувати масштаб, прямі розріджені методи VO зазвичай працюють у ковзному вікні активних ключових кадрів, маргіналізуючи далекі точки. Це може обмежувати використання повторних спостережень. Натомість розріджені прямі VSLAM-системи будують постійні карти, де ключові кадри пов'язані спільними спостереженнями в часі.

DSM (2020). [3] [11] Система прямого розрідженого SLAM, запропонована Зубізарретою та ін., базується на DSO, але розширена до повноцінного VSLAM. Основна мета — зменшення дрейфу та покращення точності за рахунок повторних спостережень сцени. На відміну від LDSO, DSM створює постійну карту без використання графа поз чи релокалізації, використовуючи фотометричне формулювання. Ключовими є вікно спільної видимості локальної карти (LMCW), фотометрична обробка повторних точок, t-розподіл як функція впливу та попіксельне управління викидами. Система включає фронтенд для оцінки пози камери та бекенд, який виконує PBA з видаленням викидів та усуненням дублювань. Результати показали перевагу DSM над іншими прямими та непрямими методами на EuRoC завдяки стратегії з 3–4 спільно видимими ключовими кадрами та t-розподілу для моделювання фотометричної похибки.

Щільні методи

Ця категорія підходів SLAM класифікується як щільна через їхню характеристику використання всієї або більшої частини вхідної інформації для виконання реконструкції. Крім того, вони класифікуються як прямі методи через відсутність етапів попередньої обробки, тому вхідне зображення безпосередньо використовується для отримання геометрії середовища. Це формулювання зазвичай застосовує фотометричну похибку та геометричні апріорні значення для оцінки щільної або напівщільної геометрії, оскільки вони безпосередньо працюють з інформацією про інтенсивність пікселів.

Stühmer et al. (2010). [3] Штюмер, Гумгольд і Кремерс запропонували варіаційну систему прямої щільної реконструкції глибини з кількох зображень у реальному часі. Метод базується на мінімізації енергетичної функції, що включає фотометричну узгодженість та згладжування.

Перевагою багатовидової реконструкції є можливість використання різних ракурсів для покращення оцінки в перекритих областях сцени, включаючи інформацію з зображень, де об'єкт видно повністю. Додатково, метод підвищує співвідношення сигнал/шум, що є корисним при роботі з зашумленими відео з веб-камер або мобільних пристроїв. Глибина оцінюється не в реальному часі, а на основі поточного кадру та N найближчих ключових кадрів, що зменшує вплив шуму. Система вбудована в PTAM і використовує його модуль відстеження для зберіган-

ня ключових кадрів та уточнення як пози камери, так і глибини.

DTAM (2011). [3] [8] Ньюкомб та ін. запропонували прямий щільний метод, орієнтований на побудову точної 3D-моделі сцени шляхом щільного та субпіксельного відстеження кожного пікселя. DTAM реалізує оцінку положення камери шляхом мінімізації енергії, що включає фотометричну похибку та просторову регуляризацію, де остання задається через зважену норму Губера градієнта оберненої карти глибини. Побудований фотометричний об'єм витрат дискретизується, а мінімізація виконується в дуальній формі, отриманій через перетворення Лежандра–Френшеля. Карта глибини витягується ітеративно для кожного пікселя з опорного кадру. Ініціалізація виконується за допомогою PTAM, після чого DTAM переходить на власний щільний конвеєр. Новий ключовий кадр додається, коли недостатньо поверхневої інформації в поточному прогнозі. Зрештою, положення камери визначається шляхом пошуку таких параметрів руху, за яких синтетичне зображення найбільш відповідає поточному кадру.

REMODE (2014). [3] [9] Піццолі, Форстер та Скарамуцца запропонували REMODE — прямий метод оцінки щільної карти глибини з використанням баєсівського підходу та регуляризованої оптимізації. Глибина кожного пікселя оцінюється незалежно як параметрична модель, яка оновлюється при нових спостереженнях. На відміну від DTAM, REMODE використовує оцінку невизначеності для опуклого формулювання, що зменшує вплив

шуму. Тріангуляція виконується між опорним та поточним кадрами, а згладжування забезпечується мінімізацією енергії з регуляризацією на основі зваженої норми Губера. REMODE використовує потік відстеження, натхненний одометричною системою SVO, використовуючи формулювання вирівнювання зображення для оцінки пози, але працюючи лише з інформацією про інтенсивність пікселів. Після цього потік відображення тріангулює глибину, використовуючи кожен кадр та опорний вигляд, таким чином глибина кожного пікселя формулюється як параметрична модель, обчислена як баєсівська задача оцінювання, що включає регуляризатор на основі градієнтної норми Губера градієнта, таким чином, рішення досягається ітеративно шляхом мінімізації, використовуючи первинно-дуальну формулювання та техніку градієнтного спуску-підйому.

LSD-SLAM (2014). [3] [10] Енгель та ін. запропонували монокулярну SLAM-систему у реальному часі з 3D-реконструкцією, що використовує напівщільне представлення, де глибина оцінюється лише в областях із високими градієнтами. Система базується на прямому вирівнюванні зображень та фільтрації глибини, будує глобальну карту як граф поз з ключових кадрів, допускаючи масштабні зміни та корекцію дрейфу. Для виявлення циклів використовує FAB-MAP — метод, що працює лише за зовнішнім виглядом, без залежності від результатів переднього кінця. Внески методу LSD-SLAM полягають у прямому методі для виконання вирівнювання двох ключових кадрів на $\boldsymbol{\xi} \in \mathrm{sim}(3)$ та ймовірнісно узгодженому включенні шумової

невизначеності оціненої глибини у відстеження. Нове вирівнювання зображень виконується шляхом мінімізації фотометричної похибки за методом Гауса–Ньютона:

$$E(\mathbf{T}) = \sum_i \left(I_{\text{ref}}(\mathbf{q}_i) - I\left(\omega\left(\mathbf{q}_i, D^*_{\text{ref}}(\mathbf{q}_i), \mathbf{T}\right)\right)\right)^2, \qquad (2.5)$$

де I — зображення, D^* — обернена карта глибини для кожного пікселя, $\mathbf{q}_i$ — точка зображення, ω — 3D проєктивна деформована функція, а $\mathbf{T}$ — положення камери. Повний метод вимагає відстеження, оцінки карти глибини та ініціалізації карти. Модуль відстеження безперервно оцінює положення твердого тіла на поточному ключовому кадрі, тому відносне положення розраховується шляхом мінімізації нормалізованої за дисперсією фотометричної похибки:

$$\min_{\mathbf{T}\in \text{SE}(3)} \sum_{\mathbf{p}\in\Omega D_i} \left\| \frac{r_q^2(\mathbf{q}, \mathbf{T}_{ij})}{\sigma^2_{r_q}(\mathbf{q}, \mathbf{T}_{ij})} \right\|_\delta, \qquad (2.6)$$

де r_q^2 та $\sigma^2_{r_q}$ — фотометричний залишок та його дисперсія відповідно. Також, для додавання ключового кадру до карти, найближчі ключові кадри знаходяться, а ребра оцінюються за допомогою SE(3), тому мінімізація виконується за рівнянням:

$$\min_{\mathbf{T}\in \text{SE}(3)} \sum_{\mathbf{q}\in\Omega D_i^*} \left\| \frac{r_q^2(\mathbf{q}, \mathbf{T}_{ij})}{\sigma^2_{r_q}(\mathbf{q}, \mathbf{T}_{ij})} + \frac{r_d^2(\mathbf{q}, \mathbf{T}_{ij})}{\sigma^2_{r_d}(\mathbf{q}, \mathbf{T}_{ij})} \right\|_\delta. \qquad (2.7)$$

Підсумовуючи, у LSD-SLAM положення камери визначається відносно поточного ключового кадру, а нові зображення використовуються для уточнення карти глибини шляхом багаторазових

оцінок з малою базовою лінією. Після завершення уточнення новий ключовий кадр додається до глобального графа, що забезпечує оптимізацію карти.

2.4 Методи машинного навчання

В останні роки, враховуючи вражаючий розвиток штучного інтелекту, особливо в глибокому навчанні, дослідники досліджували можливість розширення класичних методів SLAM шляхом впровадження глибоких нейронних мереж для виконання різних завдань, таких як зменшення похибки оцінки глибини, покращення якості 3D-реконструкції, покращення оцінки пози камери, покращення замикання циклу, оцінка масштабу тощо.

Одним із головних недоліків монокулярних методів є неоднозначність масштабу, зумовлена відсутністю початкового вимірювання глибини. Щоб компенсувати це, методи глибокого навчання інтегруються в SLAM-системи для оцінки глибини, руху камери, семантики та оптичного потоку. Дослідження охоплюють заміну ручних ознак вивченими, використання нейронних 3D-уявлень, гібридні рішення з класичними компонентами та повністю навчувані end-to-end SLAM-системи.

Зазвичай методи, засновані на навчанні, важче пояснити, і вони зазвичай мають труднощі при застосуванні в невидимих середовищах або з різними калібруваннями камери. Однак вони все ще є перспективним рішенням для монокулярного візуального SLAM, особливо в чисто обертальних застосуваннях, уповільненому русі або рухах без обертань по кренах та тангажу, де

системи SLAM можуть зіткнутися з проблемами ініціалізації або узагальнення.

2.4.1 Непрямі методи

Глибоке навчання сприяло виконанню систем 3D-реконструкції багатьма завданнями, значно покращуючи загальну продуктивність системи. Завдання варіюються від простих оцінок параметрів для ініціалізації систем SLAM до заміни цілого модуля в конвеєрі SLAM (наприклад, оцінка глибини або пози), або навіть надання нових можливостей системі (наприклад, семантична сегментація). Як непрямі методи, дослідження в цій категорії включають етапи попередньої обробки у своїх рамках, у вигляді вилучення ознак та вилучення оптичного потоку, що виконуються як кроки, що передують процесам оцінки пози або прогнозування глибини, тому якість кінцевої 3D-карти безпосередньо залежить від кількості та якості інформації, вилученої на цьому етапі попередньої обробки.
Подібно до класичних методів, ML + непрямі методи можуть відновлювати щільні та розріджені 3D-карти середовища.

Розріджені методи

Як і класичні методи, методи машинного навчання + непрямі можна внутрішньо класифікувати як щільні та розріджені залежно від щільності отриманої 3D-реконструкції. Розріджені методи мають перевагу роботи з невеликою кількістю інформації. Однак, водночас, це є обмеженням для остаточної реконструкції,

оскільки деяка цінна інформація також може бути виключена на етапах вибору пікселів або вилучення ознак, що також мотивувало розробку архітектур CNN, що спеціалізуються на ущільненні отриманої 3D-карти, досягаючи багатообіцяючих результатів, отримуючи при цьому переваги продуктивності роботи в розрідженій парадигмі.

DynaSLAM (2018). [3] [13] Бескос та ін. запропонували систему, побудовану на ORB-SLAM2 з інтеграцією CNN і багатовидової геометрії для виявлення та сегментації динамічних об'єктів. Метод використовує Mask R-CNN для попіксельної сегментації відомих рухомих класів (наприклад, людей, авто), а також геометричний аналіз паралакса для виявлення нерозпізнаних об'єктів. Потік відстеження базується на ORB-SLAM2 і виконує стандартні етапи локалізації. Поєднання CNN і геометрії дозволяє точно фільтрувати динамічні області, зменшуючи вплив рухомих об'єктів на SLAM. Крім того, застосовується фонове заповнення сцен за попередніми кадрами, після чого відстеження та відображення виконуються лише на статичних сегментах.

Steenbeek et al. (2022). [3] [14] Автори представили модифікацію ORB-SLAM2 для застосування на БПЛА в надзвичайних умовах, інтегруючи CNN для масштабованої та ущільненої реконструкції. Вони поєднали ORB-SLAM2 з SIDE — алгоритмом оцінки глибини з одного зображення — що дозволяє компенсувати відсутність масштабу та щільності в оригінальному ORB-SLAM2. RGB-кадри та розріджена глибина передавались у CNN,

яка будувала масштабовану щільнішу карту, згодом об'єднувану за допомогою OctoMap. CNN базувалась на ResNet-50 із декодером з блоками апсемплінгу; оцінена глибина використовувалась для масштабування SLAM-карти. Хоча підхід значно ущільнив карту, якість залишалась нижчою за стерео-методи, і автори зазначили обмеження продуктивності через слабке обладнання та потенціал для покращень при використанні потужніших систем.

Sun et al. (2022). [3] Автори запропонували розширення ORB-SLAM із вбудованим модулем оцінки глибини на базі DiverseDepth, що працює у двох режимах: перший — передбачає відносну глибину з одного зображення для покращення відстеження, другий — уточнює масштабовану глибину з використанням розрідженої SLAM-карти для щільнішого картографування. Архітектура модулю — кодер-декодер на основі ResNetXt-50 — була навчена на п'яти різних наборах даних для покращення узагальнення. Завдяки цьому система демонструє вищу точність пози та глибини, особливо в умовах як приміщення, так і відкритого простору. Проте, попри щільнішу реконструкцію порівняно з оригінальним ORB-SLAM, результат ще не відповідає рівню повноцінних щільних методів.

Lee et al. (2022). [3] Автори запропонували розширення ORB-SLAM, інтегруючи глибоку нейронну мережу з модулями оцінки 3D-геометрії та семантичної сегментації для підвищення точності SLAM у задачах автономного керування. Система використовує ERFNet для видалення рухомих об'єктів та покращення кутових

ознак, а також сегментує сцену для точнішого відображення. Модулі локалізації, картографування та сегментації працюють узгоджено: перший визначає ключові кадри, другий виконує тріангуляцію та оцінку масштабу, третій — семантичне очищення карти. Особливістю підходу є корекція масштабу на основі площини землі та спеціальна функція втрат, що окремо штрафує помилки трансляції й обертання. Система демонструє стійкість до різних умов освітлення та покращену якість реконструкції.

SVR-Net (2023). [3] SVR-Net — система SLAM для візуальних і візуально-далекомірних датчиків, яка поєднує регресію на основі опорних векторів (SVR) для оцінки 3D-розташування ключових точок. Це забезпечує надійне відстеження навіть за складних умов, як-от погане освітлення чи перекриття. Система підтримує онлайн-навчання, графову оптимізацію та замикання циклу для підвищення точності й узгодженості карти. Вона здатна створювати високодетальні 3D-карти складних середовищ із багаторівневою та нерівною геометрією, залишаючись обчислювально ефективною для роботи в реальному часі.
Коротко кажучи, SVR-Net SLAM реалізує стратегію від грубого до точного для ефективного відстеження та щільного глобального картографування. Система складається з двох етапів: спочатку оцінюється необроблена поза та локальна карта з пари кадрів за допомогою мережі SVR, де карта представляється як розріджені вокселі зі значеннями TSDF. Потім глобальна карта розширюється. На другому етапі виконується збільшення вибірки вокселів, уточнення пози та карти, і локальна карта інте-

грується в глобальну. Мережа SVR-Net, навчена на ScanNet(V2), приймає пару RGB-зображень і координати вокселів, видаючи локальну карту, відносну позу та TSDF. Вона будує карти ознак, перетворює їх у вокселі, оцінює кореляції між кадрами, ітеративно оновлюючи позу та карту шляхом зіставлення ознак та коригування координат. Нарешті, SLAM-конвеєр на основі Kinect-Fusion розширює глобальну карту, використовуючи уточнені локальні дані.

Щільні методи

Непрямі методи зазвичай використовують оптичний потік для прогнозування глибини. Варто зазначити, що загальновідомим складним завданням є характеристика похибки розташування об'єктів у структурі VO, де розмиття руху, перекриття та варіації точки зору можуть спотворити ці оцінки. Зокрема, ефективність прямих методів зазвичай залежить від припущень про малий рух та сталість зовнішнього вигляду, що становить обмеження для забезпечення стійкості до мінливості сцени, що зменшує їхню застосовність. Нещодавно оцінка оптичного потоку за допомогою машинного навчання, яку можна описати як комбінацію жорсткого потоку та необмеженого потоку, що описує загальний рух об'єкта, досягла найсучаснішого рівня продуктивності, демонструючи чудовий рівень точності, стійкості та узагальнення, стаючи чудовим рішенням, особливо в складних умовах, таких як поверхні без текстур, розмиття руху та великі перекриття.

DeepV2D (2020). [3] [15] DeepV2D — це наскрізна система, запропонована Тідом та Денгом, яка чергує модулі глибини та руху для спільного прогнозування карти глибини та пози камери. Модуль руху використовує оцінки глибини для уточнення пози, а модуль глибини покладається на рух для обчислення глибини, що дозволяє точну структуру з руху в наскрізно диференційованій архітектурі. Модуль глибини створює об'єм витрат із вивчених ознак за допомогою 2D-екстрактора, зворотного проектування та 3D-стереоузгодження. Модуль руху прогнозує збурення пози через залишковий потік між ознаками, використовуючи архітектуру з ініціалізацією, вилученням ознак, підрахунком помилки та оптимізацією через Гауса-Ньютона. Хоч DeepV2D переважно орієнтований на глибину, його можна адаптувати для SLAM, перетворюючи оптичний потік безпосередньо в рух камери. Важливо, що система включає Flow-SE3 — геометрично обмежену модель оцінки руху, яка покращує точність повторного проектування порівняно з попередніми методами, як DeMoN та DeepTAM.

DROID-SLAM (2021). [3] [16] DROID-SLAM — це система на основі глибокого навчання, що виконує рекурентні ітеративні оновлення поз камери та карт глибини через модуль корекції щільних зв'язків. Вона базується на архітектурі RAFT і використовує GRU-блок для формування члена корекції, оновлюючи не оптичний потік, а безпосередньо глибину й позу. Система має фронтенд (додавання кадрів, вибір ключових, локальна опти-

мізація) та бекенд (глобальна оптимізація історичних кадрів). DROID-SLAM підтримує монокулярний, RGB-D і стереовхід, досягаючи високих результатів на наборах TartanAir, EuRoC, TUM-RGB-D і ETH3D-SLAM, перевершуючи більшість класичних і вивчених методів. Основне обмеження — високе споживання пам'яті GPU (до 24 ГБ).

SDF-SLAM (2022). [3] SLAM-система, заснована на ORB-SLAM, яка реалізує новий метод вилучення ознак і щільну семантичну мережу для оцінки глибини, входячи до категорії щільних підходів. На відміну від ORB-SLAM, що забезпечує лише розріджену карту без семантики, SDF-SLAM поєднує пози камери з глибиною та семантичними мітками для створення тривимірної семантичної реконструкції.
Система складається з трьох основних модулів:

1) FPFDCNN — вилучає та зіставляє ознаки з пари кадрів, формуючи дескриптори;
2) SDFCNN — одночасно виконує прогноз глибини та семантичну сегментацію з RGB-зображень;
3) SLAM-модуль — оптимізує хмару точок на основі оціненої глибини, семантики та пози.

NeRF-SLAM (2022). [3] [17] У 2022 році було представлено NeRF-SLAM — монокулярну систему реконструкції сцен у приміщенні, що поєднує підхід Neural Radiance Fields (NeRF) із DROID-SLAM як модулем відстеження.
NeRF-SLAM складається з трьох компонентів: фронтенду, по-

будованого на основі DROID-SLAM, що оцінює пози камери та генерує розріджені 3D-хмари точок із ковзного вікна ключових кадрів; модуля нормалей, що прогнозує поверхневі нормалі з неявного представлення; та бекенду, який виконує глобальну оптимізацію сцени. Оцінювання відбувається шляхом обчислення оптичного потоку між кадрами за допомогою згорткового GRU, після чого вирішується задача Bundle Adjustment у формі лінійного рівняння найменших квадратів із використанням інверсних карт глибини. Отримані карти та пози використовуються для оптимізації параметрів поля випромінювання сцени. Потік відстеження постійно мінімізує помилку перепроєкції в активному вікні ключових кадрів, тоді як бекенд уточнює всі попередні ключові кадри.

2.4.2 Прямі методи

В останні роки машинне навчання стало перспективним напрямком для досліджень монокулярної 3D-реконструкції, головним чином завдяки своїй здатності робити внесок у вирішення відомих обмежень монокулярних систем, таких як неоднозначність масштабу, розмиття руху, поверхні без текстури та повторювані візерунки, серед іншого. Одними з найбільших проблем для класичних прямих систем є їхня сильна залежність від хорошої ініціалізації, припущення про сталість яскравості, продуктивність в умовах низької освітленості та здатність до узагальнення в невидимих середовищах. Відповідно, за останнє десятиліття дослідники зробили цікавий внесок у подолання цих

проблем, вбудовуючи архітектури нейронних мереж у класичні пропозиції SLAM, підвищуючи продуктивність і, в більшості випадків, продемонстрували перевагу над своїми класичними версіями.

Методи ML + Direct, як і решта категорій таксономії, можна розділити на щільні та розріджені, залежно від щільності відновленої 3D-реконструкції. Два найрепрезентативніші підходи машинного навчання належать до цієї категорії: CNN-SLAM та CodeSLAM, які значною мірою сприяють розвитку сучасних технологій, забезпечуючи вражаючі показники цитування, пропонуючи дослідникам два цікаві шляхи: інтеграцію семантичної сегментації та використання архітектур кодера-декодера відповідно.

Розріджені методи

У цьому розділі представлені найуспішніші інтеграції машинного навчання, виконані над системою DSO та вдосконалені версії, де інтеграція нейронних мереж була виконана для покращення якості їхньої карти, відстеження, можливостей узагальнення та подолання їхніх відомих режимів відмови, таких як порушення припущень яскравості, розмиття руху, повторювані текстури та інші.

CNN-DVO (2020). [3] [12] У 2020 році Ченг та ін. представили CNN-DVO — перший прямий фреймворк, що повністю інтегрує прогнозування глибини від CNN у компоненти ініціалізації, від-

стеження та оптимізації SLAM. Використовуючи Monodepth2 як предиктор глибини, система включає глибину як апріор для ініціалізації, пригнічення масштабного дрейфу, стабілізації локального BA та коректного замикання циклів із відновленням масштабу.

Метод поєднує грубе відстеження з уточненням карти, натхненний LDSO, та виконує спільну оптимізацію всіх параметрів (пози $SE(3)$, оберненої глибини, афінної моделі). Вибір точок вдосконалено завдяки мультишаровому аналізу градієнтів, що мінімізує кластеризацію, підвищуючи стабільність оцінки поз при великих базових лініях. Структура графа пози використовує ORB-ознаки та BoW для вибору ключових кадрів, з урахуванням маргіналізації та забезпечення замикання циклу. Оцінювання пози виконується з грубого апріору, після чого глибина уточнюється оптимізацією по епіполярній лінії через ітерації Гауса–Ньютона.

Щільні методи

Як і класичні методи SLAM, щільні формулювання машинного навчання можна класифікувати як щільні та розріджені залежно від розрідженості кінцевої реконструкції. У цій категорії деякі автори запропонували покращити оцінку пози, оцінку глибини та обидва, або навіть оцінити деякі параметри, такі як масштабні коефіцієнти або умови ініціалізації. Крім того, як показано в попередньому розділі, існує можливість навчання мереж для ущільнення виводу карти класичних систем SLAM;

таким чином, класичні розріджені методи були додані до цієї категорії з їх ущільненими версіями машинного навчання. Більшість прямих підходів машинного навчання належать до категорії щільних завдяки чудовим результатам, яких машинне навчання досягло в ущільненні своїх 3D-реконструкцій.

CNN-SLAM (2017). [3] [18] Запропоновано систему, що поєднує глибоке згорткове прогнозування глибини з прямим підходом LSD-SLAM для монокулярної реконструкції з абсолютним масштабом. Глибина, передбачена CNN, служить апріорною інформацією для кожного ключового кадру, допомагаючи уникнути неоднозначності масштабу та проблем при чисто обертальному русі камери. Архітектура CNN базується на ResNet-50, модифікованому для генерації карти глибини через апсемплінгові залишкові блоки. Результат уточнюється щільним прямим SLAM, що забезпечує точну траєкторію та масштабовану 3D-реконструкцію.

DeepTAM (2018). [3] [19] Система DeepTAM Чжоу та ін. заснована на DTAM, але переосмислена як задача глибокого навчання з двома CNN: для відстеження та картографування. Основні внески включають мережу відстеження з інкрементною оцінкою пози за допомогою гіпотез, що вирівнює поточне зображення з ключовим кадром (із глибиною та кольором), та мережу картографування, яка поєднує оцінку глибини з вхідним зображенням, використовуючи стратегію від грубого до точного. Для відстеження використовуються три мережі з різною роздільною

здатністю, що прогнозують оптичний потік і послідовно оновлюють позу. Остаточна поза є результатом множення інкрементних змін, а метод вузької смуги застосовується для уточнення карти глибини.

DeepFusion (2019). [3] Система спрямована на отримання щільних масштабованих карт глибини та пози в реальному часі з монокулярного SLAM, поєднуючи ORB-SLAM2 з прогнозами глибини та її градієнтів від CNN у ймовірнісній структурі. Прогнозовані градієнти виступають як обмеження для сусідніх пікселів, забезпечуючи глобальну узгодженість, тоді як прогнозовані невизначеності (середнє й дисперсія на піксель) допомагають злиттю даних. Глибина оптимізується окремо для кожного ключового кадру з урахуванням нових геометричних обмежень. Вартістьна функція враховує піксельні втрати та парні обмеження на основі градієнтів. Архітектура базується на модифікованій U-Net, що прогнозує логарифмічну глибину та її градієнти разом з відповідними невизначеностями, забезпечуючи масштабну інваріантність та підвищену точність реконструкції. У формулювання було включено пов'язану з цим невизначеність для об'єднання виводу CNN з оцінками монокулярної системи SLAM для кожного пікселя в кожному з зображень логарифмічної глибини та градієнта, що було виконано шляхом навчання мережі прогнозувати середнє значення та дисперсію шляхом навчання мережі на наборі даних SceneNet RGB-D, використовуючи функцію максимальної правдоподібності.

CodeSLAM (2018). [3] [20] Система CodeSLAM Блоша та ін. запропонувала компактне, але щільне представлення геометрії сцени у вигляді коду — невеликого набору параметрів, що спільно з інтенсивностями зображення дозволяє реконструювати карту глибини: $D = D(I, c)$. Ідея полягає в тому, що інтенсивності вже несуть більшість інформації, тому код зберігає лише те, що не доступне напряму. Архітектура використовує U-Net як основу для витягнення ознак, далі застосовується автоенкодер з варіаційним вузьким місцем для кодування в компактний гауссівський код. Мережа також оцінює середню глибину та її невизначеність на кількох рівнях.

CodeSLAM реалізує SLAM-фреймворк на основі N-кадрового SFM, де пози та коди оптимізуються спільно через фотометричні й геометричні втрати, з оновленням за алгоритмом Гауса–Ньютона. Система реалізована за принципом PTAM: чергування відстеження та відображення, ініціалізація двома кадрами, подальше додавання ключових кадрів і глобальна оптимізація. Завдяки такому підходу CodeSLAM демонструє підвищену точність і стійкість, зокрема при швидких або чисто обертальних рухах, перевершуючи попередні методи на наборі даних EuRoC.

DeepFactors (2020). [3] [21] Система DeepFactors, запропонована Чарновським та ін., є ймовірнісним SLAM-фреймворком, що поєднує класичні прямі методи з вивченими апріорними представленнями глибини в структурі фактор-графа. Вона базується на CodeSLAM, але вводить новий сервер відображення та формулювання задачі як багатовидового bundle adjustment.

DeepFactors використовує три типи факторів: фотометричний (інтенсивність), перепроєкції (відстань до очікуваного положення орієнтира) та геометричний (розбіжність карт глибини), що дозволяє точніше реконструювати сцени навіть у безтекстурних ділянках.

Глибина представлена компактним кодом, отриманим варіаційним автоенкодером (VAE) з модифікованим U-Net, який витягує ознаки та прогнозує глибину разом з невизначеністю. Оптимізація відбувається за допомогою логарифмічної втрати Лапласа та L1 для контролю відновленої глибини. Система орієнтована на продуктивність у реальному часі за рахунок GPGPU та є стійкою до локальних мінімумів завдяки багатофакторному баченню сцени.

2.5 Висновки

Методи машинного навчання, особливо згорткові нейронні мережі, були широко інтегровані в різні класичні пропозиції для подолання проблем та модальностей невдач, виявлених у кожній категорії таксономії. Слід зазначити, що багато дослідників вважають, що класичні геометричні підходи все ще перевершують методи машинного навчання щодо якості реконструкції або відстеження.

Однак, як проаналізовано, дослідники машинного навчання намагалися подолати цю проблему, розробляючи та впроваджуючи нові або більш просунуті архітектури CNN, тренуючи їх на нових та складніших наборах даних для покращення процесу навчання.

З іншого боку, як показано в багатьох з вищезгаданих систем, CNN можна застосовувати не лише для відновлення глибини сцени або пози камери, але й ефективно використовувати для ущільнення карт глибини, оцінки параметрів ініціалізації, виконання етапів попередньої обробки, таких як вилучення ознак або оцінка оптичного потоку, виконання додаткових завдань, таких як семантична сегментація тощо. Таким чином, з різних точок зору, CNN може позитивно внести свій вклад у SLAM. Характеристики усіх методів можна знайти у таблицях 3.1-3.8 [Додаток А і Б].

РОЗДІЛ 3

ПРАКТИЧНА РЕАЛІЗАЦІЯ SLAM

Згідно з наведеною вище інформацією, можемо визначити, де краще використовувати методи SLAM:

3.1 Класичні методи

MonoSLAM:

- Сцени з високою структурністю: наявність чітких граней, кутів, текстурованих поверхонь.
- Невеликі або контрольовані простори: закриті кімнати або лабораторії.
- Повільний або стабільний рух камери: EKF чутливий до швидких переміщень або розмиття.
- Відсутність динамічних об'єктів: сцена повинна бути статичною.
- Низька роздільна здатність зображення: метод створювався під QVGA або VGA (320×240, 640×480).
- Обмежена кількість ознак: через обмеження EKF кількість активних фіч зазвичай не перевищує 100.

PTAM (Parallel Tracking and Mapping):

- Добре освітлені сцени з достатньою кількістю візуальних ознак: працює найкраще при наявності кутів, країв, текстурованих об'єктів.

- Стабільний рух камери: метод орієнтований на ручне, поступове переміщення камери.
- Малий або середній масштаб сцени: оптимальний для закритих indoor-сцен з обмеженим простором.
- Монокулярне відео з високою частотою кадрів: забезпечує точне відстеження за рахунок щільності кадрів.
- Низька швидкість сцени або відсутність динаміки: не призначений для роботи з рухомими об'єктами.

ORB-SLAM

- Сцени з насиченою текстурою: велика кількість чітких ознак (краї, кути, об'єкти) для детектора ORB.
- Стабільне освітлення та хороша освітленість: детектор ORB менш стійкий до зміни яскравості або тіней.
- Статичне середовище: метод не справляється з динамічними об'єктами у сцені.
- Повільне або контрольоване переміщення камери: занадто швидкий рух може призвести до втрати трекінгу.
- Монокулярні або стерео-налаштування: базова версія ORB-SLAM підтримує як однокамерну, так і стерео-конфігурації.
- Середні та великі indoor-сцени: добре масштабується, підтримує *loop closure* та глобальну оптимізацію.

ORB-SLAM2

- Сцени з помірною або багатою текстурою: ORB-дескриптори забезпечують ефективне відстеження ключових точок.
- Стерео або RGB-D камери: на відміну від ORB-SLAM, друга

версія підтримує додаткові джерела глибини.

- Середні та великі сцени: метод масштабується та містить розвинену глобальну оптимізацію з петлевим замиканням.
- Постійне освітлення: ORB-ознаки менш стійкі до сильних змін яскравості.
- Статичні сцени: ORB-SLAM2 не розрахований на динамічне середовище.
- Збалансована траєкторія камери: раптові або занадто швидкі рухи можуть призвести до втрати трекінгу.

ORB-SLAM3

- Сцени з достатньою кількістю візуальних ознак: детектор ORB потребує виражених кутів, країв та текстур.
- Будь-яке налаштування камери: підтримуються монокулярні, стерео, RGB-D та мультикамерні системи.
- Змішані або великі простори: метод підтримує як локальні реконструкції, так і великомасштабні карти з loop closure.
- Стабільне освітлення: ефективність ORB-детектора зменшується при сильних перепадах освітлення.
- Використання IMU (інерційних даних): дозволяє покращити стійкість трекінгу у разі швидких рухів камери або слабкої текстури.
- Об'єкти з мінімальною динамікою: як і попередники, не пристосований до активної динаміки у сцені.

Stühmer et al.

- Сцени з великою кількістю глибини та фактур: метод за-

снований на щільному прямому узгодженні, що вимагає достатньої кількості змін у яскравості між кадрами.

- Добре освітлені indoor-сцени: найкраще працює в контрольованих умовах, де можна уникнути пересвітів або тіней.
- Поступове переміщення камери: великі міжкадрові зміщення погіршують щільне узгодження.
- Однотипне або гладке середовище: метод не обмежується ключовими точками, тому працює навіть там, де мало кутових ознак.
- Монокулярні відеопослідовності: алгоритм розрахований на одноокі налаштування.

DTAM

- Сцени з достатньою фактурністю: метод будує щільну карту, тож навіть слабко текстуровані ділянки можна реконструювати, однак для стабільного трекінгу бажана наявність текстур.
- Стабільне освітлення: яскраві перепади між кадрами можуть порушити фотометричне узгодження.
- Невеликі indoor-простори: найкраще працює в закритих контрольованих середовищах (кімнати, лабораторії).
- Плавний рух камери: великі міжкадрові зміщення ускладнюють фотометричну оптимізацію.
- Монокулярне відео: DTAM призначений для однооких камер з неперервною подачею зображень.

REMODE

- Сцени з помірною текстурованістю: хоча REMODE не залежить від ключових точок, наявність текстури покращує стабільність реконструкції.
- Монокулярна камера з відомою траєкторією: REMODE потребує відомої позиції камери для оцінки глибини.
- Постійне або стабільне освітлення: фотометрична модель передбачає, що яскравість пікселів не змінюється суттєво між кадрами.
- Indoor-сцени: переважно застосовується у закритих просторах з обмеженою динамікою.
- Плавні переміщення: алгоритм вимагає узгодження між кількома кадрами, тому великі зміщення зменшують якість.

LSD-SLAM

- Сцени з добре вираженими градієнтами яскравості: замість ключових точок метод використовує великі області з високими градієнтами, тому підходить для сцен із фактурними поверхнями.
- Монокулярне відео з високою частотою кадрів: для якісної оцінки глибини потрібен плавний і поступовий рух камери.
- Indoor-простори з помірною складністю: метод особливо ефективний для реконструкцій інтер'єрів з постійним освітленням.
- Помірна динаміка сцени: LSD-SLAM чутливий до рухомих об'єктів, які можуть впливати на оцінку глибини.
- Висока роздільна здатність зображень: пряме узгодження виграє від більшої кількості пікселів з вираженими змінами

інтенсивності.

DSM

- Сцени з помітними градієнтами інтенсивності: DSM (Direct Sparse Mapping) використовує пряме узгодження на рівні окремих пікселів з високою інформаційністю, що дозволяє працювати в умовах, де класичні ключові точки важко знайти.
- Контрольовані indoor-умови: найкраще підходить для приміщень зі стабільним освітленням та невеликою кількістю динаміки.
- Монокулярне відео з поступовим рухом камери: важливо забезпечити плавні зміни точки огляду для коректного узгодження пікселів.
- Висока роздільна здатність зображень: більша кількість інформаційних пікселів дозволяє точніше оцінювати глибину.
- Сцени без різких освітлювальних змін: DSM є чутливим до фотометричних відхилень.

3.2 Методи машинного навчання

DynaSLAM

- Сцени з динамічними об'єктами: DynaSLAM ефективно виділяє рухомі об'єкти в кадрі, завдяки чому може працювати в середовищах із людьми, транспортом, або іншими рухомими елементами.

- Indoor та outdoor середовища: алгоритм придатний для використання як у приміщеннях, так і на відкритому повітрі.
- Монокулярні, стерео або RGB-D камери: метод підтримує різні типи вхідних даних, що дозволяє адаптацію до доступного обладнання.
- Помірна або висока роздільна здатність зображень: для коректного сегментування об'єктів та точного трекінгу бажано використовувати зображення хорошої якості.
- Стабільне освітлення: хоча DynaSLAM і включає в себе модуль глибини, фотометрична узгодженість впливає на якість результатів.

Steenbeek et al.

- Сцени зі структурованими просторами: метод орієнтований на відносно передбачувані середовища, наприклад, кімнати або коридори, де можна зробити геометричні припущення.
- Монокулярні RGB зображення: використовує лише колірні зображення без потреби в додаткових сенсорах.
- Обмежена динаміка: хоча метод передбачає оцінку руху, основна реконструкція вимагає переважно статичної сцени.
- Наявність нейронної оцінки структури: підходить для задач, де важлива глобальна реконструкція приміщень з оцінкою взаємного розміщення стін, підлоги тощо.
- Підвищена освітленість сцени: для коректної роботи нейромережевих модулів потрібно якісне зображення без сильного шуму або темних зон.

Sun et al.

- Сцени з чіткою глибинною структурою: метод покладається на точну регресію глибини за допомогою глибокої нейронної мережі, тому краще працює у середовищах із добре вираженою тривимірною геометрією.
- Монокулярні або стерео RGB-кадри: метод може працювати з одним або кількома видами вхідних даних, однак найкращі результати досягаються при використанні послідовностей.
- Indoor- і outdoor-сцени: універсальність архітектури дозволяє працювати як у закритих приміщеннях, так і на відкритому просторі.
- Висока роздільна здатність зображень: нейромережа потребує детального зображення для якісної оцінки глибини.
- Контрольовані умови освітлення: хоча модель може бути навчена на змінному світлі, різкі тіні або блиски все ще можуть створювати артефакти в реконструкції.

Lee et al.

- Сцени з високою варіативністю структури: метод побудований на використанні глибоких згорткових мереж для оцінки глибини та руху, тому добре адаптується до сцен із різними геометричними особливостями.
- Монокулярні зображення: орієнтований на ситуації, де доступна лише одна камера, що зменшує вимоги до апаратного забезпечення.
- Стабільна освітленість і чіткість кадрів: якість реконстру-

кції значно покращується при відсутності сильного шуму, розмиття та артефактів у зображеннях.

- Об'єкти з м'якими градієнтами текстур: метод краще справляється з текстурно слабкими областями, ніж традиційні SLAM-системи, завдяки контекстній обробці нейромережею.
- Використання в автономних системах або AR/VR: підходить для сценаріїв, де важлива щільна реконструкція простору без високої точності трекінгу.

SVR-Net

- Сцени з вираженою структурною геометрією: метод спеціалізується на реконструкції тривимірних сцен із чіткими межами та контурами, зокрема кімнат, коридорів або офісних приміщень.
- Наявність відеопослідовностей: працює найкраще з кількома послідовними зображеннями, що дозволяє мережі враховувати часову послідовність для кращої реконструкції.
- Середовища з обмеженою динамікою: хоча метод має деяку стійкість до руху в сцені, він ефективніше функціонує в статичних або малодинамічних умовах.
- Висока роздільна здатність і якість зображень: якісне освітлення і деталізація покращують здатність мережі до відтворення глибини та форми об'єктів.
- Орієнтація на indoor-середовища: більшість тренувальних даних для SVR-Net сформовано на основі внутрішніх при-

міщень.

DeepV2D

- Сцени з глибокою структурною інформацією: метод об'єднує глибину та позицію камери через рекурентну оптимізацію, що робить його особливо ефективним для складних тривимірних сцен із глибокими перспективами.
- Відеопослідовності з монокулярної або стерео камери: потребує кількох кадрів для об'єднання вхідної інформації та покращення точності.
- Indoor та outdoor сцени: універсальність архітектури дозволяє застосування в широкому спектрі середовищ.
- Стабільні умови зйомки: метод чутливий до сильного шуму, тіней або надмірного освітлення.
- Висока якість зображень: мережа демонструє кращу реконструкцію при високій роздільній здатності та чітких текстурах.

DROID-SLAM

- Монокулярне або стерео-відео: метод працює як з одним, так і з кількома зображеннями, динамічно оновлюючи позу камери та карту.
- Сцени з багатою геометричною інформацією: добре функціонує у середовищах із чітко визначеними об'єктами, поверхнями, глибинами.
- Висока точність при коротких і середніх траєкторіях: завдяки інтеграції трекінгу та реконструкції у єдину нейронну

архітектуру.

- Відносна стійкість до динаміки сцени: глибокі особливості дозволяють частково компенсувати наявність рухомих об'єктів.
- Застосування в автономній навігації та AR/VR: метод забезпечує щільну карту і точне локалізування в режимі реального часу.

SDF-SLAM

- Сцени з чітко вираженою поверхневою геометрією: метод використовує Signed Distance Function (SDF) для представлення карти, що дозволяє досягти високої деталізації поверхонь.
- Щільна реконструкція внутрішніх середовищ: особливо ефективний у кімнатах, коридорах або інших структурованих просторах.
- Постійна подача RGB зображень: метод ґрунтується на фотометричній відповідності кадрів та об'ємній реконструкції.
- Стабільний рух камери: забезпечує точніше оновлення карти та локалізації.
- Моделі з помірним освітленням і текстурами: фотометричний error є чутливим до шуму, зміни освітлення та відблисків.

NeRF-SLAM

- Сцени з багатою текстурою та освітленням: метод базується на Neural Radiance Fields (NeRF), що дозволяє фото-

реалістичну реконструкцію лише за умови різноманітного освітлення та детального вигляду сцени.

- Монокулярні або стерео відеопослідовності: підтримується використання окремих кадрів або коротких послідовностей з різних точок зору.
- Плавний, добре покритий рух камери: для якісного навчання NeRF необхідне перекриття між кадрами з достатньою кількістю ракурсів.
- Статичні сцени: будь-який рух об'єктів у сцені викликає деградацію якості реконструкції.
- Середовища з обмеженим масштабом: застосовується переважно для малих до середніх indoor сцен.

Rosinol et al.

- Складні внутрішні сцени: метод адаптований для роботи у великих indoor-просторах з багатьма приміщеннями, рівнями або переплануванням.
- Використання багатьох сенсорів: підтримує комбінацію RGB, глибини, IMU — особливо ефективний у випадках, де доступні дані з кількох джерел.
- Застосування у багатокамерних системах: підходить для сценаріїв з кількома камерами або агентами (multi-agent SLAM).
- Потреба в об'єктно-орієнтованій карті: підтримує семантичну реконструкцію з виокремленням об'єктів у сцені.
- Інтеграція в реальні системи: сумісний з ROS2 та оптимізований для практичного використання в мобільній робото-

техніці.

CNN-SLAM

- Сцени з маловираженою глибиною: використання згорткових нейронних мереж (CNN) дозволяє оцінювати глибину навіть у слабо текстурованих або неструктурованих регіонах.
- Монокулярні послідовності: CNN-SLAM компенсує відсутність стереоінформації за рахунок попередньо навчених моделей глибини.
- Indoor-середовища: метод був розроблений для кімнат, коридорів, офісів — середовищ із передбачуваною геометрією.
- Плавний рух камери: забезпечує якісніше поєднання кадрів і узгодження карти.
- Камери з низькою або змінною роздільною здатністю: нейромережевий компонент дозволяє частково компенсувати обмежену якість зображення.

DeepTAM

- Статичні сцени з помірною текстурою: DeepTAM добре справляється в умовах середньої насиченості ознаками, поєднуючи глибинне оцінювання та трекінг.
- Монокулярні або стерео відеопослідовності: система підтримує як одиночну, так і пару камер для покращення оцінки глибини.
- Indoor-оточення: орієнтована на приміщення, квартири, офіси — із чітко вираженими геометричними структурами.

- Плавний рух камери: необхідний для послідовної локалізації та точного оновлення карти.
- Потреба у щільній карті: метод надає щільну реконструкцію, що є корисним для задач робототехніки, AR/VR тощо.

DeepFusion

- Середовища з високою варіативністю структури: метод добре працює як у структурованих (приміщення), так і в менш структурованих (вулиці) сценах.
- Потреба у щільній 3D-реконструкції: DeepFusion фокусується на фюзії кадрів з багатьох точок зору для побудови високоякісної сцени.
- Послідовності з монокамери або стерео: підтримує різні режими введення даних, зокрема глибину з RGB-D або оцінку глибини через нейромережу.
- Стабільний рух камери: забезпечує кращу акумуляцію інформації у воксельній сітці.
- Сцени без суттєвих динамічних об'єктів: динаміка може призводити до накопичення артефактів у моделі сцени.

CodeSLAM

- Сцени з великою варіативністю структури: CodeSLAM здатен відновлювати щільну карту навіть у складних сценах завдяки використанню компактного латентного коду.
- Використання тільки монокамери: метод розроблено для роботи без глибинних сенсорів або стерео, що знижує апаратні

вимоги.

- Наявність обмеженого обчислювального ресурсу: через ефективне представлення глибини у вигляді вектору коду, підходить для вбудованих систем.
- Стабільний рух камери: послідовна та передбачувана траєкторія покращує якість оцінки.
- Сцени без динаміки: метод не адаптований до сцен з рухомими об'єктами або людьми.

DeepFactors

- Сцени з насиченою геометрією: метод ефективний у середовищах із виразними структурами та текстурами, зокрема в закритих приміщеннях або технічних інтер'єрах.
- Монокулярні послідовності: не потребує стерео або RGB-D камер — реконструкція базується на оцінці глибини за допомогою нейромережі.
- Потреба у гнучкому балансі між традиційною графовою оптимізацією та глибинним навчанням: DeepFactors комбінує оптимізацію фактор-графу з глибокими представленнями для глибини.
- Плавний або квазістабільний рух камери: дозволяє точно накопичувати фактори, що впливають на модель.
- Відсутність динамічних об'єктів у сцені: як і інші методи на основі багатовидової оптимізації, DeepFactors припускає статичність оточення.

CNN-DVO

- Моно- або RGB відеопослідовності: CNN-DVO розрахований на використання монокулярних кадрів, з яких нейромережа попередньо оцінює карту глибини.
- Сцени з вираженою глибинною структурою: чітка перспектива, глибина та текстура допомагають у реконструкції.
- Стабільний, передбачуваний рух камери: оскільки метод включає пряме зіставлення інтенсивності пікселів, різкі зміщення зменшують точність.
- Статичне середовище: рухомі об'єкти негативно впливають на пряму візуальну одометрію.
- Обмежена освітлювальна варіативність: сильні зміни освітлення можуть знижувати точність DVO-компоненти.

РОЗДІЛ 4

РЕКОНСТРУКЦІЯ

Наданий певний набір кадрів (16 шт.):

Рис.3.1. Другий та дванадцятий кадри відповідно. Камера рухається по вільній кругоподібній траекторії.

Згідно з їхніми характеристиками (в приміщенні, середня кількість деталей) та параметрами кожного розглянутого метода, які визначають, чи підходить певний з них для нашого набору, були обрані класичний прямий напів-щільний метод LSD-SLAM та класичний непрямий розріджений метод ORB-SLAM2 як один

з найкращих та один з слабших виборів для 3D-реконструкції даних зображень, відповідно.

Даний вибір було зроблено через наступні причини (на основі також розписаних у попередньому розділі характеристик):

LSD-SLAM опрацьовує безпосередньо рівні яскравості пікселів у кадрах, що означає, що йому не потрібна наявність чітких фіч (наприклад, кутів, країв), які часто відсутні на білих стінах або у слабко текстурованих ділянках. Це робить LSD-SLAM стійкішим до малотекстурованих або "порожніх" сцен у приміщенні.

Натомість, ORB-SLAM2 працює добре тільки при наявності достатньої кількості ORB-ознак. Даний набір кадрів мають великі однорідні області — фони, стіни, неструктуровані ділянки, що призводить до поганої детекції окремих фіч, а отже — невдачі в трекінгу або падінню SLAM після декількох кадрів. ORB-SLAM2 може працювати тільки за умови сильного ручного тюнінгу (низькі пороги фіч), але це не гарантуватиме результату.

Зроблені мінімальні зміни та використані публічно доступні репозиторії описаних у роботі методів. Реалізація була виконана на робочій станції Ubuntu 14.04 в IDE Visual Studio Code. Головна зміна в коді полягає у відключенні відображення позиції камери в певному кадрі, для зручнішого відображення реконструкцій сцени.

Результат роботи:

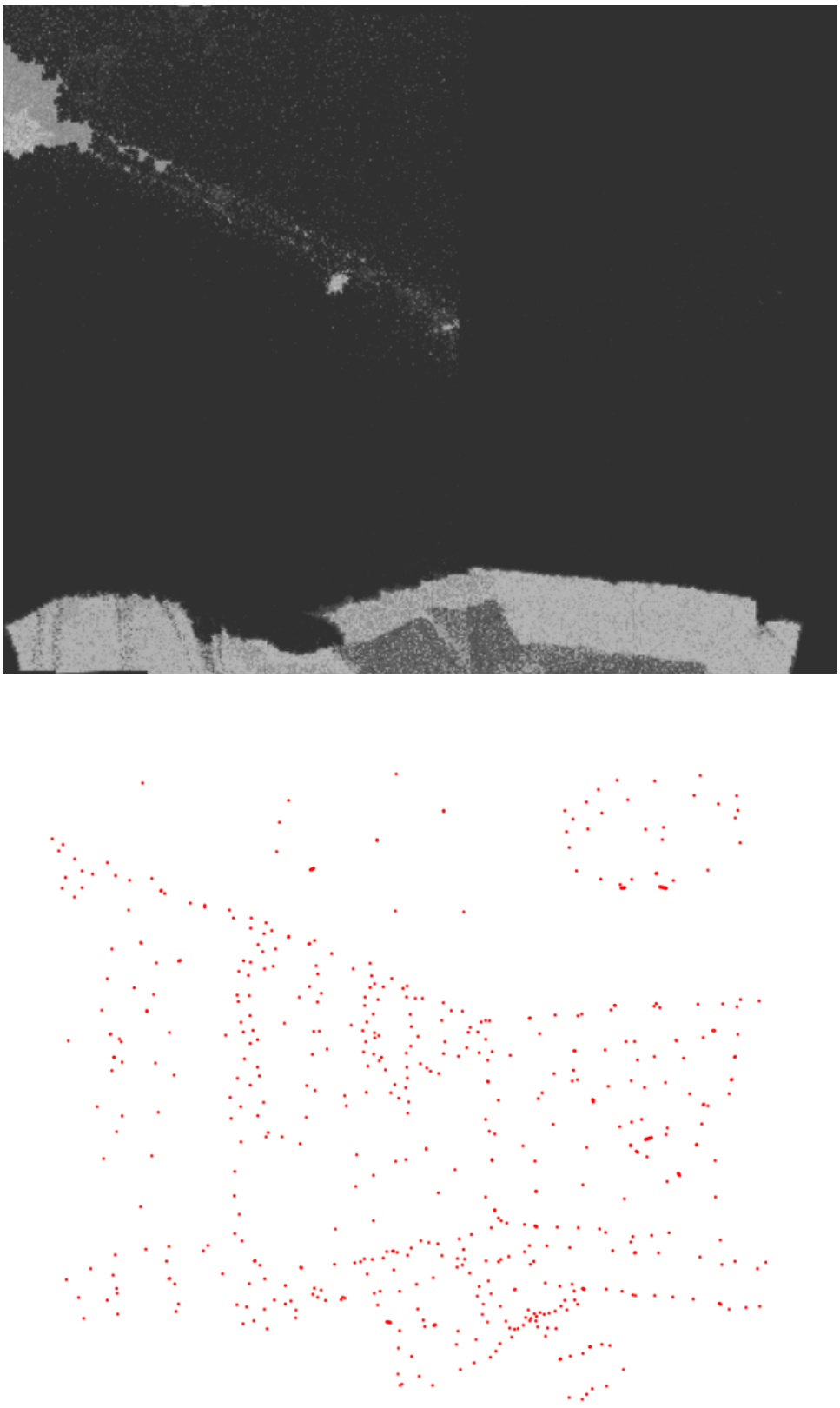

Рис. 3.2-3. Результати реконструкції за допомогою LSD-SLAM і ORB-SLAM2. Перший результат не зафіксував кут та іншу стіну, другий виявився занадто шумним через кадри з мінімальною кількістю деталей.

ВИСНОВКИ

В роботі зроблено наступне:

1. Розглянуто сучасні методи візуального SLAM, зокрема як класичні (на основі ознак або пікселів), так і засновані на методах машинного навчання.
2. Систематизовано методи за критеріями щільності карти (sparse/dense), типу обробки (direct/indirect), а також за участю глибинного навчання (classical/ML-based).
3. Проаналізовано умови ефективного використання кожного з методів, їх переваги та недоліки при застосуванні до реальних даних.
4. Запропоновано рекомендації щодо вибору методу реконструкції 3D-сцени залежно від типу даних, складності сцени та обчислювальних ресурсів.
5. Обрано два контрастні методи (LSD-SLAM, ORB-SLAM2) для подальшого порівняльного аналізу на основі наданого датасету фотографій.
6. Визначено, що LSD-SLAM є ефективним варіантом для даного випадку завдяки стійкості до слабко текстурованих сцен та можливості роботи з обмеженою кількістю кадрів.

СПИСОК ЛІТЕРАТУРИ

1. Pedrosa, E., L. Reis, C. M. D. Silva and H. S. Ferreira. Autonomous Navigation with Simultaneous Localization and Mapping in/outdoor. 2020.
2. Дослідження методів SLAM для навігації мобільного робота усередині приміщень. Досвід дослідження R2 Robotics: https://habr.com/en/articles/560856/
3. E. P. Herrera-Granda, J. C. Torres-Cantero, and D. H. Peluffo-Ordóñez, "Monocular Visual SLAM, Visual Odometry, and Structure from Motion Methods Applied to 3D Reconstruction: A Comprehensive Survey", pp. 1-56, 2024, doi: 10.1016/j.heliyon.2024.e37356
4. R. Castle, "PTAM-GPL: Parallel Tracking and Mapping," GitHub repository. GitHub, 2013. [Online]. Available: https://github.com/Oxford-PTAM/PTAM-GPL
5. R. Mur-Artal, "ORB-SLAM Monocular," GitHub repository. GitHub, 2016. [Online]. Available: https://github.com/raulmur/ORB_SLAM
6. R. Mur-Artal, "ORB-SLAM2," GitHub repository. GitHub, 2017. [Online]. Available: https://github.com/raulmur/ORB_SLAM2
7. J. D. Tardós, "ORB-SLAM3," GitHub repository. GitHub, 2021. [Online]. Available: https://github.com/UZ-SLAMLab/ORB_SLAM3
8. P. Foster, "OpenDTAM," GitHub repository. GitHub, 2016. [Online]. Available:

https://github.com/anuranbaka/OpenDTAM

9. M. Pizzoli, "REMODE," GitHub repository. GitHub, 2015. [Online]. Available: https://github.com/uzh-rpg/rpg_open_remode
10. J. Engel, "LSD-SLAM: Large-Scale Direct Monocular SLAM," GitHub repository. GitHub, 2014. [Online]. Available: https://github.com/tum-vision/lsd_slam
11. J. Zubizarreta, "DSM: Direct Sparse Mapping," GitHub repository. GitHub, 2021. [Online]. Available: https://github.com/jzubizarreta/dsm
12. R. Cheng, "CNN-DVO," McGill repository. McGill, 2020. [Online]. Available: http://www.cim.mcgill.ca/∼mrl/ran/crv2020
13. B. Bescos, "DynaSLAM," GitHub repository. GitHub, 2019. [Online]. Available: https://github.com/BertaBescos/DynaSLAM
14. A. Steenbeek, "Sparse-to-Dense: Depth Prediction from Sparse Depth Samples and a Single Image," GitHub repository. GitHub, 2022.
15. Z. Teed and J. Deng, "DeepV2D," GitHub repository. GitHub, 2020.
16. Z. Teed and J. Deng, "DROID-SLAM," GitHub repository. GitHub, 2022.
17. A. Rosinol, "NeRF-SLAM: Real-Time Dense Monocular SLAM with Neural Radiance Fields," GitHub repository. GitHub, 2022. [Online]. Available: https://github.com/ToniRV/NeRF-SLAM
18. A. Sundar, "CNN-SLAM," GitHub repository. GitHub, 2018.

19. H. Zhou, B. Ummenhofer, and T. Brox, "DeepTAM," GitHub repository. GitHub, 2019.

20. S. Troscot, "CodeSLAM," GitHub repository. GitHub, 2022.

21. J. Czarnowski and M. Kaneko, "DeepFactors," GitHub repository. GitHub, 2020.

22. R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "ORB-SLAM: A Versatile and Accurate Monocular SLAM System," IEEE Transactions on Robotics, vol. 31, no. 5, pp. 1147–1163, Oct. 2015, doi: 10.1109/TRO.2015.2463671.

23. H. Strasdat, J. M. M. Montiel, and A. J. Davison, "Scale drift-aware large scale monocular SLAM," in Robotics: Science and Systems, MIT Press Journals, 2011, pp. 73–80.

24. H. Strasdat, A. J. Davison, J. M. M. Montiel, and K. Konolige, "Double window optimisation for constant time visual SLAM," Proceedings of the IEEE International Conference on Computer Vision, pp. 2352–2359, 2011, doi: 10.1109/ICCV.2011.6126517.

25. C. Mei, G. Sibley, and P. Newman, "Closing loops without places," IEEE/RSJ 2010 International Conference on Intelligent Robots and Systems, IROS 2010 - Conference Proceedings, pp. 3738–3744, 2010, doi: 10.1109/IROS.2010.5652266.

26. R. Elvira, J. D. Tardós, and J. M. M. Montiel, "ORBSLAM-Atlas: a robust and accurate multi-map system," in 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2019, pp. 6253–6259. doi: 10.1109/IROS40897.2019.8967572.

ДОДАТОК А

Method	**Tracking method**	**Map density**	**Pixels used**	**Estimation**
MonoSLAM (2007)	Feature-based	Sparse	Shi Tomasi	EKF
PTAM (2007)	Feature-based	Sparse	Hi. grad.	BA
ORB-SLAM (2015)	Feature-based	Sparse	Hi. grad.	Local BA
ORB-SLAM2 (2017)	Feature-based	Sparse	Hi. grad.	Local BA
ORB-SLAM3 (2021)	Feature-based	Sparse	Hi. grad.	Local BA
Stühmer et al. (2010)	Direct	Dense	Hi. grad.	Cost volume refinement
DTAM (2011)	Direct	Dense	Hi. grad.	Cost volume refinement
REMODE (2014)	Direct	Dense	Hi. grad.	Bayesian estimation
LSD-SLAM (2014)	Direct	Semi-dense	Edgelets	Pose graph optimization
DSM (2020)	Direct	Sparse	Hi. grad.	GPGO

Табл. 4.1. Характеристика класичних SLAM-методів 3D-проектування (1 частина).

Method	Global optimization	Relocalization	Loop closure
MonoSLAM (2007)	-	-	-
PTAM (2007)	+	+	-
ORB-SLAM (2015)	+	+	+
ORB-SLAM2 (2017)	+	+	+
ORB-SLAM3 (2021)	+	+	+
Stühmer et al. (2010)	+	+	-
DTAM (2011)	+	+	-
REMODE (2014)	-	-	-
LSD-SLAM (2014)	+	-	+
DSM (2020)	+	-	-

Табл. 4.2. Характеристика класичних SLAM-методів 3D-проектування (2 частина).

ДОДАТОК Б

Method	**Tracking method**	**Map density**	**Pixels used**	**Estimation**
DynaSLAM (2018)	Feature-based	Sparse	Hi.grad.	Local BA
Steenbeek et al. (2022)	Feature-based	Sparse	Hi.grad.	BA
Sun et al. (2022)	Feature-based	Sparse	Hi.grad.	BA
Lee et al. (2022)	Feature-based	Sparse	Hi.grad.	BA
SVR-Net (2023)	Feature-based	Sparse	Learned features	Optimal match recurrent network
DeepV2D (2020)	Optical flow, Feature-based	Dense	Learned features	3D Stereo matching over cost volumes
DROID-SLAM (2021)	Optical flow	Dense	Learned features, Between keyframes edges	BA
SDF-SLAM (2022)	Feature-based	Dense	Learned features and descriptors	BA

Табл. 4.3. Характеристика SLAM-методів машинного навчання 3D-проектування 1 (1 частина)

NeRF-SLAM (2022)	Optical-flow	Dense	Learned features, Between keyframes edges	BA, Radiance field optimization
Rosinol et al. (2023)	Optical-flow	Dense	Learned features, Between keyframes edges	BA, Probabilistic volumetric fusion
CNN-SLAM (2017)	Direct	Semi-dense	Hi.grad.	Pose Graph optimization
DeepTAM (2018)	Direct, Optical flow	Dense	Hi. grad.	Cost volume refinement
DeepFusion (2019)	Direct	Semi-dense	Hi. grad.	Opt. framework
CodeSLAM (2018)	Direct	Dense	Hi. grad.	BA
DeepFactors (2020)	Direct	Dense	Hi. grad.	Multiview BA
CNN-DVO (2020)	Direct	Sparse	Hi. grad., Dynamic up- and downsampling	BA

Табл. 4.4. Характеристика SLAM-методів машинного навчання 3D-проектування 1 (2 частина)

Method	**CNN architecture**	**CNN's main estimation tasks**
DynaSLAM (2018)	Mask R-CNN	Instance segmentaton
Steenbeek et al. (2022)	ResNet-50, Enc.-dec.	Scale, Depth map densify
Sun et al. (2022)	ResNetXt-50, Enc.-dec.	Scale, Relative depth, Depth
Lee et al. (2022)	Enc.-dec.	Scale, Semantic segmentation, Feature refinement
SVR-Net (2023)	ScanNet	Local map, Relative pose, TSDF values
DeepV2D (2020)	Hourglass Residual Flow, Enc.-dec.	Depth, Pose, 3D stereo matching
DROID-SLAM (2021)	Residual blocks	Feature extraction, Optical flow, Estate estimation
SDF-SLAM (2022)	Enc.-dec.	Feature and descriptor extraction, Semantic segmentation
NeRF-SLAM (2022)	Residual blocks, Neural Radiance Fields	Feature extraction, Optical flow, Estate estimation
Rosinol et al. (2023)	Residual blocks	Feature extraction, Optical flow, Estate estimation
CNN-SLAM (2017)	ResNet-50, FCN	Depth, Semantic segmentation
DeepTAM (2018)	Enc.-dec.	Pose hypotheses, Optical flow, Depth, Depth refinement

Табл. 4.5. Характеристика SLAM-методів машинного навчання 3D-проектування 2 (1 частина)

DeepFusion (2019)	U-Net	Log-depth gradients and uncertainties, Scale
CodeSLAM (2018)	U-Net, Variational Enc.-dec.	Code, Compact depth
DeepFactors (2020)	U-Net, Variational Enc.-dec.	Code, Compact depth, Uncertainty
CNN-DVO (2020)	U-Net, Enc.-dec.	Depth

Табл. 4.6. Характеристика SLAM-методів машинного навчання 3D-проектування 2 (2 частина)

Method	**Global optimization**	**Relocalization**	**Loop closure**
DynaSLAM (2018)	+	+	+
Steenbeek et al. (2022)	+	+	+
Sun et al. (2022)	+	+	+
Lee et al. (2022)	+	+	+
SVR-Net (2023)	+	-	-
DeepV2D (2020)	+	-	-
DROID-SLAM (2021)	+	-	+
SDF-SLAM (2022)	+	+	+

Табл. 4.7. Характеристика SLAM-методів машинного навчання 3D-проектування 3 (1 частина)

NeRF-SLAM (2022)	+	-	+
Rosinol et al. (2023)	+	-	+
CNN-SLAM (2017)	+	-	+
DeepTAM (2018)	+	+	-
DeepFusion (2019)	+	-	-
CodeSLAM (2018)	+	-	-
DeepFactors (2020)	+	+	+
CNN-DVO (2020)	+	-	+

Табл. 4.8. Характеристика SLAM-методів машинного навчання 3D-проектування 3 (2 частина)

ДОДАТОК В

Репозиторії обох використаних методів:
https://github.com/oleksandrkuzmychov/lsd_slam_dip/tree/master
https://github.com/oleksandrkuzmychov/ORB_SLAM2_dip