МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Харківський національний університет імені В.Н.Каразіна

Факультет математики і інформатики

Кафедра теоретичної та прикладної інформатики

Кваліфікаційна робота

магістр

на тему “Аналіз тривимірних сцен на основі даних відеопотоків”

Виконав: студент 6 курсу, групи МФ-61

спеціальність 122 «Комп’ютерні науки»

освітньо-наукова програма «Інформатика»

Грульов Д. І.

Керівник: доц. Доля П. Г.

Рецензент:

ХАРКІВ - 2023

Зміст

ВСТУП

Мета роботи - дослідити можливості застосування сучасних алгоритмів відновлення відновлення тривимірних сцен з двох зображень (Two-View Structure from Motion) для відновлення просторової інформації із відеопотоку.

У даній роботі ставляться наступні задачі:

- Проаналізувати актуальні алгоритми та підходи, що допомагають відновити просторову інформацію з кадрів відео;
- Обрати сучасні алгоритми що підходять для відновлення тривимірних сцен лише із двох зображень;
- Знайти алгоритм відновлення тривимірної інформації про сцену з усіх кадрів відео, що дає прийнятну точність на відео, використаних для дослідження;

Не дивлячись на те, що існують алгоритми, що розроблені спеціально для отримання тривимірної інформації з відео, є сенс розглянути новітні алгоритми що вирішують задачу всього для двох зображень, щоб дослідити можливості застосування цих алгоритмів для аналізу відеоданих та розглянуті новітні ідеї, що можуть бути розвинуті для отримання додаткової інформації, що випливає з того, що зображення утворюють відео. Проте алгоритми, що ефективно використовують цю інформацію, часто базуються на багатократному застосуванні методів попарної обробки зображень і тому є обчислювально дуже складними,

отже потребують потужних графічних прискорювачів, та зокрема з цієї причини є непридатними для обробки у реальному часі.

Надалі будемо розглядати здебільшого відновлення мап глибини. Мапа глибини - це зображення, в якому кожному пікселю відповідає значення, що відображає його відстань від камери або іншого датчика. Відновлення мапи глибини зображення або відео може бути використане для отримання інформації про тривимірну геометрію сцени, таку як відстані між об'єктами, розміри об'єктів, або форми поверхонь.

Будемо вважати задачі відновлення 3D структури та відновлення мап глибини еквівалентними. Обидві ці задачі стосуються відтворення тривимірної інформації про сцену з двовимірних зображень або відео. Знання мапи глибини дозволяє відновити тривимірні координати об'єктів або точок в сцені, використовуючи геометричні принципи, такі як тріангуляція. І навпаки, коли тривимірна структура відома, можуть бути відновлені мапи глибини.

РОЗДІЛ 1
СУЧАСНІ ПІДХОДИ ДО ВІДНОВЛЕННЯ ПРОСТОРОВОЇ ІНФОРМАЦІЇ З ДВОВИМІРНИХ ЗОБРАЖЕНЬ

1.1 Терміни та визначення

Реконструкція 3D структури з відео — це задача оцінки просторової інформації сцени, як правило, за допомогою однієї камери. Це активна область досліджень комп’ютерного зору, у якій за останні роки спостерігається значний прогрес, і ця задача не вважається вирішеною у загальному випадку.

Структура з руху (**Structure from Motion**, **SfM**) — це техніка, яка використовується в комп’ютерному зорі для реконструкції 3D-структури

сцени або об’єкта з набору 2D-зображень або відеокадрів, знятих з різних точок зору. Метою SfM є оцінка 3D-позицій набору точок у просторі, а також пози камери (позиції та орієнтації), які зафіксували зображення або кадри. Основна ідея SfM полягає у використанні візуальної відповідності між об’єктами на різних зображеннях для оцінки 3D-положень цих об’єктів, а також пози камери. У загальній постановці, задача ставиться як відновлення з довільної кількості ракурсів, або **multi-view SfM**.

Структура із руху з двох ракурсів (two-view SfM) — це окремий випадок SfM, де 3D-структура сцени оцінюється за двома зображеннями або кадрами, знятими з різних точок зору. У SfM з двох ракурсів 3D-положення набору точок у сцені та положення камери, яка зафіксувала зображення, оцінюються шляхом тріангуляції відповідностей між двома зображеннями.

Класичний SfM - це традиційний метод вирішення задачі two-view SfM, що включає виявлення та зіставлення ознак, оцінку пози камери, 3D-точкову тріангуляцію та налаштування пучка. Це фундаментальний підхід до SfM, з якого розвинулися інкрементальні та глобальні методи SfM.

Інкрементний SfM - метод вирішення задачі SfM створює 3D-модель і послідовно оцінює пози камери, додаючи нові зображення по одному до реконструкції. Цей підхід підходить для малих і середніх сцен і може забезпечити продуктивність у реальному часі в деяких випадках. Однак він може страждати від накопичення помилки, особливо в довгих послідовностях.

Глобальний SfM (global SfM) - метод вирішення проблеми пози камери та оцінки структури більш цілісним способом, як правило, використовуючи представлення на основі графів. Він використовує надлишковість у проблемі оцінки пози для досягнення більш точних і

ефективних результатів, що робить його придатним для великомасштабних і складних сцен.

Налаштування пучка – це техніка нелінійної оптимізації, яка використовується для одночасного вдосконалення 3D-структури та поз камери. Вона мінімізує помилку повторної проекції, яка є різницею між спостережуваними точками 2D-зображення та повторно спроектованими 3D-точками, шляхом коригування 3D-точок і пози камери, щоб найкраще відповідати спостережуваним даним.

Multi-view stereo (**MVS**) — це техніка для реконструкції 3D-геометрії поверхні сцени з набору 2D-зображень, зроблених з різних точок зору. MVS не оцінює пози камери явно, а натомість зосереджується на оцінці 3D-геометрії сцени безпосередньо із зображень.

Одночасна локалізація і картографування (**Simultaneous localization and mapping**, **SLAM**) - техніка комп'ютерного зору та робототехніки, яка використовує візуальні дані, такі як зображення чи відео, для оцінки пози камери в невідомому середовищі, одночасно будуючи карту середовища. Він спрямований на створення узгодженого представлення середовища та відстеження положення та орієнтації камери в цьому середовищі в режимі реального часу.

Регуляризація 3D реконструкції - техніка, яка вводить обмеження або забезпечує дотримання певних властивостей реконструйованої моделі для покращення її якості, точності та правдоподібності, одночасно вирішуючи неоднозначності та запобігаючи переобладнанню. Це допомагає скеровувати процес оптимізації до більш правдоподібних і стабільних рішень, включаючи попередні знання, плавність або інші бажані характеристики в процес реконструкції.

Оптичний потік — схема видимого руху об'єктів у візуальній сцені, викликана відносним рухом між спостерігачем і сценою. Це поле

векторів руху, яке описує, як об’єкти на зображенні рухаються відносно спостерігача.

Візуальна одометрія - задача комп’ютерного зору та робототехніки, що полягає у визначенні позицій та орієнтацій рухомої камери шляхом аналізу кадрів відео. Візуальна одометрія є ключовим компонентом у різних програмах, таких як навігація роботів, автономні транспортні засоби та одночасна локалізація та картографування (**SLAM**), де потрібна точна оцінка пози в реальному часі.

Тріангуляція — це геометрична техніка, яка використовується в різних областях, таких як геодезія, навігація, комп’ютерне зір та фотограмметрія, для визначення положення точки або об’єкта в просторі шляхом вимірювання кутів до точки з відомих місць. Основний принцип тріангуляції полягає в тому, що за допомогою двох відомих опорних точок і кутів між цими точками та цільовою точкою можна обчислити положення цільової точки.

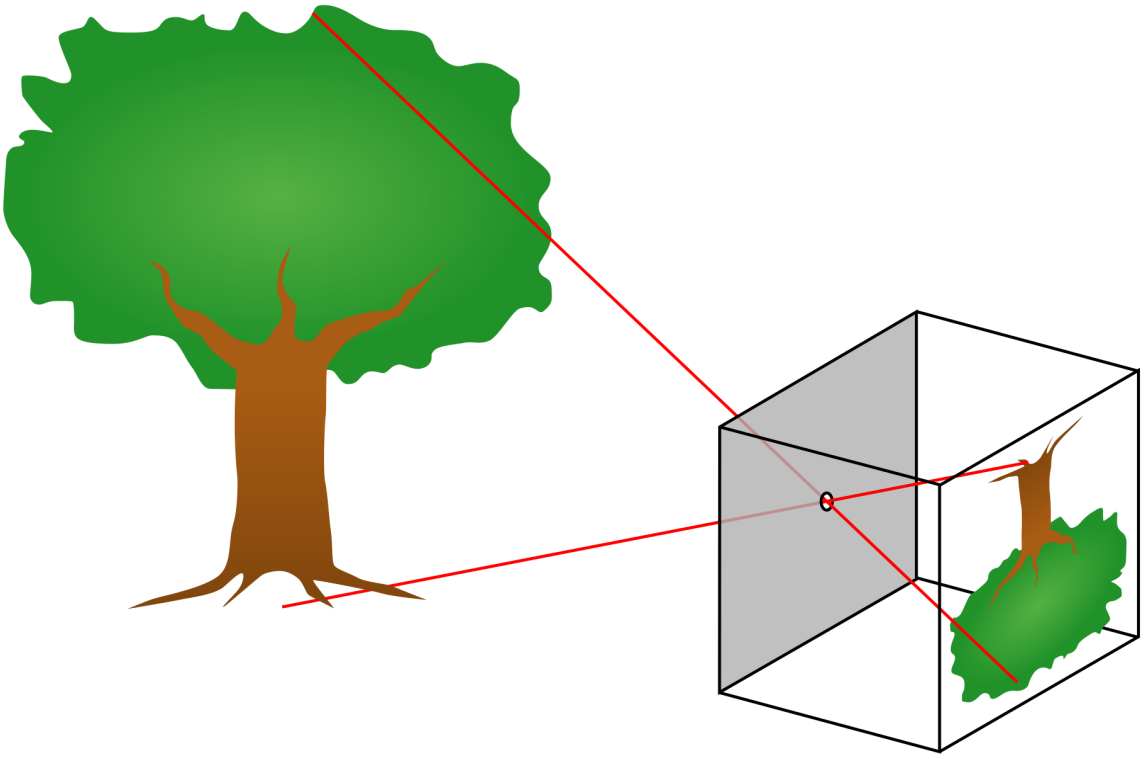

Рис. 1.1 Модель камери пінхол

Модель камери пінхол (pinhole camera model) - описує геометричний зв'язок між 3D-точками світу та їхніми 2D-проекціями на площину зображення. Він розглядає камеру як одну точку (обскуру), через яку проходять світлові промені та створюють перевернуте зображення на площині зображення позаду обскури (рис. 1.1). Ця модель включає як внутрішні, так і зовнішні параметри, такі як фокусна відстань, оптичний центр, положення та орієнтація камери у просторі.

Внутрішня матриця, яку часто позначають як K, є матрицею 3x3, яка інкапсулює внутрішні параметри камери. Ці параметри включають фокусну відстань та головну точку (c_x, c_y) (центр зображення):

$$K = \begin{pmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix}$$

Також, матриця може містити параметри спотворення. Треба зауважити Внутрішня матриця важлива для перетворення 3D координат світу в координати 2D зображення в комп'ютерному зорі та фотограмметрії.

Зовнішня матриця камери представляє зовнішні параметри камери, які описують її положення та орієнтацію в світовій системі координат. Це матриця 3x4, яка поєднує матрицю обертання (R) і вектор трансляції (t), які разом визначають перетворення твердого тіла між світовими координатами та координатами камери, та записується як $(R \mid t)$.

Площина зображення, також відома як фокальна площина або **площина проекції** - фундаментальне поняття у комп'ютерному зорі, фотографії та оптиці що означає двовимірну площину, на яку 3D-сцена проектується системою об'єктивів камери. У термінах оптики, площина

зображення розташована всередині камери, зазвичай за об’єктивом, і перпендикулярна до оптичної осі, яка є уявною лінією, що проходить через центр об’єктива. У контексті проективної геометрії камеру можна моделювати як проективне перетворення, яке відображає точки в 3D проективному просторі (світові координати) на точки в 2D проективному просторі (координати зображення) площини зображення (рис. 1.2). Це перетворення представлено проекційною матрицею 3x4, яка називається матрицею камери, яка поєднує в собі внутрішні та зовнішні параметри камери.

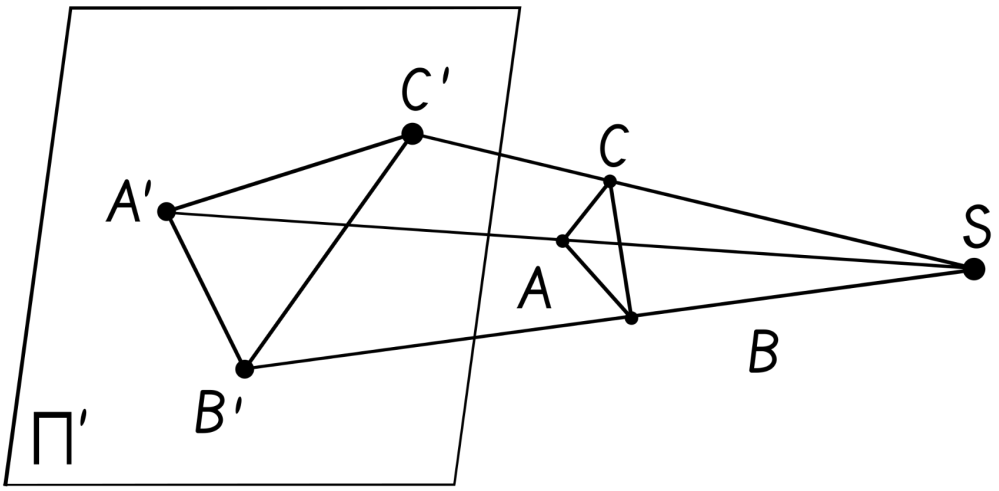

Рис. 1.2 проекція на площину

Помилка повторної проекції (**reprojection error**) — це показник, який використовується в комп’ютерному зорі та 3D-реконструкції для кількісного визначення розбіжності між спостережуваними точками 2D-зображення та відповідними 2D-точками, спроектованими назад на площину зображення з реконструйованих 3D-точок. По суті, він вимірює похибку в орієнтовній позі камери або 3D-структурі шляхом порівняння вихідних точок зображення з їх відтвореними аналогами.

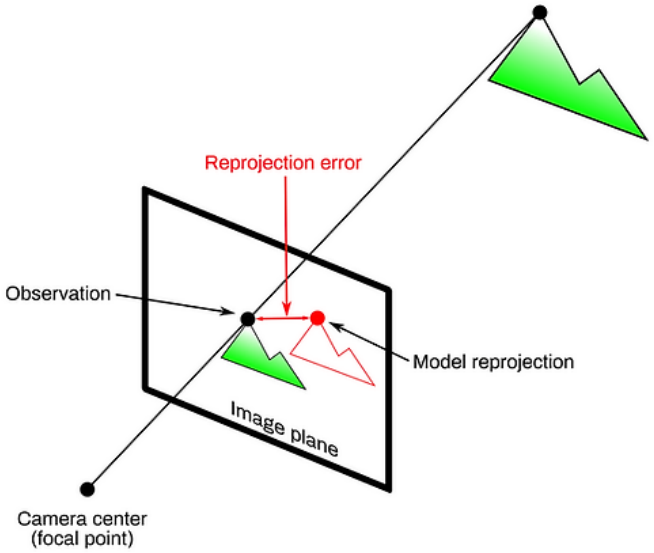

Рис. 1.3 помилка повторної проекції

RANSAC (**RAN**dom **SA**mple **C**onsensus) — це ітераційний алгоритм для надійної оцінки параметрів моделі з набору даних, що містить значну кількість викидів або шуму. Алгоритм працює шляхом повторного вибору випадкових підмножин точок даних, підгонки моделі до підмножини та ідентифікації елементів, узгоджених із моделлю. Як кінцевий результат вибирається найкраща модель із найбільшою кількістю різних значень, що ефективно мінімізує вплив викидів на оцінку параметра (рис 1.4).

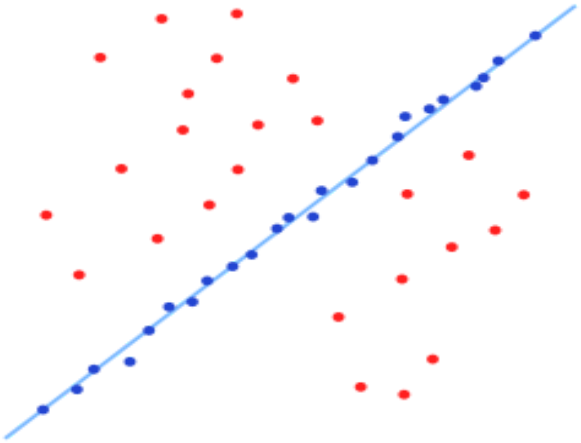

Рис. 1.4 RANSAC для підбору лінійної моделі

1.2. Загальні підходи до вирішення задачі

Задача відновлення просторових даних з відео відноситься до загальної задачі відновлення структури із руху (Structure-from-Motion, SfM).

На даний момент відомі алгоритми, що якісно відновлюють просторову інформацію з відеопотоку. Відеопоток заздалегідь містить більше інформації ніж окремі фото, зокрема послідовність, яких ці фото розташовані, і самі послідовні фото є зображеннями, що різняться дуже мало, що дозволяє, наприклад, точно знаходити щільні відповідності між точками та відстежувати тракторії об’єктів.

Проте так чи інакше, як уже було зазначено, більшість алгоритмів що аналізують просторову структуру на основі відео спираються технічно чи ідейно на попарний аналіз зображень, бо за допомогою попарної обробки кадрів, алгоритми можуть визначати відповідності між об’єктами на кадрах. Ці параметри можуть бути використані для подальшої реконструкції тривимірної моделі сцени. Але при цьому об’єм обчислень може бути неприйнятно великим для ряду застосувань.

Методи тривимірної реконструкції загалом можна розділити на класичні (традиційні) та підходи машинного навчання (ML).

Класичні методи 3D-реконструкції зазвичай спираються на геометричні принципи, такі як тріангуляція, епіполярна геометрія та калібрування камери. Ці методи часто передбачають явне моделювання геометрії та фізики процесу створення зображень і можуть вимагати ручних або напівручних кроків для виділення ознак, зіставлення та реконструкції.

З іншого боку, підходи машинного навчання (ML) для 3D-реконструкції використовують алгоритми, навчені на великих наборах даних, щоб вивчати шаблони та зв’язки між даними.

У багатьох випадках, в останніх дослідженнях класичні та засновані на машинному навчанні методи тривимірної реконструкції використовуються разом, щоб доповнити сильні та слабкі сторони першого та другого підходів. Поєднання класичного підходу та підходу на основі машинного навчання може використовувати переваги обох підходів, що призводить до підвищення точності та надійності завдань 3D-реконструкції.

Машинне навчання корисне для добування апріорної інформації, що явно не витікає з даних, що аналізуються. Методи машинного навчання можуть отримати попередні знання з даних різними способами. Наприклад, методи на основі МН можуть навчитися оцінювати глибину або 3D-структуру з великих наборів анотованих даних, що дозволяє їм фіксувати статистичні моделі та закономірності з реальних сцен: поведінку текстур, об’єктів, тощо. Інколи ці закономірності використовуються як орієнтири групування пікселів, тобто ознаки більш високого рівня.

Важливим класичним методом, що заслуговує уваги, є епіполярна геометрія, також відома як бінокулярна геометрія, — це область комп’ютерного зору та обробки зображень, яка вивчає геометричні співвідношення між двома видами або зображеннями однієї сцени, зробленими з різних точок зору або камер. Не дивлячись на те, що існують методи, що застосовують ідею епіполярної геометрії спеціально для аналізу багатьох зображень, типовим випадком є попарне застосування класичної інтерпретації методу. Тому важливим є він також для розуміння, що задачі відновлення 3D структури з багатьох зображень, часто зводяться до попарного аналізу.

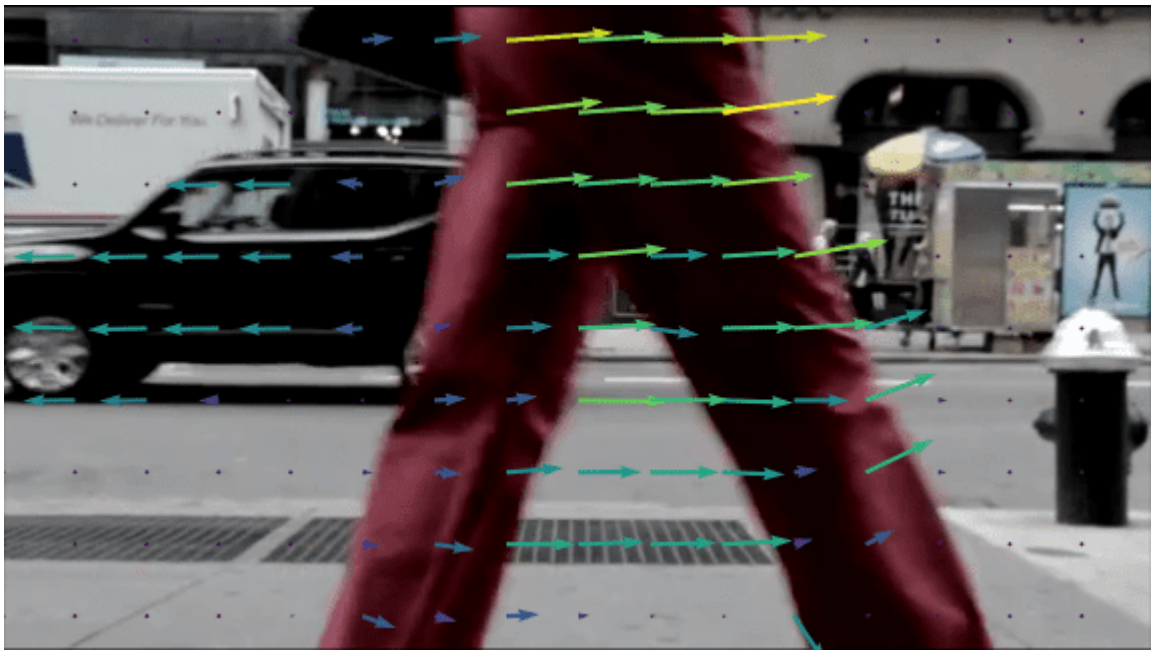

Рис. 1.5 Оптичний потік відображає рух об'єктів

Іншим ключовим підходом є оцінка оптичного потоку. Оптичний потік спирається на шаблони видимого руху об'єктів на зображенні чи відеопослідовності. Математично, це векторне поле, яке у кожній своїй точці описує зміщення пікселів між послідовними кадрами у відео або між послідовними зображеннями в послідовності. Оптичний потік надає інформацію про відносний рух об'єктів у сцені, включаючи напрямок і величину їхнього руху. Якщо сцена статична, можна вважати що об'єкти рухаються відносно камери. Оптичний потік також є методом, що призначений для обробки двох зображень. Один із новітніх алгоритмів реконструкції 3D моделі з відео використовує саме попарну оригінальну інтерпретацію [2].

Оптичний потік можна оцінити за допомогою різних методів, включаючи традиційні, засновані на градієнтах зображення, припущеннях сталості яскравості та ітераційних методах оптимізації, а також сучасних методах на основі глибокого навчання, які використовують згорткові або рекурентні нейронні мережі.

1.3 Останні досягнення в області

За останні два десятиліття сфера 3D-реконструкції з відеокадрів значно розвинулась завдяки прогресу як у традиційних методах вилучення ознак, так і в сучасних методах глибокого навчання. На початку 2000-х років впровадження надійних алгоритмів виділення ознак, таких як Масштабно-інваріантне перетворення ознак (SIFT), забезпечило основу для вилучення та зіставлення ключових точок між зображеннями, забезпечуючи точнішу та надійнішу 3D-реконструкцію. Поряд із SIFT, були розроблені інші методи вилучення ознак, такі як SURF, ORB, BRISK і AKAZE, які ще більше вдосконалили процес реконструкції на основі ознак у структурі з руху (SfM) та MVS.

У міру розвитку галузі поява глибокого навчання та його застосування до задач комп’ютерного зору призвело до зміни парадигми 3D-реконструкції. Згорткові нейронні мережі (CNN) продемонстрували свою здатність вивчати потужні представлення функцій безпосередньо з даних, відкриваючи шлях до методів реконструкції на основі наскрізного навчання. Включення методів навчання з учителем та без учителя дозволило оцінювати глибину та рух камери з відеопослідовностей без потреби в явній інформації про істинну глибину, що ще більше просунуло найсучасніші 3D-реконструкції. Останні розробки демонструють потенціал глибокого навчання для вирішення складних сценаріїв реконструкції та продовжують розширювати межі можливого в цій галузі.

У 2020 році розроблений алгоритм, що реконструює мапу глибини для кожного кадру відео [1]. Алгоритм показує якісні та стабільні результати застосовано до відео що знімають довільні сцени, проте, наприклад, у сенсі реконструювання динамічних сцен, спеціалізований до оцінки руху людей, що зазначається у статті. Не зважаючи на те, що

перевагою алгоритму висувається виключна ступінь узгодженості реконструкції між кадрами, алгоритм використовує попарну обробку кадрів, що вибираються за описаним принципом, як базовий етап навчання моделі. На цьому етапі використовується оптичний потік, що потрібен для того, щоб валідувати узгодженість реконструкцій, що були отримані для двох різних моментів часу, що відповідають обраним кадрам. Помилка використовується для ітерації навчання мережі.

Також алгоритм у своїй найпершій стадії спирається на COLMAP - програмне забезпечення загального призначення для вирішення задач SfM і Multi-View Stereo (MVS) що пропонує широкий спектр функцій для реконструкції впорядкованих і невпорядкованих колекцій зображень. Цей пакет ПЗ реалізує класичні алгоритми, тому дозволяє отримувати лише розріджені реконструкції, що використовуються як опорна інформація. Також COLMAP та подібні системи можуть використовуватися для оцінки позицій та орієнтацій камери для кожного кадру відео, таким чином розв’язуючи задачу візуальної одометрії.

Візуальна одометрія в реальному часі досягла значного прогресу за останні роки, і тепер існує кілька методів, здатних забезпечити точну та ефективну оцінку руху камери в реальному часі. Однак важливо зазначити, що продуктивність цих методів може змінюватися залежно від конкретної програми, апаратного забезпечення та умов навколишнього середовища.

В останні роки розробка застосуваь візуальної одометрії на основі засобів, таких як ORB-SLAM [6] і DSO (пряма розріджена одометрія), продемонструвала здатність забезпечити оцінку руху камери в реальному часі з високою точністю. Ці методи використовують ефективні методи вилучення функцій, зіставлення та оптимізації для досягнення продуктивності в реальному часі на сучасному обладнанні. ORB-SLAM - це алгоритм та програмне забезпечення, що активно доповнюється, розробляється та покращується. На даний момент розроблена вже третя

версія цього - алгоритму ORB-SLAM3 [7]. Останній алгоритм переважає найпершу версію у точності, а також, за твердженням авторів, є швидшим на декілька порядків.

Сучасні методи глибокого навчання також дозволили побудувати алгоритми візуальної одометрії, що дають точні та надійні результати та на даний момент наближаються до готовності використання у реальних умовах. Одним із новітніх алгоритмів, що заслуговує уваги, є Deep Patch Odometry [5], дослідники Прінстонського університету оприлюднений у 2022 році. В основі даного алгоритму лежить розбиття кадрів відео на клаптики (patches) за допомогою однієї нейронної мережі та відстежування їх руху за допомогою іншої, рекурентної нейронної мережі.

Повертаючись до задачі відновлення тривимірних сцен з двох зображень, в 2021 та ж сама група дослідників оприлюднила нейромережеву модель RAFT-Stereo, що вирішує цю задачу, та на момент 2022 року посідає друге місце у рейтингу RVC Stereo [3]. Цікаво, що алгоритм є модифікацією нейромережі для знаходження оптичного потоку RAFT (Recurrent All-Pairs Field Transforms). У 2022 році іншими дослідниками був також винайдений алгоритм CREStereo, що показує ще кращий результат, може давати, згідно з авторами статті та рейтингом RVC, ще точніші реконструкції [4].

Проте, треба зазначити, що алгоритми, які базуються на машинному навчанні, сильно залежать від вибірки, на якій були навчені. Більш того, такі підходи часто мають погану узагальнювальну здібність між вибірками, тому на практиці потребують навчання під конкретні умови, для повного охоплення яких потрібні надто великі вибірки. Тому, хоча такий алгоритм і буде достатньо універсальним, адаптація його під конкретні умови буде потребувати дуже серйозного об’єму обчислень без жодної гарантії на прийнятний результат в умовах, які можуть лише незначно відрізнятися від тих, до яких він був адаптований.

Дещо інший сценарій - коли модель машинного навчання навчається на не тільки на навчальних даних, а ще коригується у процесі її застосування. Саме цей підхід використовується у алгоритмі узгодженої оцінки мап глибини з відеопотоку [1], що зазначений у цьому підрозділі. Він використовує підхід під назвою test-time training. Test-time training (навчання під час тестування), також відоме як адаптація під час тестування або “тонке налаштування” (fine-tuning), — це техніка, яка використовується в машинному навчанні для адаптації попередньо навченої моделі, щоб вона краще відповідала певному тестовому набору даних або набору зразків під час фази висновку. Ідея полягає в тому, щоб зробити незначні коригування параметрів моделі, щоб вона краще працювала на даних тестових даних.

Таким чином, хоча монокулярна реконструкція глибини відео досягла значного прогресу та застосовувалася до різних програм, таких як робототехніка, доповнена реальність та автономне водіння, це ще не повністю вирішена проблема та залишається активною областю досліджень із постійними викликами та досягненнями.

РОЗДІЛ 2
РОЗВИТОК АЛГОРИТМІВ ДВОРАКУРСНОЇ РЕКОНСТРУКЦІЇ: ВІД ПРОЕКТИВНОЇ ГЕОМЕТРІЇ ДО ГЛИБОКОГО НАВЧАННЯ

2.1 Класичний підхід до відновлення просторової структури з декількох ракурсів

Реконструкцію 3D-сцени з відеопотоку можна розглядати як окремий випадок багаторакурсної реконструкції. В обох випадках метою є оцінка 3D-структури сцени з кількох двовимірних зображень. У реконструкції з декількох ракурсів 3D-структура оцінюється з набору

2D-зображень, знятих з різних точок зору, тоді як у реконструкції з відео 3D-структура оцінюється з послідовності 2D-зображень, знятих з однієї рухомої точки огляду.

Основна відмінність між цими двома підходами полягає в тому, що при багаторакурсній реконструкції камери зазвичай нерухомі або керовано рухаються, тоді як при реконструкції з відео камера рухається вільно, і рух камери потрібно оцінювати як частину процесу реконструкції.

Реконструкція з відео зазвичай передбачає оцінку руху камери, а потім використання зображень для оцінки 3D-структури сцени. Це можна зробити за допомогою таких методів, як оптичний потік, відстеження функцій і коригування пакетів, які також використовуються в багаторакурсній реконструкції.

Дещо більш загальна задача, ніж багаторакурсна реконструкція, структура з руху (SfM) — це класична задача комп'ютерного зору, яка крім оцінки 3D-структури сцени, передбачає знаходження позицій та орієнтацій камери для кожного з зображень. З цього випливає, що вирішуючи задачу SfM, ми також вирішуємо задачу візуальної одометрії.

SfM є фундаментальною темою дослідження комп'ютерного зору протягом кількох десятиліть і заклала основу для багатьох пов'язаних методів, таких як мультиракурсне стерео, візуальний SLAM і фотограмметрія.

За своєю суттю SfM має на меті відновити 3D-геометрію сцени шляхом аналізу руху камери та спостережуваних змін зовнішнього вигляду сцени на декількох зображеннях. Задача за своєю суттю є складною через втрату інформації про глибину, коли 3D-сцена проектується на площину 2D-зображення.

SfM особливо корисний для реконструкції відео, оскільки він оцінює як рух камери, так і тривимірну структуру сцени, що робить його добре придатним для обробки послідовностей зображень, знятих рухомою

камерою, як у випадку відео. SfM пропонує більш комплексне рішення ніж MVS, оскільки воно не тільки відновлює 3D-структуру, але й оцінює рух камери, що є важливим для обробки відеоданих. Після того, як пози камери та розріджена 3D-структура оцінені за допомогою SfM, можна застосувати методи MVS для створення більш детальної та щільної 3D-реконструкції сцени, використовуючи оцінені пози камери з процесу SfM. Традиційний підхід до SfM зазвичай включає наступні кроки:

- Виявлення ознак і зіставлення: виявлення ключових точок у кожному зображенні та зіставлення їх між кількома зображеннями для встановлення відповідності.
- Оцінка фундаментальної або істотної матриці: ця матриця фіксує геометричні співвідношення між парами зображень і може бути оцінена за допомогою таких алгоритмів, як 8-точковий алгоритм, або його більш просунута версія - 5-точковий алгоритм.
- Оцінка пози камери: відновлення відносного обертання та переміщення між позами камери на основі основної матриці.
- Тріангуляція: реконструкція тривимірної структури сцени шляхом тріангуляції відповідних ключових точок за допомогою орієнтовних поз камери.
- Налаштування пучків: техніка нелінійної оптимізації, яка використовується для одночасного вдосконалення 3D-структури та поз камери шляхом мінімізації помилки повторного проектування між спостережуваними точками 2D-зображення та повторно спроектованими 3D-точками.

Описана схема описує застосування геометрії двох ракурсів для реконструкції сцени. Класичний SfM в основному зосереджувався на геометрії двох ракурсів і заклав основу для більш просунутих підходів, таких як інкрементний і глобальний SfM, який може працювати з кількома видами та реконструювати великомасштабні сцени. Розвиток SfM також

сприяв прогресу в інших галузях, таких як робототехніка і доповнена реальність.

На задачу SfM можна дивитися більш загально, а саме як на реконструкцію з N зображень по L відповідним точкам на кожному зображенні. Тоді розглядається система рівнянь, що описує зв'язок між проекціями точок на зображеннях та їх просторовими координатами:

$$\begin{pmatrix} x \\ y \end{pmatrix} = k \left\{ R \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} + t \right\} \quad (1)$$

де u, v - координати точок, k - коефіцієнт масштабування, R - матриця повороту, t - вектор трансляції.

Координати *(u, v)* слід розуміти як однорідні координати *(u, v, 1)*. Ці координати описують з одного боку, координати пікселя на зображенні, а з іншого - координати проекції точки, що представляє піксель на площині зображення у системі координат камери. Площина зображення у системі координат камери має Z координату, рівну 1. Таким чином коефіцієнт масштабування k відображає віддаленість точки від площини зображення. Для кожної відповідності точки площини зображення, або іншими словами - пікселя зображення *(u, v)* точці у просторі *(X, Y, Z)* коефіцієнт *k* різний.

Більш загальний запис цього рівняння, через просторові однорідні координати та враховуючи матрицю калібровки *K* виглядає наступним чином:

$$z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = K \begin{bmatrix} R & T \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} = M \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix}$$

Для того, щоб скласти систему рівнянь, що можливо буде вирішити і знайти R та t для кожного ракурсу, треба підібрати деяку кількість точок

(X, Y, Z), щоб кількість рівнянь у системі стала більше або дорівнювала кількості невідомих.

Аналізуючи рівняння (1), можна побачити, що таке рівняння вводить 9 невідомих. Матриця орієнтації камери *R* представляє насправді 3 координати, це кути повороту відносно осей координат. Вектор переміщення представляє 3 координати, і нарешті точка у тривимірному просторі представляє 3 координати. Коефіцієнт *k* є залежною змінною, так як знаючи тривимірні координати точки, матрицю орієнтації та вектор переміщення можна знайти коефіцієнт масштабування.

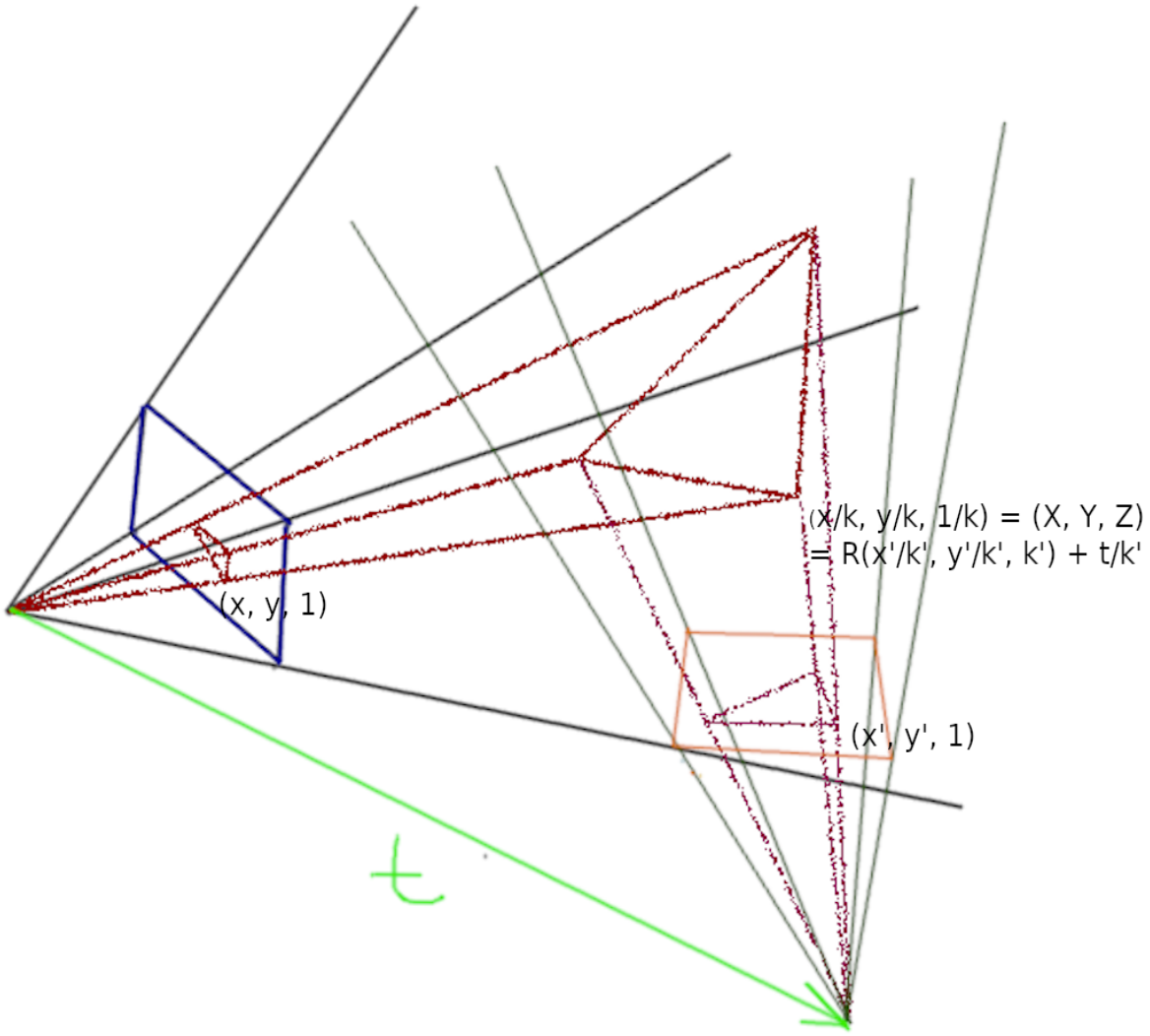

Рис. 2.1 Проекції об'єкту на двох зображеннях

Для того, щоб скласти систему рівнянь, як зазначалося вище, треба підібрати точки (u_i, v_i), $i = 1 \ldots N$, такі, що представляють проекції точок *(X, Y, Z)* на площини зображень кожного з *N* ракурсів. Тому для кожної точки *(X, Y, Z)* можна скласти $2 * N$ рівнянь, що зв'язують координати пікселів (u_i, v_i) та тривимірні координати точки, використовуючи рівняння (1).

Треба зауважити, що хоча для кожного ракурсу є матриця *R*, що описує поворот камери для ракурса відносно деякої світової системи координат, можна вважати, що одна із камер має одиничну матрицю *R = I*, або ж іншою мовою, розглядати матриці орієнтації інших ракурсів відносно одного обраного. Також, органічним кроком далі є елімінація вектору переміщення для цього ракурсу і встановлення його рівним нулю. Після цього можна вважати, що положення камери для обраного зображення і представляє світову систему координат.

Описаний вище прийом дозволяє вилучити 6 невідомих що представляють 6 ступенів свободи камери у просторі і таким чином скоротити кількість рівнянь, потрібних для того щоб система мала певний розв'язок. Це також дозволяє значно спростити систему рівнянь, особливо коли кількість ракурсів невелика.

Після такого спрощення виходить наступна система рівнянь:

$$\begin{pmatrix} u_{0j} \\ v_{0j} \end{pmatrix} = k_{0j} \begin{pmatrix} X_j \\ Y_j \\ Z_j \end{pmatrix}$$

$$\begin{pmatrix} x_{ij} \\ y_{ij} \end{pmatrix} = k_{ij} \left\{ R_i \begin{pmatrix} X_j \\ Y_j \\ Z_j \end{pmatrix} + t_i \right\} \qquad (2)$$

$$i = 0..N-1,\ j = 1..M$$

У даній системі 3*M+6*(N-1) незалежних змінних та 3*N*M рівнянь. Таким чином, для двох ракурсів, теоретично потрібно всього 2 пари відповідних точки на зображеннях, для того щоб відновити положення першої камери відносно іншої.

Однак, треба підкреслити, що це теоретичний результат, і на практиці зазвичай використовують більше відповідних точок та більш надійні методи, такі як RANSAC, щоб покращити точність і надійність рішення. Крім того, реальні додатки часто мають зашумлені дані, що може значно вплинути на точність рішення при роботі з мінімальною кількістю відповідностей.

Більш гнучкий підхід, що є загальноприйнятим, це замість прямого використання рівняння перспективної проекції, використовувати поняття фундаментальної та/або істотної матриць. Істотна матриця описує зв'язок відповідними точками на двох зображеннях:

$$(y_1)^T E y_0 = 0$$

де y_0 та y_1 є однорідними нормалізованими координатами проекціями тривімірної точки на зображеннях 1 і 2 відповідно.

У термінах системи рівнянь (1), зв'язок між координатами відповідних пікселів, що є проекціями деякої тривимірної точки, виражається наступним чином:

$$\begin{pmatrix} u_1 & v_1 & 1 \end{pmatrix} \begin{pmatrix} e_{11} & e_{12} & e_{13} \\ e_{21} & e_{22} & e_{23} \\ e_{31} & e_{32} & e_{33} \end{pmatrix} \begin{pmatrix} u_0 \\ v_0 \\ 1 \end{pmatrix} = 0 \qquad (3)$$

Як можна бачити, підхід, що передбачає використання поняття істотної матриці, позбавляє необхідності явно розглядати та знаходити тривимірні координати точок, що були співставлені на зображеннях. Крім того, даний підхід випливає з безпосередніх геометричних міркувань і тому має певний геометричний сенс.

Насправді, істотна матриця беспосередньо виражається у термінах зовнішньої матриці камери, а саме через її компоненти R та t:

$$\mathbf{E} = \mathbf{R}\,[\mathbf{t}]_\times$$

При перемноженні матриць, що задіяні у рівнянні (3), отримуємо лінійне рівняння з 9 змінними:

$$u_1u_0e_{11} + u_1v_0e_{12} + u_1e_{13} + v_1u_0e_{21} + v_1v_0e_{22} + v_1e_{23} + u_0e_{31} + v_1e_{32} + e_{33} = 0$$

Позначимо відповідні точки на зображеннях як y_i та y_i', тоді рівняння буде записуватися так:

$$u_i'u_ie_{11} + u_i'v_ie_{12} + u_i'e_{13} + v_i'u_ie_{21} + v_i'v_ie_{22} + v_i'e_{23} + u_ie_{31} + v_i'e_{32} + e_{33} = 0 \quad (4)$$

У такому випадку, $\left\{e_{ij} \mid i = 1,2,3;\ j = 1,2,3\right\}$ можна розглядати як лінійний базис, тоді координати вектора, що описує рівняння, мають наступний вигляд:

$$\tilde{y} = \begin{pmatrix} u_i'u_i \\ u_i'v_i \\ u_i' \\ v_i'u_i \\ v_i'v_i \\ v_i' \\ u_i \\ v_i' \\ 1 \end{pmatrix} \quad (5)$$

Класичний восьмиточковий алгоритм, що був зазначений на початку даного підрозділу, спирається саме на таку математичну інтерпретацію. Як можна здогадатися, кількість відповідних точок на парі зображень що обираються, безпосередньо залежить від розмірності цього вектора, саме тому, що це число диктується кількістю лінійно незалежних векторів виду (5), що можна утворити.

Істотна матриця E, а отже, і вектор розв'язку f визначено лише до невідомого масштабу. Конструктивний підхід, що показує, що 8 точок необхідно і достатньо для того, щоб скласти лінійну систему, що буде мати єдиний розв'язок, полягає у тому, щоб покласти $e_{33} = 1$, оскільки це рівняння однорідне за коефіцієнтами E.

$$\begin{pmatrix} u_1u'_1 & u_1v'_1 & u_1 & v_1u'_1 & v_1v'_1 & v_1 & u'_1 & v'_1 \\ u_2u'_2 & u_2v'_2 & u_2 & v_2u'_2 & v_2v'_2 & v_2 & u'_2 & v'_2 \\ u_3u'_3 & u_3v'_3 & u_3 & v_3u'_3 & v_3v'_3 & v_3 & u'_3 & v'_3 \\ u_4u'_4 & u_4v'_4 & u_4 & v_4u'_4 & v_4v'_4 & v_4 & u'_4 & v'_4 \\ u_5u'_5 & u_5v'_5 & u_5 & v_5u'_5 & v_5v'_5 & v_5 & u'_5 & v'_5 \\ u_6u'_6 & u_6v'_6 & u_6 & v_6u'_6 & v_6v'_6 & v_6 & u'_6 & v'_6 \\ u_7u'_7 & u_7v'_7 & u_7 & v_7u'_7 & v_7v'_7 & v_7 & u'_7 & v'_7 \\ u_8u'_8 & u_8v'_8 & u_8 & v_8u'_8 & v_8v'_8 & v_8 & u'_8 & v'_8 \end{pmatrix} \begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \end{pmatrix} = - \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

Таким чином, ми отримуємо лінійну систему з 8 невідомими, що складається з 8 рівнянь, і тому має єдиний розв'язок. У випадку, якщо більше восьми відповідних точок, загальний підхід до вирішення задачі полягає в тому, щоб описати його як загальну проблему найменших квадратів який мінімізується за умови, що $||e|| = 1$.

Очевидно, що цей підхід обмежує реконструкцію двома кадрами, якщо його намагатися застосувати до відео. Можна записати рівняння, яке зв'язує відповідні точки на трьох зображеннях наступним чином:

$$\left((y_1)^{\mathrm{T}} E_{01} y_0 \right)^{\mathrm{T}} E_{12} y_2 = 0$$

Проте, обмеження вищого порядку, що включають кілька основних матриць, можуть ускладнити задачу, що може призвести до значного збільшення обчислень. Також, використання істотних матриць попарно, ефективно є послідовним зв'язуванням обмежень, що може бути не оптимальним для включення інформації з усіх ракурсів одночасно.

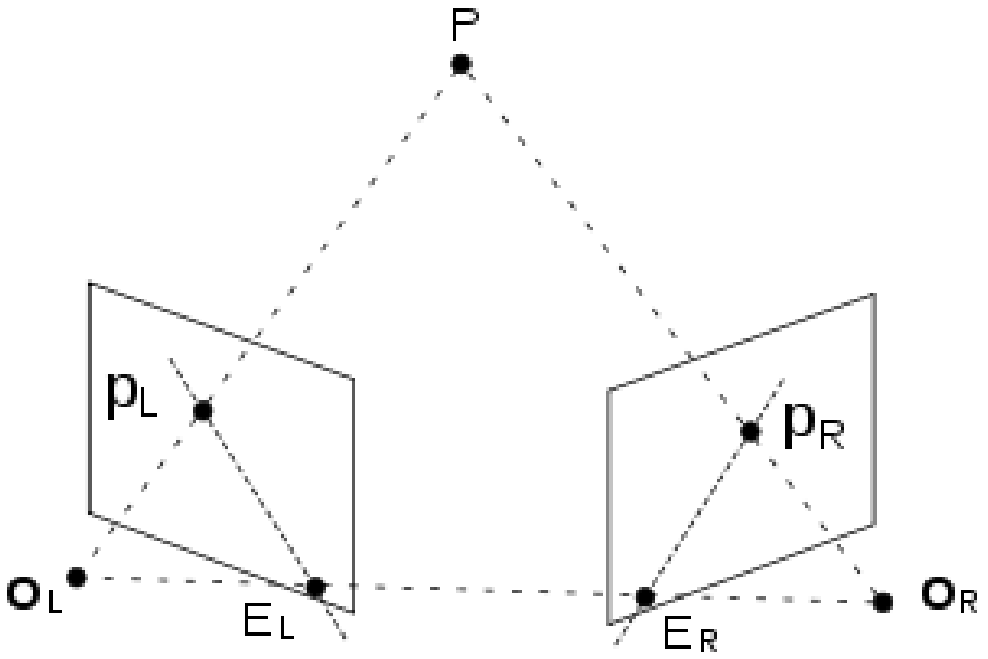

Рис. 2.2 Геометрія двох видів

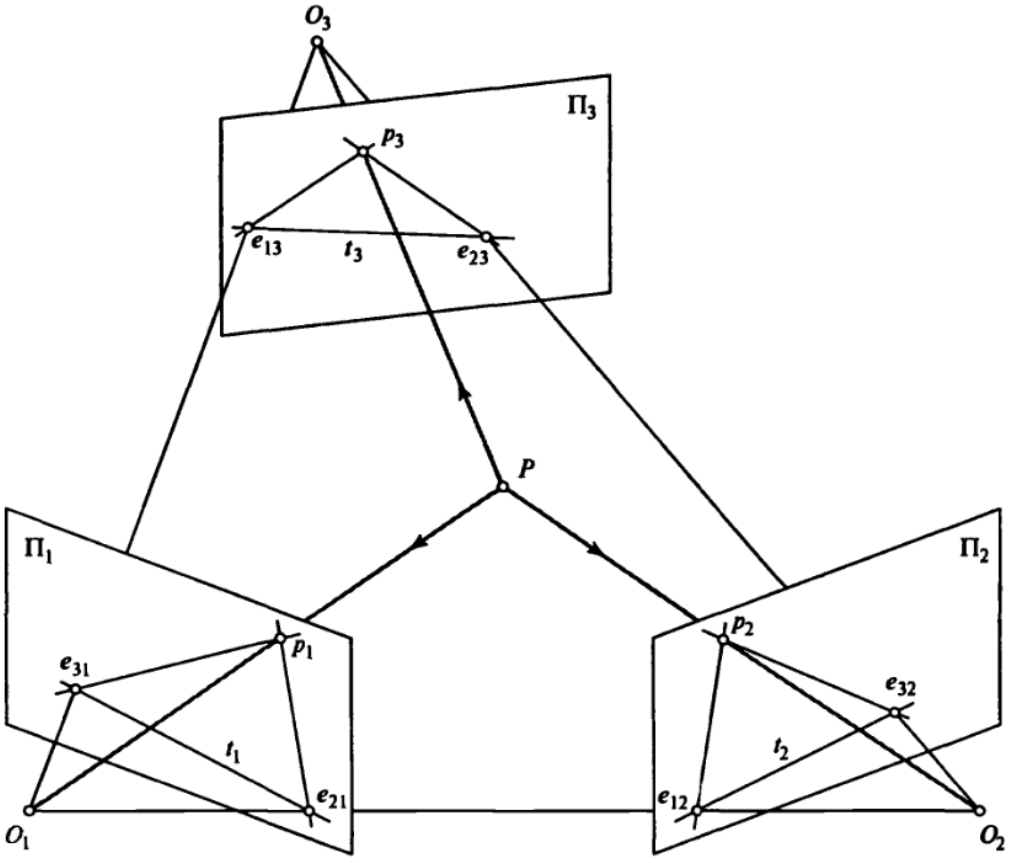

Рис. 2.3 Геометрія трьох видів

Замість цього, використовується трифокальний тензор, що є є розширенням основної матриці для трьох видів. Це матриця 3x3x3, яка

кодує геометричні співвідношення між точковими відповідностями в трьох зображеннях [9]. Трифокальний тензор використовується в багаторакурсних задачах комп'ютерного зору, що включають три ракурси, оскільки він інкапсулює геометричні зв'язки між відповідними точками на трьох зображеннях більш прямо та компактно, ніж використання попарних основних матриць.

Багаторакурсні тензори є подальшим узагальненням трифокального тензора для роботи з більш ніж трьома видами. Вони кодують геометричні зв'язки між відповідністю точок у кількох зображеннях. Однак із збільшенням кількості переглядів оцінка багаторакурсних тензорів стає складнішою та обчислювально дорожчою.

На практиці більшість алгоритмів багаторакурсної реконструкції використовують інкрементальний підхід, коли вони оцінюють попарні істотні матриці або трифокальні тензори, а потім об'єднують їх разом, використовуючи такі методи, як налаштування пучка. Глобальні методи, такі як алгоритми на основі факторизації, також можна використовувати для багаторакурсної реконструкції, але вони, як правило, не покладаються на багаторакурсні тензори.

У міру додання кадрів до реконструкції стає критично важливим уточнювати початкові оцінки поз камери та 3D-точок, щоб мінімізувати помилки та невідповідності, які виникають під час процесу попарної оцінки. Щоб досягти цього уточнення, використовується метод глобальної оптимізації, який називається налаштування пучка. Налаштування пучка уточнює оцінки поз камери та 3D-точок, мінімізуючи помилку повторної проекції для всіх видів і 3D-точок.

У процесі поетапної реконструкції групове коригування зазвичай застосовується після об'єднання попарних оцінок поз камери та 3D-точок. Це можна робити через регулярні проміжки часу, наприклад, після додавання нового кадру в систему або після обробки попередньо

визначеної кількості кадрів. Уточнені оцінки з коригування групи потім використовуються як вхідні дані для наступних кроків поетапної реконструкції, забезпечуючи більш точне та узгоджене представлення сцени.

Існує достатньо програмного забезпечення, що реалізує традиційні підходи до задач SfM та MVS. Особливої уваги заслуговує COLMAP — відкрита бібліотека та набір програмного забезпечення для 3D реконструкції з набору зображень за допомогою Structure-from-Motion (SfM) та Multi-View Stereo (MVS). Назва "COLMAP" означає "COncurrent Library for Modern Appearance Modeling". Розроблений Йоганнесом Л. Шенбергером, COLMAP надає потужний та гнучкий набір інструментів для виконання 3D реконструкції від початку і до кінця, починаючи від видобутку особливостей та співставлення до щільної реконструкції та злиття хмари точок.

Інша альтернатива - OpenMVG (Open Multiple View Geometry) — бібліотека з відкритим вихідним кодом для комп’ютерного зору, яка зосереджена на розв’язанні проблем геометрії кількох зображень, наприклад 3D-реконструкції з кількох зображень. OpenMVG, розроблений П’єром Мулоном та його співавторами, надає повний набір алгоритмів та інструментів для виконання SfM, MVS та інших пов’язаних завдань. OpenMVG надає надійні методи оцінки, такі як RANSAC, для обробки викидів і шуму у вхідних даних.

2.2 Традиційний підхід до обчислення оптичного потоку

Традиційний підхід до обчислення оптичного потоку базується на припущенні, що яскравість або інтенсивність рухомого об’єкта на зображенні залишається незмінною з часом. Це припущення відоме як обмеження сталості яскравості. Метою обчислення оптичного потоку є

оцінка руху пікселів між двома послідовними кадрами в послідовності зображень.

Одним із найперших і найвідоміших методів обчислення оптичного потоку є метод Лукаса-Канаде, розроблений Брюсом Д. Лукасом і Такео Канаде в 1981 році. Цей метод базується на таких ключових етапах, як обчислення градієнтів, формулювання рівняння оптичного потоку та його розв’язання.

Обчислення градієнтів зображення означає обчислення просторових та часові градієнти (похідні) інтенсивності зображення відносно горизонтального (x), вертикального (y) і часового (t) вимірів. Ці градієнти обчислюються за допомогою кінцево-різницевих наближень або інших відповідних методів, таких як похідні Гауса. Розглянемо таке кінцево-різнецеве наближення, виглядає наступним чином:

$$I(x, y, t) = I(x + \Delta x, y + \Delta y, t + \Delta t) \quad (6)$$

Формулювання рівняння оптичного потоку базується на тому, що обмеження сталості яскравості можна виразити диференціальним рівнянням у частинних похідних, яке називається рівнянням оптичного потоку, яке пов’язує просторові та часові градієнти зображення з горизонтальними та вертикальними компонентами оптичного потоку (u, v).

Після цього кінцево-різницевий запис розписується у термінах градієнтів:

$$I(x + \Delta x, y + \Delta y, t + \Delta t) = I(x, y, t) + \frac{\partial I}{\partial x}\Delta x + \frac{\partial I}{\partial y}\Delta y + \frac{\partial I}{\partial t}\Delta t \cdot \quad (7)$$

З рівнянь (6) та (7) випливає рівняння оптичного потоку у градієнтах:

$$\frac{\partial I}{\partial x}V_x + \frac{\partial I}{\partial y}V_y + \frac{\partial I}{\partial t} = 0$$

що коротко записується, як

$$I_x V_x + I_y V_y = -I_t$$

де I_x,I_y та I_t представляють просторові та часові градієнти зображення, а V_xта V_y є горизонтальними та вертикальними компонентами оптичного потоку.

Далі треба розв’язати рівняння оптичного потоку. Рівняння оптичного потоку є недовизначеною системою, тобто невідомих (u і v) більше, ніж рівнянь. Щоб вирішити цю проблему, метод Лукаса-Канаде припускає, що оптичний потік локально постійний у маленькому вікні навколо кожного пікселя. Це припущення дозволяє агрегувати численні рівняння у вікні для формування надвизначеної лінійної системи, тобто такої, у якої рівнянь більше ніж невідомих. Потім систему можна вирішити за допомогою методу найменших квадратів, щоб отримати оцінку оптичного потоку для кожного пікселя, так само як це описувалося для 8-точкогового алгоритму.

Метод Лукаса-Канаде ефективний для невеликих рухів і добре текстурованих областей, але може зіткнутися з проблемами за наявності великих рухів, шуму та оклюзій. Щоб вирішити ці проблеми, були розроблені більш просунуті методи оптичного потоку, такі як багатомасштабні (пірамідні) підходи, методи регулярізації та моделі на основі глибокого навчання.

Варіаційні методи, такі як метод Горна та Шунка, покладаються на оптимізацію енергетичної функції, яка включає обмеження як на точність даних, так і на гладкість:

$$E = \iint \left[(I_x u + I_y v + I_t)^2 + \alpha^2 (\|\nabla u\|^2 + \|\nabla v\|^2) \right] \mathrm{d}x\mathrm{d}y$$

Ці методи намагаються мінімізувати розбіжності між припущеннями сталості яскравості та просторової гладкості в полі

оптичного потоку, що призводить до більш точної та когерентної оцінки руху.

2.3 Глибоке навчання як метод вирішення неоднозначностей

Під час 3D-реконструкції неоднозначність часто виникає через різні фактори, такі як шум у вхідних даних, оклюзії, самоподібні структури та неповна або відсутня інформація. Класичні методи, які спираються на геометричні та фотометричні обмеження, можуть важко вирішити ці неоднозначності, не роблячи сильних припущень щодо сцени. Ці припущення, однак, можуть не завжди відповідати реальним сценаріям, що призводить до неточних реконструкцій.

Методи глибокого навчання можуть допомогти вирішити ці проблеми, використовуючи попередні знання, отримані з великої кількості даних, які можна використовувати для керування процесом реконструкції та вирішення неоднозначностей. Ці попередні знання можуть мати, наприклад, такі форм:

- Розуміння сцени: моделі глибокого навчання можна навчити на великих наборах даних, щоб неявно фіксувати зв’язки між різними елементами сцени, такими як об’єкти, їх форми та просторове розташування. Використовуючи ці знання в процесі тривимірної реконструкції, система може приймати більш обґрунтовані рішення про те, як інтерпретувати неоднозначні дані, що призводить до більш точних реконструкцій.
- Семантична інформація: моделі глибокого навчання, такі як мережі семантичної сегментації, можна використовувати для зв’язування семантичних міток з різними частинами вхідних даних. Цю семантичну інформацію можна використовувати як форму попередніх знань для коригування процесса реконструкції. Наприклад, знання того, що певна область належить до фасаду

будівлі, може допомогти системі застосувати архітектурні обмеження та краще вирішувати неоднозначності.

- Попередні знання про форму: моделі глибокого навчання також можуть вивчати попередні знання про форми, які є представленнями загальних форм або структур, знайдених у даних. Ці попередні можуть бути використані для регулярізації процесу реконструкції, допомагаючи вирішити неоднозначності, заохочуючи реконструйовану поверхню відповідати відомим формам або структурам.
- Часова інформація: у випадках, коли вхідними даними є послідовність зображень або хмари точок, моделі глибокого навчання можна використовувати для вивчення попередніх даних про час, які фіксують динаміку та послідовність сцени з часом. Використовуючи цю часову інформацію, система може краще вирішувати неоднозначності та генерувати точніші реконструкції.

Глибоке навчання може доповнити класичні методи 3D-реконструкції, надаючи цінний підхід для вилучення попередніх знань і вирішення неоднозначностей. Інтеграція методів глибокого навчання в традиційні конвеєри може бути застосована наступними методами:

- Надійне виділення ознак: класичні конвеєри в 3D-реконструкції часто покладаються на елементи ручної роботи, які можуть бути ненадійними або неінваріантними до різних умов, таких як освітлення, точка зору або текстура. Моделі глибокого навчання, зокрема згорткові нейронні мережі (CNN), продемонстрували здатність вивчати ієрархічні представлення ознак, які є більш надійними та узагальненими. Використовуючи CNN на етапі виділення ознак, конвеєр 3D-реконструкції може отримати переваги від цих надійних функцій, покращуючи загальну продуктивність.

- Вивчення попередньої інформації про об’єкти реального світу на основі даних: класичні методи часто покладаються на сильні припущення щодо даних, які не завжди відповідають реальним сценаріям. Моделі глибокого навчання, з іншого боку, мають здатність вивчати попередні знання з великих обсягів даних, неявно фіксуючи статистику та зв’язки всередині даних. Завдяки інтеграції глибокого навчання в конвеєр 3D-реконструкції система може вивчати та використовувати ці попередні для вирішення неоднозначностей і покращення якості реконструкції.
- Регулярізація на основі даних: класичні методи часто використовують методи регулярізації, щоб забезпечити гладкість або інші бажані властивості реконструйованої поверхні. Однак ці регуляризатори можуть базуватися на простих припущеннях, які не завжди відображають справжню складність сцен реального світу. Моделі глибокого навчання можна використовувати для вивчення регуляризаторів на основі даних, які можуть краще відображати справжню базову структуру сцени та будувати до більш точні і правдоподібні реконструкції.
- Наскрізна оптимізація: моделі глибокого навчання навчаються за допомогою наскрізної оптимізації, що дозволяє одночасно оптимізувати всі компоненти конвеєра. Завдяки інтеграції моделей глибокого навчання в конвеєр 3D-реконструкції стає можливим спільно оптимізувати всю систему, що призводить до кращої загальної продуктивності та ефективніших обчислень.
- Покращена масштабованість: деякі класичні методи 3D-реконструкції можуть бути дорогими з обчислювальної точки зору та їх важко масштабувати до великих наборів даних або програм реального часу. Моделі глибокого навчання можуть

використовувати переваги прискорення GPU та ефективних алгоритмів, що забезпечує швидшу обробку та масштабованість.

Незважаючи на значний прогрес у розробці підходів, заснованих на глибокому навчанні, для оцінки глибини з відеопотоку, все ще залишаються проблеми. Деякі з основних проблем включають точне оцінювання глибини в областях із оклюзіями, обробку різних умов освітлення, роботу з розмиттям руху чи тремтінням камери та обробку сцен із відбиваючими поверхнями чи прозорими об’єктами.

Існуючі методи оцінки глибини з відеопотоку часто покладаються на навчання з учителем за мітками істинної глибини, які не завжди доступні в реальних сценаріях. Крім того, точна оцінка глибини з однієї монокулярної камери може бути за своєю суттю неоднозначною через відсутність інформації про істинну 3D структуру, що ускладнює досягнення високої точності в усіх сценах і сценаріях.

2.4 Обчислення оптичного потоку за допомогою машинного навчання

Протягом багатьох років методи оцінки оптичного потоку еволюціонували від традиційних, створених вручну методів до підходів, заснованих на глибокому навчанні, що призвело до значного покращення точності та продуктивності.

FlowNet, представлений у 2015 році була першою архітектурою глибокого навчання, спеціально розробленою для оцінки оптичного потоку. FlowNet продемонстрував, що наскрізне навчання можливе для оцінки оптичного потоку, і він перевершив класичні методи за кількома тестами. Однак його точність все ще була обмеженою порівняно з останніми підходами глибокого навчання.

Архітектура FlowNet складалася з двох основних компонентів: кодера та декодера. Кодер обробив пари вхідних зображень і витягнув

багатомасштабні представлення ознак. Потім декодер взяв ці характеристики та реконструював поле оптичного потоку від грубого до точного, включаючи шари підвищення дискретизації та уточнення.

SpyNet, запропонований Ранджаном і Блеком у 2016 році, усунув обмеження FlowNet за допомогою мережевої архітектури просторової піраміди. Це дозволило мережі оцінити оптичний потік у кількох масштабах, дозволяючи фіксувати як великий, так і малий рух. SpyNet використовував невеликий CNN на кожному рівні піраміди та викривлення шарів для вирівнювання зображень у різних масштабах. Ця компактна мережа покращила продуктивність FlowNet, значно зменшивши розмір моделі та вимоги до обчислень.

RAFT (Recurrent All-Pairs Field Transforms), запропонований у 2020 році, ознаменував ще один значний прогрес в оцінці оптичного потоку на основі глибокого навчання. RAFT використав новий підхід для побудови щільного 4D об’єму вартості (cost volume), який агрегував інформацію з усіх пар пікселів у вхідних зображеннях. Потім мережа ітеративно оновлювала поле оптичного потоку, використовуючи повторюваний модуль перетворення поля, який можна вивчати. RAFT досягла рекордної продуктивності в багатьох тестах, перевершивши попередні методи, і встановила новий стандарт для оцінки оптичного потоку. Даний алгоритм вбудований в таку бібліотеку, як PyTorch. Можна бачити, що алгоритм був революційним у своїй області, і майже не не поступається останнім розробкам, що намагаються покращити даний результат [10].

2.5 Алгоритми Two-View SfM, що базуються на машинному навчанні

Ідеї, що використовуються для оптичного потоку, ті, що почерпнуті з алгоритмів, зокрема RAFT, можуть буди плідно застосовані до задачі two-view SfM.

RAFT-Stereo — це алгоритм стереозору, який ґрунтується на успішному алгоритмі оптичного потоку RAFT (Recurrent All-Pairs Field Transforms). Алгоритм RAFT набув популярності завдяки своїй здатності моделювати складні довгострокові залежності в задачах щільного співставлення. Він використовує навчальний шар перетворення полів із усіма парами та періодичні оновлення поля відповідності для створення точних полів щільного потоку.

RAFT-Stereo використовує ті самі принципи та методи, що використовуються в алгоритмі RAFT, але адаптує їх до проблеми стереозв'язку. Щільне співставлення передбачає пошук відповідних точок у парі зображень, зроблених з різних точок зору. Ці відповідності використовуються для оцінки інформації про глибину сцени, яка є важливою для 3D-реконструкції, навігації та інших завдань комп'ютерного зору.

Щоб адаптувати RAFT для стереозору, алгоритм модифіковано для обробки невідповідностей замість повних полів оптичного потоку. Архітектура мережі розроблена для ефективної обробки пар стереозображень і вивчення оцінок невідповідності із вхідних зображень. Крім того, ціль навчання змінено, щоб заохотити мережу вивчати точні стереовідповідності.

Спираючись на сильні сторони алгоритму RAFT, RAFT-Stereo може досягти найсучаснішої продуктивності в завданнях просторового співставлення. Це демонструє універсальність архітектури RAFT і її застосовність до різноманітних проблем щільної відповідності в комп'ютерному зорі.

CREStereo - інший підхід, що також базується на попарній кореляції між пікселями двох зображень. Проте, цей підхід не сівсталяє усі пари пікселів, та не використовує ієрархічну кореляцію. На відміну від мережі оцінки оптичного потоку RAFT та її модифікації для стереозору

RAFT-Stereo, де кореляція всіх пар обчислюється за допомогою множення двох мап особоливостей, отриманих за допомогою згорткової нейронної мережі ней C × H × W, що обчислює об’єм вартості 4D H × W × H × W або 3D H × W × W, CRE лише обчислюємо кореляцію у вікні локального пошуку, яке виводить набагато менший обсяг D × H × W для збереження вартість пам'яті та обчислення. H і W позначають висоту та ширину зображень, а D – це число кореляційних пар, набагато менших за W.

Також є методи, що поєднують класичні геометричні методи, в особливості епіполярну геометрію та налаштування пучка зі знаходженням оптичного потоку [11][12] для побудови більш надійних та гнучких методів реконструкції з двох зображень.

РОЗДІЛ 3
ЗАСТОСУВАННЯ СУЧАСНИХ ЗАСОБІВ КОМП’ЮТЕРНОГО ЗОРУ ДЛЯ ВІДНОВЛЕННЯ МАП ГЛИБИНИ З КАДРІВ ВІДЕО

3.1 Комбінування перевірених засобів вирішення задач SfM та обчислення оптичного потоку для побудови надійного алгоритму

Не дивлячись на те, що задача відновлення мап глибини з двох зображень, є складною, та дослідження якої ведеться багатьма науковцями у даний момент, реконструкція мап глибини з відео, як уже зазначалося, має досить задовільні рішення, якщо не брати до уваги швидкість роботи алгоритму.

У даній роботі пропонується алгоритм відновлення мап глибини, що використовує традиційні геометричні методи для відновлення орієнтації та позиції камери у просторі для кожного кадру, та обчислення оптичного потоку за допомогою глибокого навчання.

Як було зазначено, для реалізації класичного Structure-from-Motion, існує достатньо готового програмного забезпечення. Алгоритм, що вирішує задачу Structure-From-Motion, за визначенням також вирішує задачу одометрії - задачі знаходження орієнтації та позиції камери. Вибір засобу для вирішення задачі одометрії не є принциповим, якщо швидкість виконання не є важливою характеристикою.

Основна проблема, що є у існуючих класичних засобів вирішення задачі Structure-from-Motion - знаходження щільних мап глибини. Мапи глибини, хоча і можуть бути отримані за допомогою таких програмних засобів як COLMAP, мають значні області невизначеності, оскільки реконструкція базуються на знаходженні деякої кількості обраних пікселів на зображеннях та їх зіставленні як проекцій на площину зображення для знаходження відповідностей між ними. В результаті отримується хмара точок у просторі, кожна точка якої проектується на деяку кількість зображень. Таким чином, мапи глибини фактично будуються на базі проекцій хмари відтворених точок на різні кадри відео, що мають різні ракурси.

Рис. 3.1 Мапа глибини за допомогою COLMAP

При застосуванні вищеописаного підхода, попіксельна мапа глибини не отримується. Більш щільні мапи глибини будуються вже на базі первинної реконструкції, проте, і ці мапи глибини на практиці мають області невизначеності.

Для того, щоб відновити щільні мапи глибини, пропонується спочатку отримати позиції камер. Для того, щоб відновити мапу глибини для деякого кадру, пропонується використовувати інформацію про позиції камери інших кадрів, та поле оптичного потоку між кадром, для якого будується мапа глибини, та відповідними кадрами відео.

Для відновлення положення камер, обрано COLMAP, так як це добре перевірене ПЗ, що дуже тонко конфігурується та має зручний консольний інтерфейс та дозволяє зчитувати результати реконструкції, зокрема позиції камер для кожного кадру відео, та параметри калібровки камери.

Після цього отримані позиції камер треба перевести у спільну систему координат. Дані про положення камери, згідно з традиційною схемою, отримуються у вигляді матриць повороту R та векторів переміщення t. Для того щоб перетворити вектор $(X,\ Y,\ Z)^T$ зі світової системи координат до локальної системи координат камери, треба застосувати вектор переміщення та матрицю повороту камери:

$$\begin{pmatrix} X' \\ Y' \\ Z' \end{pmatrix} = R \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} + t$$

Відповідно, для того, щоб знайти глобальні координати вектора, треба виразити його з рівняння:

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = R^{-1} \left(\begin{pmatrix} X' \\ Y' \\ Z' \end{pmatrix} - t \right)$$

Враховуючи, що $R^{-1} = R^T$, бо матриця повороту завжди ортогональна, отримуємо зручний для обчислення перехід до єдиної системи координат.

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = R^T \left(\begin{pmatrix} X' \\ Y' \\ Z' \end{pmatrix} - t \right)$$

Тепер, розглянемо вектор між фокусним центром камери та пікселем *(u, v)*, що є точкою, що лежить на площині зображення. Z-координата будь якої точки площини зображення дорівнює фокальній відстані f, а центровані координати пікселя - $(u - c_x, v - c_y)$. Таким чином, координати точки на площині зображення у системі координат камери дорівнюють $(f, u - c_x, v - c_y)$. Оскільки оптичний центр камери знаходиться у центрі системи координат камери, то і шуканий вектор матиме вигляд

$$a = (f, u - c_x, v - c_y) \qquad (8)$$

Тепер, якщо позиція камери відома, переворенням вектору *a* до світової системи координат безпосередньо отримується напрям променя, на якому лежить точка, що відображається у піксель *(u, v)*. Сам промінь отримується з обмеження, що у рівнянні

$$l = as + b$$

$b = t$ при $s = 0$.

За допомогою оптичного потоку, для кожної точки першого зображення, можна знайти відповідну точку на іншому. Оптичний потік відображає відповідності краще всього, коли зображення, що порівнюються, відрізняються не сильно. Тому, для кожного кадру, є сенс підбирати деяку кількість найближчих кадрів. Для кожного кадру, що співставляється обчислимо оптичний потік, для кожного пікселя вихідного

кадру *(u, v),* підберемо відповідний піксель *(u', v')* за допомогою обчислених полів оптичного потоку.

Рис. 3.2 Відповідності між точками зображень

Для того, щоб отримати відповідну точку на іншому зображенні за допомогою оптичного потоку, треба просто змістити її на значення оптичного потоку між першим та другим зображенням:

$$(u', v') = (u, v) + flow_{u,v} \qquad (9)$$

Тоді вектор між центром координат камери іншого кадру та відповідною точкою на іншому кадрі дорівнює

$$a' = (f, u' - c_x, v' - c_y)$$

В такий спосіб, з наявної інформації про відповідності між точкою на кадрі, для якого будується мапа глибини, та точками на кожному іншому кадрі, отримуються промені у просторі, на перетині яких має безпосередньо лежати точка, що проектується у відповідні пікселі на двох кадрах.

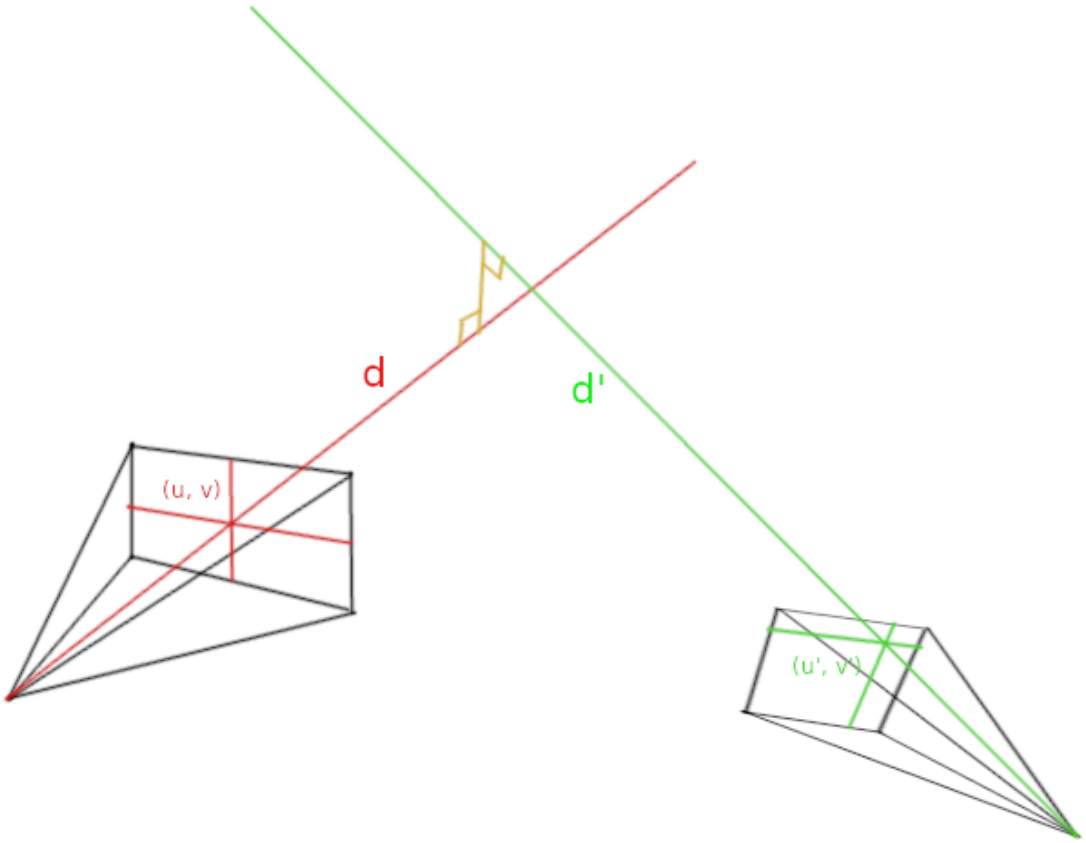

Рис. 3.3 Відстань до точки, що проектується на два кадри

На практиці, промені, що розглядаються, не будуть точно перетинатися, бо для знаходження відповідностей для одометрії використовується розріджене співставлення точок, за допомогою алгоритмів на кшталт SIFT, а для знаходження щільних відповідностей використовується оптичний потік, отриманий за допомогою алгоритмів глибокого навчання. Тому, за допомогою такого методу точку перетину можна знайти лише з деякою точністю.

Треба зауважити, що точку перетину при побудові мапи глибини знаходити не обов’язково, а лише треба знайти відстань d від початку променя у центрі координат камери до найближчої точки до прямої l' на прямій l. Позначимо відповідний промінь як функцію від s:

$$l(s) : l = as + b,\ s \geq 0$$

Якщо вектор a нормалізований, то s дорівнює відстані від початку променя до $l(s)$. Таким чином, $d = s$.

Розглянемо прямі, що відповідають двом відповідним точках на двох кадрах:

$$l(d) = ad + b$$
$$l'(d') = a'd' + b'$$

Тепер, задачу можна переформулювати як знаходження таких $s,\ s'$, за яких відстань від $l(s)$ до $l'(s')$ мінімальна (рис 3.3). Щоб вирішити цю задачу, треба прийняти до уваги той факт, що вектор між точками двох прямих у просторі, що має найменшу можливу довжину, перпендикулярний до обох прямих:

$$< l(d) - l'(d'), a > = 0$$
$$< l(d) - l'(d'), a' > = 0$$

Дана система лінійна відносно *d* та *d'*. Безпосередньо після вирішення системи з двома рівняннями та невідомими, отримуємо значення *d*, що дорівнює відстані від фокусного центру камери до об'єкта у просторі, що відображається на обраний піксель.

Проводячи дані обчислення для кожного пікселя кадру, для якого будується мапа глибини стільки разів, скільки є кадрів, на яких було знайдено відповідний піксель до даного, отримуємо "гіпотези", які можна певним чином фільтрувати та усереднювати для отримання остаточного передбачення відстані до об'єкта. В імплементації цього алгоритма було вирішено брати до уваги тільки ті відповідності для яких

$$\frac{||l(d) - l'(d')||}{d} < 0.01$$

Також, для базової фільтрації хибних відповідностей, що генерує оптичний потік, алгоритм обчислює також оптичний потік між кадром, що

є середнім за номером між вихідним та одним із кадрів на якому шукаються відповідності. Це означає, що крім оптичного потоку між вихідним кадром I_j та кадром I_{j+k}, також обчислюється оптичний потік між $I_{j+\frac{k}{2}}$ та I_{j+k}.

Позначимо положення пікселів з кадру I_j на кадрі I_{j+k} як $I_j \to I_{j+k}$. Так, як оптичний потік між I_j та $I_{j+\frac{k}{2}}$ вже обчислений за побудовою алгоритму, то для кожного пікселя також можна знайти відповідність $(I_j \to I_{j+\frac{k}{2}}) \to I_{j+k}$. Якщо оптичний потік знаходиться алгоритмом ідеально, то $I_j \to I_{j+k}$ повинне представляти ту ж саму відповідність що і $(I_j \to I_{j+\frac{k}{2}}) \to I_{j+k}$. Але на практиці, навіть такий точний алгоритм як RAFT, може давати хибні результати у ряді випадків. Для фільтрації хибних відповідностей у даній роботі пропонується перевіряти вищезазначену умову. Проте, як і у випадку знаходження “приблизного” перетину між променями, треба задати деякий поріг допустимого відхилення. Загалом, це дає просте правило, що відсіює значну кількість хибних відповідностей.

Для усереднення результатів використовувалися емпіричні ваги, які виникли з ідеї про те, що при малих кутах між прямими, важливу роль мають помилки обчислення та неточності роботи алгоритмів одометрії та знаходження оптичного потоку, тому достовірність таких результатів знижується, і при нульовому куті - це взагалі вироджений випадок. Простими для обчислення ваговими коефіцієнтами, що дозволяють “заглушити” шум від недостовірних результатів, є $sin^2(\alpha)$, де α - кут між a та a'. Обчислити, як відомо, ці коефіцієнти можна як

$$\sin^2(\alpha) = 1 - \cos^2(\alpha) = 1 - < a, a' >^2$$

ВИСНОВКИ

У роботі були проаналізовані можливості застосування сучасних алгоритмів відновлення відновлення тривимірних сцен з зображень для відновлення просторової інформації із відеопотоку.

Було показано розмаїття методів, підходів та алгоритмів, і сучасні тренди в області реконструкції тривимірної інформації з відео. Приділено увагу послідовності розвитку підходів до вирішення задачі - можна підсумувати, що алгоритми відновлення тривимірної інформації починали розвиватися з суто геометричних підходів і в останні роки все більше використовують глибоке навчання.

У процесі дослідження області та результатів, пов’язаних з тривимірною реконструкцією на основі зображень та відеопотоків, був винайдений алгоритм, що дозволяє будувати щільні мапи глибини, використовуючи інформацію з усіх кадрів відео.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. https://arxiv.org/pdf/2004.15021.pdf
2. https://openaccess.thecvf.com/content/CVPR2021/papers/Yang_LASR_Learning_Articulated_Shape_Reconstruction_From_a_Monocular_Video_CVPR_2021_paper.pdf
3. http://www.robustvision.net/leaderboard.php
4. https://openaccess.thecvf.com/content/CVPR2022/papers/Li_Practical_Stereo_Matching_via_Cascaded_Recurrent_Network_With_Adaptive_Correlation_CVPR_2022_paper.pdf
5. https://arxiv.org/pdf/2208.04726.pdf
6. http://webdiis.unizar.es/~raulmur/MurMontielTardosTRO15.pdf
7. https://arxiv.org/pdf/2007.11898.pdf

8. https://www.academia.edu/65495130/Computer_Vision_A_modern_approach
9. Richard Hartley, Andrew Zisserman, “Multiple View Geometry in Computer Vision, 2nd Edition”.
10. https://paperswithcode.com/sota/optical-flow-estimation-on-sintel-clean?p=flownet-20-evolution-of-optical-flow
11. https://openaccess.thecvf.com/content/CVPR2021/papers/Wang_Deep_Two-View_Structure-From-Motion_Revisited_CVPR_2021_paper.pdf
12. https://arxiv.org/pdf/2302.00523.pdf

АНОТАЦІЯ

Дана робота присвячена застосуванню сучасних алгоритмів відновлення відновлення тривимірних сцен з зображень для відновлення просторової інформації із відео.

У роботі розглядається розмаїття сучасних методів, підходів та алгоритмів та трендів в області. Приділено увагу послідовності розвитку підходів до вирішення задачі. Розглянуто, як алгоритми відновлення тривимірної інформації починали розвиватися з суто геометричних підходів і в останні роки все більше використовують глибоке навчання.

У процесі дослідження області та результатів, пов’язаних з тривимірною реконструкцією на основі зображень та відеопотоків, був винайдений алгоритм, що дозволяє будувати щільні мапи глибини, використовуючи інформацію з усіх кадрів відео. Ідея полягає у тому, щоб використовувати готові, загальноприйняті та перевірені рішення для вирішення двох задач: COLMAP - для візуальної одометрії, та RAFT - для обчислення оптичного потоку. Алгоритм показує досить точні результати, та відновлює мапу глибини в деталях на довільних статичних сценах.

ANNOTATION

This work is dedicated to the application of modern algorithms for reconstructing spatial scenes from images to restore spatial information from video.

The work is looking at a variety of modern methods, approaches, algorithms and trends in the field. The attention was paid to the sequence of development of approaches to the completion of the task. The work shows how the algorithms for restoring three-dimensional information began to develop from purely geometric approaches and in recent years are increasingly using deep learning.

While researching the field and results related to three-dimensional reconstruction based on images and video streams, an algorithm was invented that allows constructing dense depth maps using information from all video frames. The idea is to use ready-made, commonly accepted, and tested solutions to solve two problems: COLMAP for visual odometry, and RAFT for computing optical flow. The algorithm shows quite accurate results and reconstructs the depth map in detail on arbitrary static scenes.