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 візуального виявлення місць»

«Algorithms of visual place recognition»

Виконав: здобувач денної форми навчання
спеціальності 113 Прикладна математика
Освітня програма «Прикладна математика»
Данило Сергійович ЖИКУЛ

Керівник: _____ канд. тех. наук, доц. Володимир МОРОЗ
Рецензент: канд. фіз.-мат. наук, доц. Віктор ВЕРБІЦЬКИЙ

Рекомендовано до захисту:

Протокол засідання кафедри

№ ___ від _____ 2025 р.

Завідувач кафедри
_____ Ольга КІЧМАРЕНКО
_____ _____

Захищено на засіданні ЕК № _____

Протокол № ___ від _____ 2025 р.

Оцінка _____ / _____ / _____

Голова ЕК

_____ _____

Одеса — 2025 р.

ЗМІСТ

УМОВНІ ПОЗНАЧЕННЯ

1) VPR – візуальне виявлення місць
2) loop closure – замикання петель
3) perceptual aliasing – перцептивне дублювання
4) occlusion – оклюзія
5) single-session VPR – односеансова задача візуального виявлення місць
6) multi-session VPR – багатосеансова задача візуального виявлення місць
7) DoG – різниця гаусіанів
8) LoG – лапласіан гаусіана
9) BoVW – мішок візуальних слів
10) VLAD – вектор локально агрегованих дескрипторів
11) NN – найближчі сусіди
12) MNN – взаємні найближчі сусіди
13) ANN – наближений пошук найближчого сусіда
14) Lowe ratio – тест Лоу
15) SMNN – взаємні найближчі сусіди з тестом Лоу
16) SNN – найближчі сусіди з тестом Лоу
17) inlier – спостереження, яке узгоджується з підлаштованою моделлю в межах заданого порогу похибки.
18) dense matcher – щільний зіставник
19) ground truth – істинні значення або пари, істинна траєкторія

ВСТУП

За останні роки задача візуального виявлення місць (Visual Place Recognition або VPR) привернула до себе увагу великої кількості дослідників, що призвело до значного зростання кількості досліджень у цьому напрямку. Станом на 20.05.2025 в Google Scholar налічується понад 5060 статей, що містять "visual place recognition"у своїй назві, з яких 4520 було опубліковано після оглядової роботи Lowry та ін. [1] (2015). За період з початку 2024 року до 20.05.2025 налічується 1120 робіт. Це говорить про **актуальність** проблеми в задачах комп'ютерного зору та робототехніки.

Задача VPR ускладнюється через низку факторів: наявність візуально схожих місць, зміну точки спостереження, варіації освітлення, сезонні й структурні зміни середовища, наявність оклюзій. Сучасні методи, які зазвичай класифікують за типом дескрипторів, пропонують різні шляхи подолання цих складнощів. Більшість алгоритмів і відповідний математичний апарат походять із суміжних завдань або були спочатку розроблені для інших напрямків (SLAM, 3D-реконструкції, обробки сигналів).

У цій роботі розглянуто головні алгоритми для розв'язку задачі візуального виявлення місць.

Об'єктом дослідження є задача візуального виявлення місць.

Предметом дослідження є аналіз ефективності сучасних підходів до задачі VPR.

Метою роботи є пошук ефективного алгоритму VPR шляхом аналізу та порівняння існуючих методів на основі локальних, глобальних дескрипторів та їх комбінацій. Порівняльний аналіз проводиться за результатами тестування на кількох наборах даних, з огляду на якість і швидкість алгоритмів.

За результатами експериментів оцінено вплив комбінації локальних та глобальних дескрипторів на якість роботи алгоритмів порівняно з їх окремим використанням. Практичне значення цих результатів полягає в тому, що вони дозволяють чітко визначити компроміси між швидкістю та якістю роботи алгоритмів VPR, залежно від конфігурації дескрипторів.

В першій главі детально розглянуто історію розвитку алгоритмів VPR, постановку задачі та поділ її на категорії, вплив навколишнього середовища на роботу підходів та види змін середовища, які ускладнюють роботу алгоритмів.

У другій главі розглянуто математичну постановку задачі, описано ключові етапи та метрики оцінювання. З-поміж основних етапів виділено:

- детектування та опис зображень за допомогою локальних і глобальних дескрипторів;
- методи агрегації локальних дескрипторів;
- алгоритми співставлення локальних і глобальних дескрипторів;
- геометричну верифікацію для локальних дескрипторів;
- правила відбору кандидатів за матрицею схожості;
- гібридний підхід, що поєднує глобальні та локальні дескриптори.

У третій главі подано опис основних підходів, що аналізуються в цій роботі.

У четвертій главі здійснено експериментальну перевірку методів, розглянутих у третій главі. Тестування проводиться на трьох наборах даних, які відрізняються рівнем складності та характером візуальних змін.

Результати роботи доповідались на II (VIII) Міжнародній науково-практичній конференції здобувачів вищої освіти і молодих учених «Інформаційні технології: теорія і практика» [2], яка проходила 2-4 квітня 2025 року в Національному університеті «Запорізька політехніка», видані тези. Сертифікат розміщений у додатку А.

Код доступний за посиланням github.com/DanielZhicool/VPR_diploma.

РОЗДІЛ 1

ОГЛЯД ЗАДАЧІ

Візуальне виявлення місць представляє собою задачу з розпізнавання попередньо відвіданих місць на основі візуальної інформації. Ця задача зустрічається в багатьох галузях, в тому числі в робототехніці, БПЛА, БППА, автономних транспортних засобах, AR технологіях та ін. Алгоритми VPR використовуються для виявлення замикання петель (Loop closure) в контексті методів навігації SLAM та для підбору кандидатів для візуальної 6DoF локалізації. Візуальне виявлення місць також використовується, коли є необхідність у навігації без доступу до супутникових систем навігації (GPS, Galileo, BeiDou та ін.) або у якості резервної системи.

Зазвичай проблему візуального виявлення місць розглядають як задачу пошуку екземпляра (instance retrieval task), де локація зображення із запиту оцінюється найбільш візуально схожим зображенням з бази даних із геоприв'язкою [3]. Саме таким чином ми й будемо її розглядати.

З початку 20 століття і донині майже усі підходи були засновані на міркуваннях, в основі яких полягає репрезентація зображення через розрахунок дескрипторів, наприклад, Mobile Robot Localization and Mapping with Uncertainty using Scale-Invariant Visual Landmarks [4]. Тобто, розв'язок задачі репрезентації зображень є важливою частиною візуального виявлення місць. Для розв'язку цієї підзадачі спочатку активно використовувались глобальні (холістичні) дескриптори по типу GIST (2006) [5], але пізніше увага перейшла до локальних дескрипторів, які використовувались з моделлю Bag of Visual Words (BoVW) [6], словник якої формувався з центрів кластерів локальних дескрипторів, отриманих методом к-середніх. Прикладом використання локальних дескрипторів є система навігації appearance-only SLAM під назвою FAB-MAP 2.0 (2011) [7]. Класичними представниками локальних дескрипторів є SIFT (2004) [8] та SURF (2006) [9]. До появи дескрипторів, які базуються на глибинному навчанні, дескриптори по типу SIFT та SURF лежали в основі більшості систем VPR.

Слідом за збільшенням кількості та якості результатів, що пов'язані з

використанням глибокого навчання в галузі комп'ютерного зору, почався новий етап дослідження задачі VPR. Фокус дослідників змістився у напрямку глибинного навчання та почали публікуватись перші дослідження в цьому напрямку. Ключовим моментом у розповсюдженні глибинного навчання можна відмітити публікацію наукової роботи ImageNet classification with deep convolutional neural networks [10] у 2012 році, у цій роботі було продемонстровано потенціал глибинних нейронних мереж для розв'язання задач класифікації і відповідно їх потенціал у роботі з візуальною інформацією на прикладі запропонованої глибинної мережі AlexNet. У наступні роки почали з'являтись перші підходи до задачі VPR, де замість раніше згаданих класичних дескрипторів запропонували використовувати глибинні згорткові нейронні мережі. У 2014 році у науковій роботі Chen та ін. [11] було продемонстровано першу спробу використання глибинної нейронної мережі, а саме OverFeat [12] в якості дескриптора для задачі VPR. Запропонований підхід продемонстрував гарний результат у порівнянні з сучасними на той час підходами [11]. Далі, у 2019 році, було запропоновано NetVLAD, який дозволяє навчати дескриптор одразу для задачі VPR, замість використання мереж, навчених для інших задач комп'ютерного зору [3]. Хоч NetVLAD і є дескриптором, але він поєднує у собі кроки репрезентації та агрегації, де репрезентацією займається деяка нейронна мережа, наприклад AlexNet [10], ResNet [13] або VGG-16 [14], а агрегацією вихідної feature map у компактний дескриптор займається додатковий шар, на базі Vector of Locally Aggregated Descriptors (VLAD) [15]. На досить складному наборі даних 24/7 Tokyo [16] NetVLAD показав набагато кращі результати у порівнянні з сучасними на той час дескрипторами, які використовували попередньо натреновані згорткові мережі чи класичні алгоритми [3]. Наразі більшість сучасних підходів використовують неглибокі шари агрегації, які підключаються до попередньо навчених згорткових нейронних мереж, обрізаних на останньому feature-rich шарі [17], також нещодавно, почали з'являтись дескриптори, в основі яких лежать трансформери, наприклад LoFTR (2021) [18]. LoFTR має змогу одразу видавати набори співпадаючих ключових точок без явного розрахунку дескрипторів. [18].

У 2023 році було продемонстровано [19], що гібридні методи, які поєднують глобальні та локальні дескриптори, помітно покращують якість

у складних середовищах.

Усі провідні дескриптори для VPR сьогодні є навченими (глибокі нейронні мережі, трансформери) й поділяються на дві групи:

- глобальні: NetVLAD, CosPlace [20] та інші;
- локальні: зокрема SuperPoint [21], а також нові DISK [22] і DeDoDe [23], які базуються на CNN архітектурах, та LoFTR, RoMA [24] – трансформерні методи прямої відповідності (dense matchers) без окремого детектора ключових точок.

Можна побачити, що лише в плані репрезентації зображень підходи до задачі візуального виявлення місць дуже сильно еволюціонували за останні 20 років. Починаючи з підходів, заснованих на класичних алгоритмах, і до сучасних, що базуються на глибинних нейронних мережах, трансформерах та з року в рік активно покращують результати попередників.

1.1 Постановка задачі

Візуальне виявлення місць використовується у різних контекстах, в результаті чого задача має трохи різний вигляд залежно від деяких параметрів. Ми будемо використовувати постановку, наведену в роботі Schubert та ін. [25]. Спочатку задамо формальне визначення задачі: нехай задана база даних відвіданих місць представлена множиною зображень $DB = \{I_1, I_2, \ldots, In\}$, де кожне зображення відповідає деякому місцю та йде в парі з інформацією про його положення на мапі (IMU, GPS та ін.), а множина запитів задана множиною $Q = \{I_1, I_2, \ldots, Im\}$, тоді задача полягає в тому, щоб для кожного зображення $I_j \in Q$ знайти відповідне зображення $I_i \in DB$, яке відповідає тому ж самому місцю. Тепер, коли ми маємо формальне визначення задачі, можемо розглянути поділ задач на категорії залежно від вхідних даних, способу обробки та бажаного результату [25] (Табл. 1.1).

	Обробка	
Вхідні дані	Online VPR	Batch VPR
Одно-сеансовий VPR	Online SLAM	Mapping
Багатосеансовий VPR	DB зростає: Multirobot mapping; DB стала: Visual (re-)localization	Multisession Mapping

Табл. 1.1. Типи задач на основі вхідних даних та режимів обробки у VPR

1) Вхідні дані: якщо Q та DB – різні множини: $Q \cap DB = \emptyset$, то задачу називають багатосеансовою (multi-session VPR), а коли йде порівняння однієї множини самої до себе, тобто $Q = DB$, задачу називають односеансовою (single-session VPR). У випадку, коли задача односеансова, важливо відкидати нещодавно отримані зображення під час пошуку відповідних зображень задля уникнення знаходження відповідностей між зображеннями з одного і того самого візиту.
2) Обробка: коли зображення I_j поступають та проходять обробку по одному, тобто множина Q постійно зростає, а DB може як і зростати, так і не змінюватися (online VPR). Або, коли усі I_j доступні одразу (batch VPR). Зростання множин обмежує кількість підходів, що ми можемо застосувати, а тому ускладнює задачу.
3) Результат: якщо завданням є знаходження одного зображення, що відповідає шуканому місцю (single-best-match VPR), тобто задача полягає в знаходженні зображення, яке підходить найбільше. Іншим завданням є знаходження усіх або декількох зображень, що відповідають шуканому місцю (multi-match VPR).

У підсумку можна сказати, що DB представляє множину зображень вже відвіданих місць (інформація щодо положення яких на мапі вже відома), в той час як Q – це зображення про з поточного місцезнаходження (зображення, яке ми хочемо локалізувати). Також варто відзначити, що коли $DB \cap Q = \emptyset$, то елементи DB та Q можуть належати до різних генеральних сукупностей, це може бути викликано різницею в часі, зовнішніх умовах та способу збору інформації для бази даних. Наприклад, інформація бази даних збирається автомобілем з направленою вперед камерою, а запити (поточна інформація з Q) поступають з камери смартфона.

1.2 Вплив навколишнього середовища

Середовища, в яких оперують системи візуального виявлення місць, включають до себе різні типи місцевості. Це різноманіття середовищ є одним із факторів, що стимулює дослідження цієї проблеми, оскільки підхід, який добре працює в одному середовищі, може показувати набагато гірший результат на інших типах місцевості. В різних сферах розглядають різні типи середовищ, такі як: на відкритому повітрі/у приміщенні, рукотворне/природне, структуроване/відкрите та інші [27]. Наприклад, у сфері автономних транспортних засобів в якості середовища найчастіше зустрічаються міста (рукотворні та на відкритому повітрі). Варто зазначити, що алгоритми візуального виявлення місць, окрім наземних, також застосовуються на літальних та підводних апаратах. Вони мають свою специфіку та існують окремі дослідження, що їх розглядають: [26, 28].

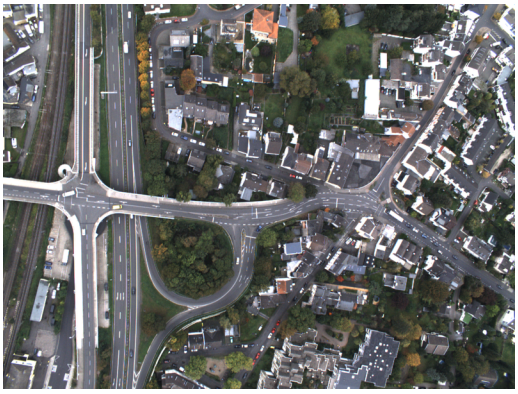


Рис. 1.1. Фото місцевості зроблене з крила літака [26]

Інформація, яку використовують алгоритми, зазвичай є зображеннями середовища з встановленої на апараті камери. Кількість, тип та орієнтація камер вже залежать від самого апарату (призначення, середовище експлуатації та ін.). Наприклад, для літального апарату візуальна інформація може представляти собою зображення місцевості, що були зняті на спрямовану донизу камеру, яка закріплена на крилі чи іншій частині літака (Рис. 1.1). Здебільшого VPR розглядають саме для наземних середовищ.

Однією з головних проблем візуального виявлення місць є те, що в реальному світі є велика кількість факторів, які впливають на якість роботи алгоритмів:

1) Зміна точки спостереження (Рис. 1.4). Цей фактор має місце майже незалежно від середовища, але у випадках коли місцевість досить структурована, наприклад дорожня інфраструктура у місті, ступінь варіації точки зору значно обмежений [29]. Схожий ефект також присутній при досягненні достатньої висоти літальним апаратом [27].

2) Зміна зовнішнього вигляду місць. Основними та найбільш розповсюдженими чинниками є зміна освітлення, погодних умов та сезонів (Рис. 1.4, 1.5). Більш складними є випадки, коли зміни не мають періодичного характеру, як зміна дня і ночі та зміна погоди. Такими є випадки, коли місце зазнає значних структурних змін з плином часу (зміна вигляду/старіння/руйнування будівлі у місті) (Рис. 1.3, 1.7).
3) Наявність візуально схожих місць (perceptual aliasing) (Рис. 1.8).
4) Наявність оклюзій (occlusion) (Рис. 1.2, 1.3, 1.5). Тобто наявність об'єктів, які перекривають ключові елементи середовища. Наприклад у міському середовищі такими є пішоходи та автомобілі.

В останні роки багато уваги було приділено саме візуальному виявленню місць у випадках, коли зовнішній вигляд середовища зазнає значних змін [16, 30, 31].

Рис. 1.2. Зміна зовнішнього вигляду місця з плином часу, зміни точки спостереження та наявності оклюзій : 2009 → 2024, зображення з Google Street View [32]

Рис. 1.3. Зміна зовнішнього вигляду місця з плином часу: 2015 → 2024, зображення з Google Street View

Рис. 1.4. Зміна зовнішнього вигляду місця, залежно від часу доби, сезону та точки зору. Зображення з Freiburg across seasons [31]

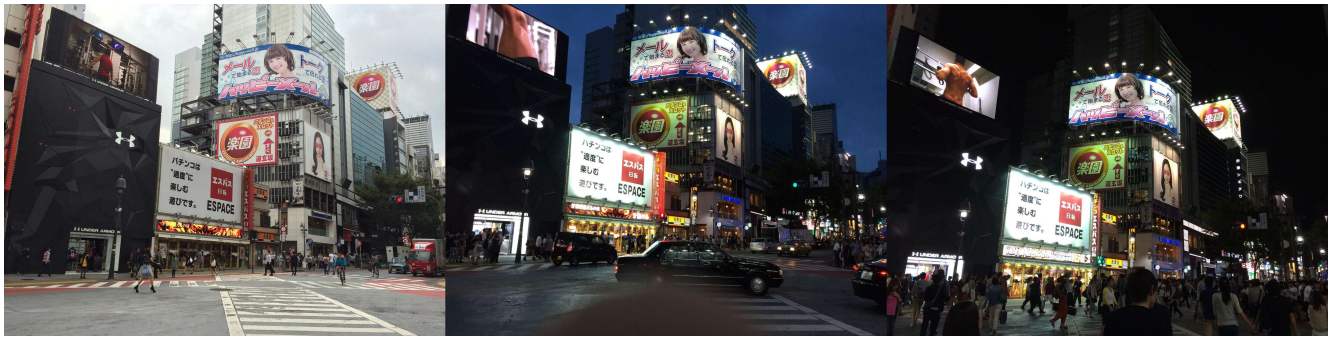


Рис. 1.5. Зміна зовнішнього вигляду місця, залежно від часу доби. Зображення з 24/7 Tokyo

Рис. 1.6. Зміна зовнішнього вигляду місця, через зміну часу доби та появи оклюзій. Зображення з 24/7 Tokyo

Рис. 1.7. Зміна зовнішнього вигляду місця з плином часу: 2013 → 2014 → 2022, Зображення з 24/7 Tokyo та Google Street View

Рис. 1.8. Візуально схожі, але різні місця. Зображення з 24/7 Tokyo

РОЗДІЛ 2

МАТЕМАТИЧНА МОДЕЛЬ ЗАДАЧІ VPR

Перед тим як почати огляд різних підходів, розглянемо загальну ідею. Будемо розглядати багатосеансову задачу batch VPR, тобто $Q \cap DB = \emptyset$ та всі запити поступають одним пакетом. В секції з постановкою задачі вже було відмічено, що ми представляємо навколишній світ як множину зображень. Для того, щоб отримувати локацію поточних зображень $I_j \in Q$, з кожним $I_i \in DB$ іде деяка інформація про локацію відповідного місця, наприклад, GPS, LiDAR, IMU.

Першим кроком є відображення $I_i \in DB$ з простору зображень у простір репрезентацій за допомогою деякого екстрактора:

$$F : \mathbb{I}^{c \times h \times w} \longrightarrow \mathbb{D}$$

Тут $\mathbb{D}$ – це простір репрезентацій, $\mathbb{I}^{c \times h \times w}$ – простір зображень розміру (h, w) з кількістю каналів c.

Задля уникнення повторного розрахунку репрезентацій, вони зберігаються в пам'яті для подальшого використання (ми вважаємо, що вміст DB не змінюється під час роботи системи).

Під час роботи системи для кожного запиту $I_j \in Q$ аналогічно обчислюють $F(I_j)$. Далі обчислюють міру схожості

$$S\big(F(I_j), F(I_i)\big)$$

(наприклад, Евклідова відстань, косинус подібності або кількість збігів локальних дескрипторів), і на її основі знаходять найбільш подібне зображення з DB:

$$i^* = \arg\max_{i \in DB} S\big(F(I_j), F(I_i)\big).$$

Локація запиту I_j приймається рівною локації її зображення $I_{i^*} \in DB$.

В залежності від контексту відображення F та репрезентацію $F(I)$ будемо називати дескриптором.

Загалом можемо поділити загальний підхід на декілька основних кроків (Рис. 2.1):

1) Репрезентація зображень на основі дескрипторів (feature extraction).
2) Порівняння цих дескрипторів та відповідно оцінка схожості кожної пари зображень $(I_j, I_i) \in Q \times DB$ (descriptor similarity).
3) На основі отриманих оцінок подібності проводимо відбір необхідної кількості кращих кандидатів з DB для кожного $I_j \in Q$ (matching).

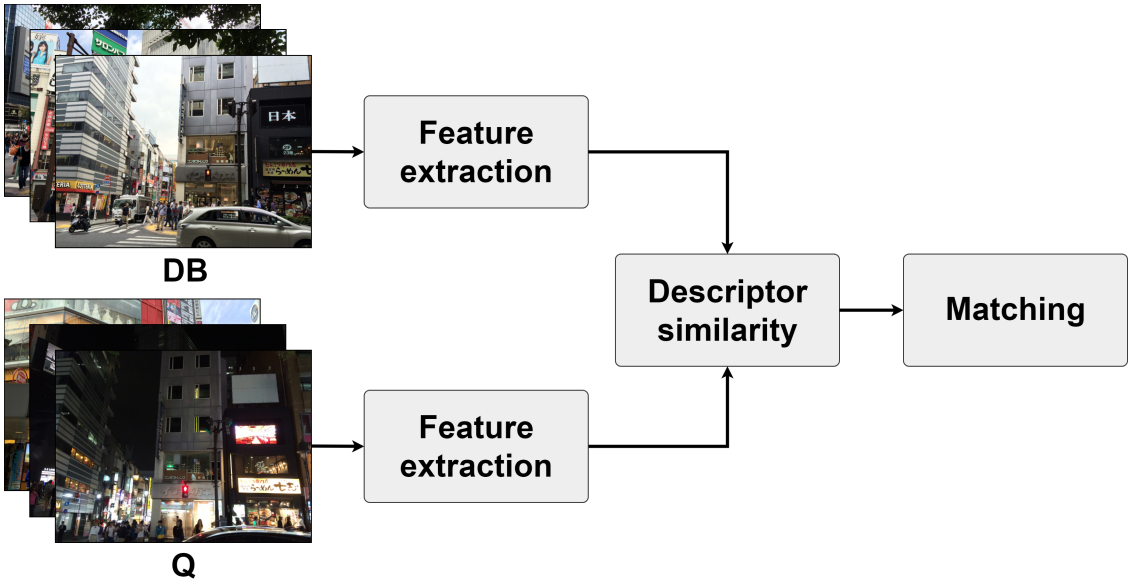


Рис. 2.1. Загальний підхід до візуального виявлення міць

2.1 Feature extraction

На цьому кроці ми розраховуємо репрезентації за допомогою дескрипторів, які ставлять у відповідність зображенню I один або більше векторів $d_i \in \mathbb{R}^p$. Основна задача цих дескрипторів – якомога краще передати характер місця на зображенні. Це значно спрощує задачу порівняння зображень, бо в результаті проблема зводиться до порівняння векторів. Дескриптори за характером опису зображення поділяють на локальні та глобальні(холістичні), а за методом побудови самого дескриптора на рукотворні та навчені.

2.1.1 Локальні дескриптори

Ідея локальних дескрипторів полягає в тому, що вони репрезентують зображення набором з n векторів $d_i \in \mathbb{R}^p$, $i = \overline{1,n}$, розрахованих в n ключових точках: $k_i \in \mathbb{R}^2$, $i = \overline{1,n}$ (Рис. 2.2). Класичний підхід до розрахунку локальних дескрипторів проходить у два етапи:

1) Пошук ключових точок зображення (keypoint detection).
2) Розрахунок локальних дескрипторів для кожної ключової точки.

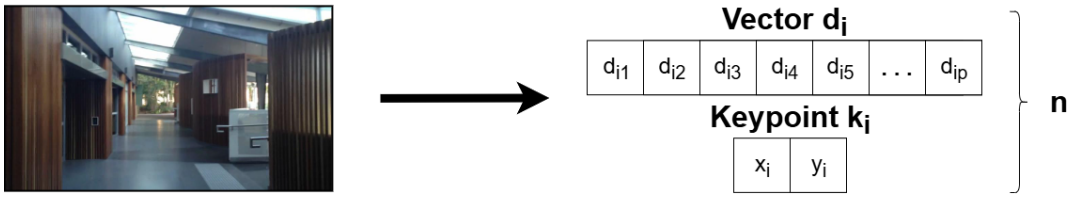


Рис. 2.2. Локальний дескриптор

Для пошуку ключових точок традиційно використовували спеціальні детектори. Задача детекторів ключових точок полягає у знаходженні точок певного характеру. Характер ключових точок буває різним та залежить безпосередньо від детектора. Класичним прикладом є детектори кутів (Harris corner detector [33], FAST [34]), вони добре знаходять кути та границі між об'єктами та є досить швидкими [34], але в сучасних підходах майже не використовуються. Іншими відомими детекторами є детектори плям: лапласіан гаусіана (LoG), різниця гаусіанів (DoG) (Рис. 2.3а) та інші, DoG є стандартним детектором для дескриптора SIFT [8]. Пошук ключових точок за допомоги додаткового алгоритму є здебільшого характерним для класичних дескрипторів (SIFT, SURF, ORB [35], AKAZE [36], BRISK [37] та інші), хоча й деякі сучасні дескриптори використовують таку схему, наприклад DeDoDe [23] (Рис. 2.3б), SuperPoint [21]. Велика кількість сучасних локальних дескрипторів, які ще називають неявними локальними дескрипторами, бо вони не розраховують ключові точки явно, а одразу повертають пари зіставлених точок (які вказують на одні й ті ж об'єкти чи місця) для 2 вхідних зображень. Усі такі дескриптори є навченими, як приклади: LoFTR [18], RoMa [24].

(а) DoG (SIFT, 2004) (б) DeDoDe (2023)

Рис. 2.3. Візуалізація 2000 найкращих ключових точок, що були знайдені за допомоги DoG та DeDoDe. Зображення з 24/7 Tokyo [16]

Етап розрахунку локальних дескрипторів полягає у розрахунку дескриптора для кожної знайденої ключової точки. В результаті отримуємо набір з n дескрипторів: $d_i \in \mathbb{R}^p$, $i = \overline{1, n}$, які описують зображення. Розмір дескриптора залежить насамперед від самого методу, наприклад, дескриптор SIFT – це 128-вимірний вектор, а DeDoDe – 256.

Історично локальні дескриптори показували гарні результати при зміні точки зору, але стикалися з проблемами при наявності змін в освітленні та інших значних варіаціях вигляду місць [1]. Загалом, в більшості випадків локальні дескриптори показують кращі результати, ніж глобальні, але ціною цьому є велика обчислювальна складність процесу їх зіставлення.

2.1.2 Глобальні дескриптори

На відміну від локальних, глобальні дескриптори не потребують розрахунку ключових точок, натомість вони описують зображення цілком та повертають 1 вектор $d \in \mathbb{R}^p$ (Рис. 2.4). Представником класичних глобальних дескрипторів є GIST. Сучасні глобальні дескриптори всі базуються на глибинному навчанні, наприклад NetVLAD, CosPlace [20], MixVPR [17].

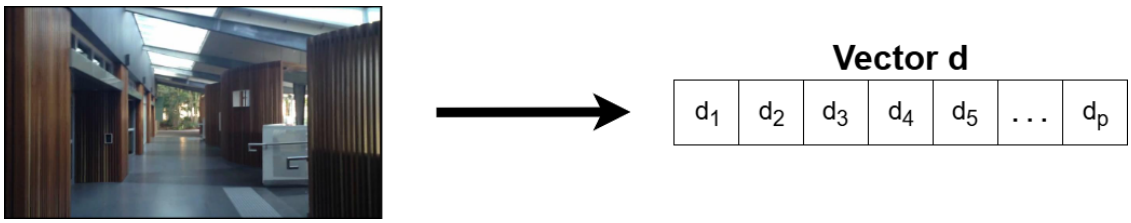


Рис. 2.4. Глобальний дескриптор

Глобальні дескриптори, на відміну від локальних, показують добрі результати при змінах зовнішнього вигляду місця, але мають проблеми зі змінами точки зору [1]. Розмір глобальних дескрипторів зазвичай більший, ніж у локальних, наприклад, розмірність дескриптора NetVLAD та MixVPR – 4096, а CosPlace – від 32 до 2048 [19].

2.1.3 Агрегація локальних дескрипторів (BoVW та VLAD)

Як вже було згадано в секції 2.1.1, співставлення локальних дескрипторів є обчислювально складним процесом у порівнянні з глобальними дескрипторами. Одним із найбільш популярних способів уникнення цієї проблеми є проведення агрегації локальних дескрипторів у один глобальний. Для агрегації зазвичай використовують модель bag of visual words (BoVW) або vector of locally aggregated descriptors (VLAD). Основна ідея обох моделей одна [15]:

1) Розраховуємо набори локальних дескрипторів $D_j = \{d_t\}_{t=1}^{t=N_j}$ для зображень $I_j \in DB$.
2) Формуємо словник моделі. Словник складається з набору візуальних слів (дескрипторів). Для того, щоб сформувати словник зазвичай проводять кластеризацію усіх дескрипторів з першого пункту методом к-середніх [38] та в якості словника беруть отримані k центрів кластерів $\{c_i\}_{i=1}^{i=k}$, тобто c_i – центр кластеру C_i, $i = \overline{1,k}$.
3) Розрахунок глобального дескриптора для зображень $I_i \in DB \cup Q$ проходить за наступними формулами (B – BoVW, V – VLAD):

$$B_i(n) = \sum_{t=1}^{N_i} a_{tn}, \quad n = \overline{1,k}$$

$$V_i(n) = \sum_{t=1}^{N_i} a_{tn}(d_{it} - c_n), \quad n = \overline{1,k}$$

$$a_{tn} = \begin{cases} 1, & d_{it} \in C_n, \\ 0, & d_{it} \notin C_n, \end{cases}, \quad n = \overline{1,k}$$

Де N_i – кількість дескрипторів в зображенні I_i, а B_i, V_i – відповідні дескриптори, $B_i(n)$ – $n-$й елемент вектора B_i, а $V_i(n)$ – $n-$й стовпець матриці V_i.

Компоненти вектора B_i також можна зважити за допомоги idf (inverse document frequency) [15]. Це дозволяє зменшити вплив тих візуальних слів, які часто зустрічаються та в результаті не надають багато інформації про місце.

Після розрахунку також нормалізуємо отримані вектори:

$$B_i := \frac{B_i}{||B_i||_2} \qquad V_i := \frac{V_i}{||V_i||_2}, \quad i = \overline{1,|DB \cup Q|}$$

В результаті отримуємо глобальні дескриптори розмірності k для BoVW та $k \times d$ для VLAD, де d – вимір локального дескриптора, k – розмір словника. Можемо побачити в 3 пункті, що BoVW враховує лише те, до якого кластера належить відповідний дескриптор (гістограма), в той час як VLAD ще враховує відстань до центру цього кластера [15]. Дескриптор VLAD можна перетворити до розміру kd шляхом конкатенації $V_i(n)^T$ в один вектор ($V_i(n)$ – вектор стовпець).

Більшість сучасних підходів застосовують NetVLAD – шар, що додається на кінці згорткової нейромережі та проводить агрегацію її вихідних даних.

2.2 Descriptor similarity

На цьому кроці йде порівняння дескрипторів. Процес для різних типів дескрипторів відрізняється, тому розглянемо обидва випадки.

2.2.1 Локальні дескриптори

Співставлення дескрипторів зазвичай виконується за допомогою алгоритмів типу nearest neighbours (NN), mutual nearest neighbours (MNN) або approximate nearest neighbours (ANN) – останні особливо актуальні для великих баз даних. Також активно використовуються навчені співставники, зокрема LightGlue [39] та SuperGlue [40]. Для підвищення точності результати співставлення часто проходять додаткову обробку за допомогою геометричної верифікації, зокрема методом RANSAC або його покращеними варіантами (наприклад, MAGSAC). Ці методи дозволяють відфільтрувати помилкові відповідності, виходячи з оцінки геометричного перетворення між зображеннями.

Співставлення дескрипторів

1) Nearest neighbours (NN). Перебирає усі пари для кожного дескриптора $I_j \in Q$ та знаходить найближчі (Рис. 2.5 – 2.8).
2) Mutual Nearest neighbours (MNN). Перебирає усі пари для кожного дескриптора $I_j \in Q$ та знаходить такі пари, де обидва дескриптори є найближчими один до одного серед відповідних множин.
3) Навчені підходи SuperGlue та LightGlue. SuperGlue співставляє локальні дескриптори шляхом розв'язання диференційованої транспортної задачі, витрати на яку прогнозуються графовою нейронною мережею. Більш швидкою, модифікованою версією SuperGlue є LightGlue.
4) Approximate nearest neighbors (ANN). ANN не гарантує співставлення найближчих дескрипторів, в обмін на більшу швидкість (Рис. 2.9, 2.10).
5) Lowe ratio [8] полягає в пошуку двох найближчих дескрипторів d_1 та d_2 до заданого d_0, після чого порівнюються відповідні відстані $dist_1$ та $dist_2$. Якщо виконується умова $dist_1 < r\, dist_2$, де $r \in (0, 1)$ — d_0 вважається відповідним до d_1, інакше відповідність відкидається (Рис. 2.5-2.10). В подальшому застосування Lowe ratio до NN і MNN позначається як SNN та SMNN відповідно.

Рис. 2.5. Візуалізація результату SNN ($r = 0.8$)

Рис. 2.6. Візуалізація результату SNN ($r = 0.5$)

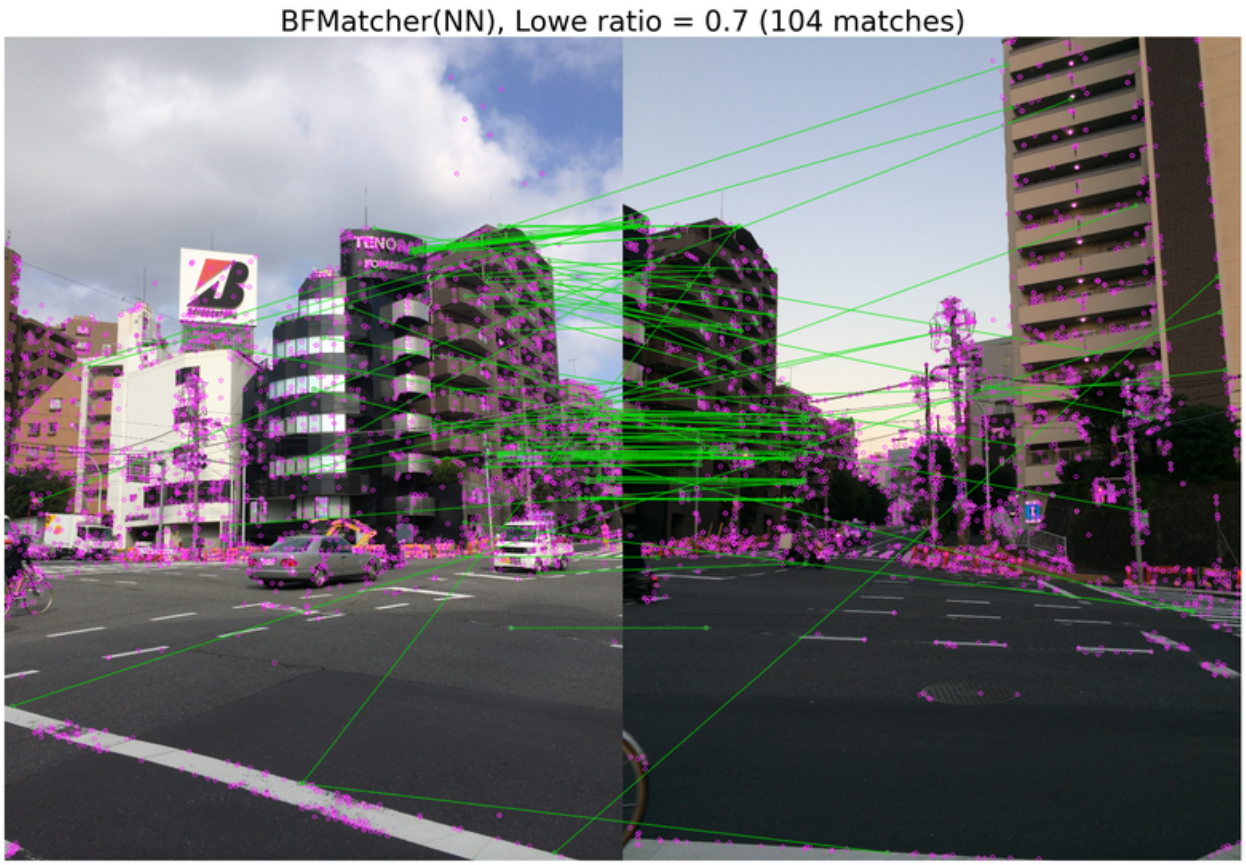


Рис. 2.7. Візуалізація результату SNN ($r = 0.7$)

Рис. 2.8. Візуалізація результату SNN ($r = 0.5$)

Рис. 2.9. Візуалізація результату ANN + Lowe ratio ($r = 0.7$)

Рис. 2.10. Візуалізація результату ANN + Lowe ratio ($r = 0.5$)

Після зіставлення дескрипторів ми, наприклад, можемо використати кількість співпадінь для кожної пари зображень для побудови матриці подібності $S^{|Q|\times|DB|}$, за допомогою якої далі зможемо робити висновки щодо схожості відповідних зображень.

Геометрична верифікація

Оскільки в парі з кожним дескриптором йде ключова точка, то варто також розглянути алгоритми, які цю інформацію використовують. Такі алгоритми зазвичай використовують в парі з попередньо розглянутими, як другий етап фільтрації пар дескрипторів.

RANSAC [41] – ітеративний алгоритм, який на кожній ітерації генерує гіпотезу геометричної моделі та відбирає відповідні їй пари ключових точок. Основна ідея полягає в тому, щоб, виходячи з початкових відповідностей дескрипторів, ітеративно будувати геометричне перетворення між зображеннями і відсівати ті пари ключових точок, що не вкладаються в поточну модель. Зазвичай алгоритм складається з чотирьох етапів (Рис. 2.11):

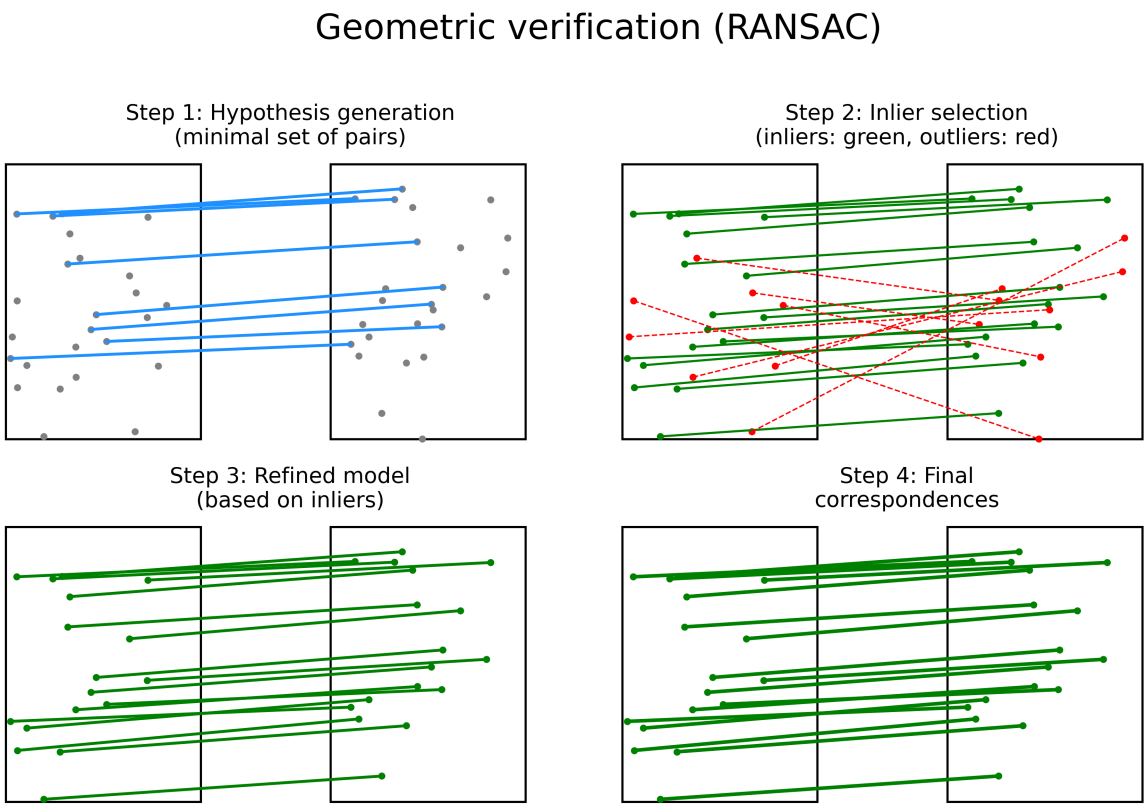


Рис. 2.11. Спрощена діаграма геометричної верифікації для пошуку фундаментальної матриці

1) Генерація гіпотез. Випадковим чином вибирають мінімальний набір пар точок (наприклад, 8 для фундаментальної матриці, 4 для гомографії). На їх основі обчислюють параметри моделі (фундаментальної матриці або гомографії).
2) Оцінка та відбір інлайерів (inliers, узгоджені точки). Для кожної пари ключових точок рахують геометричну помилку (епіполярна відстань або reprojection error). Всі пари, у яких помилка нижча за попередньо заданий поріг, вважають інлайерами.
3) Побудова остаточної моделі. Найкращу гіпотезу (з найбільшою кількістю інлайерів) уточнюють методом найменших квадратів на всіх інлайерах.
4) Фільтрація пар. Підсумкові інлайери використовують як остаточний набір відповідностей для подальших етапів.

Розглянемо приклад роботи геометричної верифікації на одній з раніше розглянутих пар (Рис. 2.12). Знайдемо співпадаючі дескриптори за допомогою SNN ($r = 0.8$) (Рис. 2.13) та на основі цих пар проведемо геометричну верифікацію алгоритмом RANSAC (Рис. 2.14).

Рис. 2.12. Пара тестових зображень з 24/7 Tokyo

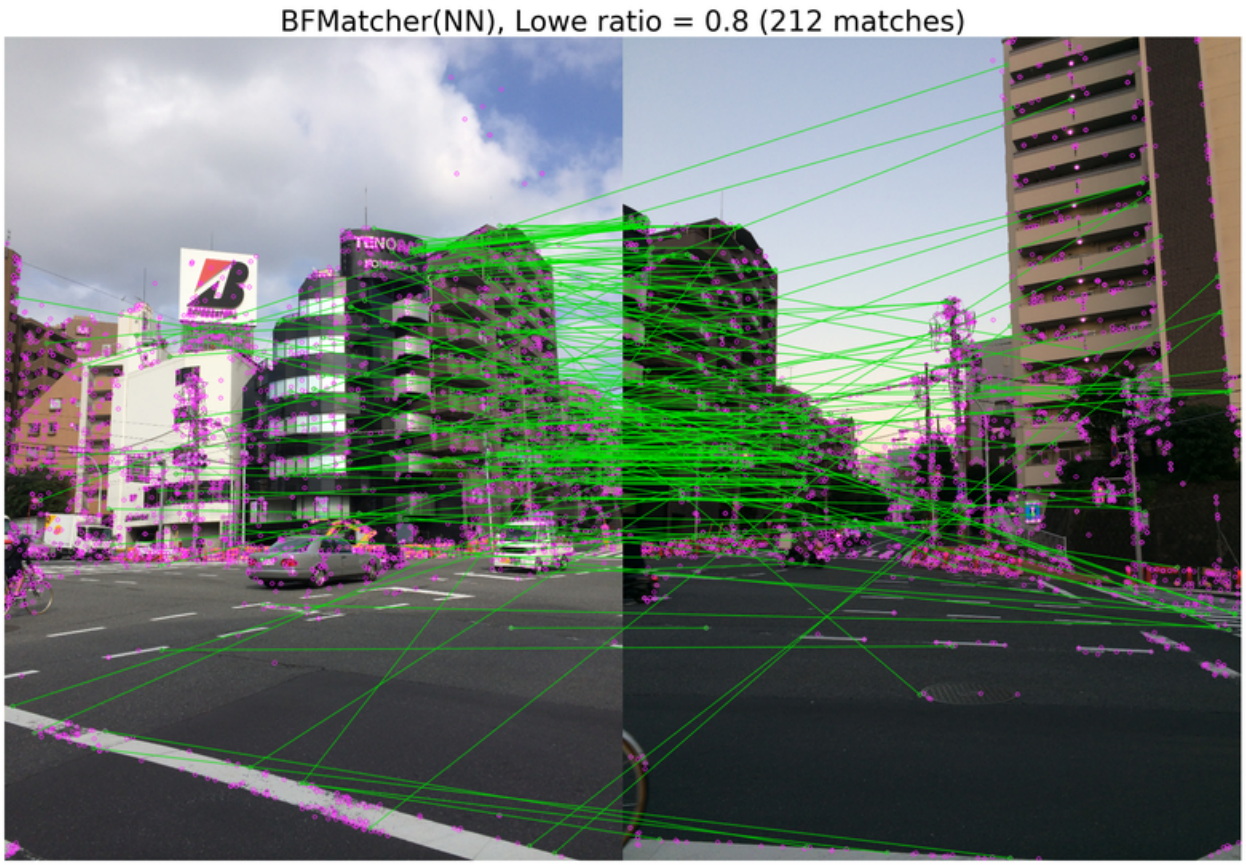


Рис. 2.13. Візуалізація результату SNN ($r = 0.8$)

Рис. 2.14. Візуалізація результату SNN ($r = 0.8$) + RANSAC

Можемо побачити (Рис. 2.14), що RANSAC вдалося відкинути неправильні пари, та кількість результуючих пар вища, ніж у випадку зниження значення порогу r (Рис. 2.8).

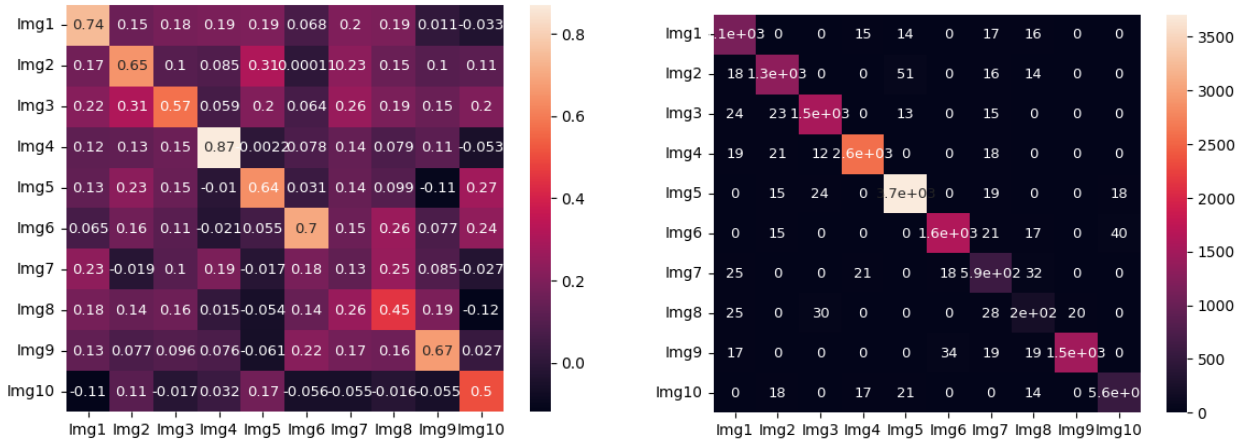
Матриця подібності $S^{|Q|\times|DB|}$ для цього випадку розраховується аналогічно: кількість співпадінь (в даному випадку кількість інлайерів) для кожної пари зображень, як елемент матриці.

2.2.2 Глобальні дескриптори

Будемо вважати, що дескриптор глобальний, тобто, якщо ми працюємо з локальними дескрипторами, то була проведена агрегація у глобальний. Тоді йде розрахунок або відстані між дескрипторами (відстань Евкліда), або схожості векторів (косинус подібності). Результат цього кроку є матриця подібності $S^{|Q|\times|DB|}$ [25]. Далі на основі цієї матриці можна буде робити висновки щодо вибору кандидатів.

2.3 Matching

Для пошуку пар зображень з одного місця ми використовуємо отриману на минулому кроці матрицю подібності $S^{|Q|\times|DB|}$ (Рис. 2.15).



(а) Глобальні дескриптори. Елементи – косинус-подібність між дескрипторами.

(б) Локальні дескриптори. Елементи – кількість знайдених пар.

Рис. 2.15. Приклад матриці подібності де у Q та DB по 10 зображень та правильні пари лежать на головній діагоналі.

Кожний рядок матриці $S^{|Q|\times|DB|}$ відповідає відповідному запиту $I_j \in Q$, а стовпець – зображенню з бази даних $I_i \in DB$. Отже, подібність двох зображень знаходиться на перетині відповідних рядка та стовпця.

Якщо необхідний 1 найкращий кандидат, то у відповідність кожному поточному зображенню $I_j \in Q$ обирається зображення з бази даних $I_i \in DB$ з найбільшим значенням у відповідній стрічці. Якщо є необхідність обрати більшу кількість кандидатів, то замість найкращого кандидата, наприклад, обирають усіх, для кого значення на перетині більші за певне порогове значення: $s_{ij} < \theta$, чи просто перші n кандидатів [25].

2.3.1 Гібридний підхід

У випадках, коли йде відбір більш ніж одного кандидата за допомогою глобальних дескрипторів, є можливість провести переранжування цих кандидатів шляхом розрахунку локальних дескрипторів або використовуючи ті, що збереглися з минулих етапів. Далі зіставляємо дескриптори методами, розглянутими в секції 2.2.1 та додатково проводимо геометричну верифікацію отриманих пар. Далі зображення переранжуються на основі кількості інлайерів, знайдених RANSAC [19].

2.4 Оцінка результату роботи

Оцінка якості роботи проходить за допомоги розрахунку різних показників. Основними критеріями є:

1) Precision. Precision показує відношення правильних пар, які вдалося знайти до загальної кількості знайдених пар. precision $\in [0, 1]$
2) Recall. Recall показує відношення правильних пар, які вдалося знайти до загальної кількості правильних пар. recall $\in [0, 1]$
3) AUPRC. Це площа під кривою залежності precision та recall. AUPRC $\in [0, 1]$
4) Recall@K. Ця метрика показує частку поточних зображеннь, що збігаються хоча б з одним зображення серед K кращих кандидатів з бази даних. Recall@K $\in [0, 1]$. Варто відмітити, що Recall@1 = Precision.

РОЗДІЛ 3

АНАЛІЗ АЛГОРИТМІВ VPR

Алгоритми VPR будуються за схемою, вказаною у главі 2. Тобто для побудови щонайменше необхідні дескриптор $F : \mathbb{I} \longrightarrow \mathbb{D}$ та алгоритм для порівняння отриманих репрезентацій $d \in \mathbb{D}$. Наразі вибір дескрипторів та алгоритмів порівняння є досить великим, тому будуть обрані декілька передових та доступних варіантів. Як вже було показано в минулих главах, загалом можна виділити 3 основні типи підходів:

1) На основі глобальних дескрипторів. Кожному зображенню ставимо у відповідність 1 вектор D розміру p.
2) На основі Локальних дескрипторів. Кожному зображенню ставимо у відповідність n пар: ключова точка (x,y) та вектор d розміру M.
3) Гібридний підхід з використанням глобальних дескрипторів для формування початкового набору кандидатів (відбираються k найкращих кандидатів для кожного запиту), після чого здійснюється додаткове переранжування з використанням локальних дескрипторів.

Розглянемо кожен з цих підходів та відповідні компоненти, що будуть використані.

3.1 На основі глобальних дескрипторів

Це найпростіший з 3 підходів. В якості глобального дескриптора було обрано CosPlace. У CosPlace можна обирати глибоку нейромережу, яка лежить в основі дескриптора, а також розмір результуючого вектора. Для різних тестів було обрано дескриптори на основі ResNet50 і ResNet152 із розмірами дескриптора 512 і 2048 відповідно. Для порівняння цих репрезентацій буде використано косинус подібності:

$$S_{cos}(a,b) = \frac{a \cdot b}{|a||b|}, \quad |a| = \sqrt{a \cdot a}, \quad |b| = \sqrt{b \cdot b}$$

Тобто, чим менший кут між векторами a, b – тим більше $S_{cos}(a,b)$, а максимальне значення $S_{cos}(a,b) = 1$ буде досягатись, коли дескриптори a, b однакові. Але оскільки дескриптори CosPlace вже нормалізовані, то можна значно спростити формулу:

$$S_{cos}(a,b) = a \cdot b, \quad |a| = \sqrt{a \cdot a} = 1, \quad |b| = \sqrt{b \cdot b} = 1$$

З цією формулою буде набагато простіше працювати. Далі на основі цього косинуса подібності будується матриця попарних порівнянь $S^{|Q|\times|B|}$.

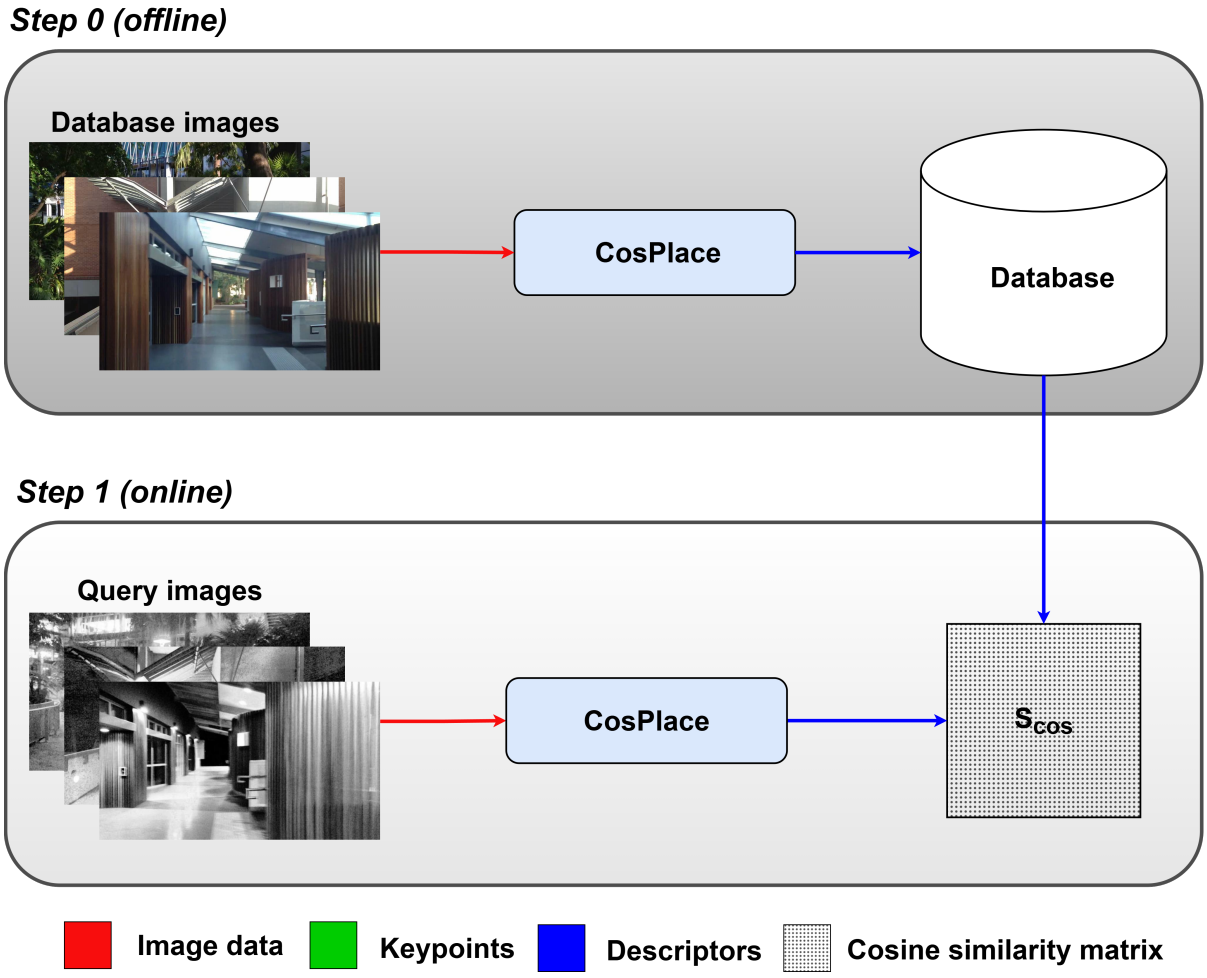


Рис. 3.1. Візуалізація підходу

Розрахунок дескрипторів для бази даних проводиться один раз перед початком роботи системи (offline). Це виконується для уникнення повторного розрахунку дескрипторів для зображень бази даних. Варто нагадати, що ми розглядаємо проблему, в якій база даних не змінюється під час роботи системи.

Для вибору кращого кандидата з DB для запиту $I_j \in Q$ просто обирається елемент з найбільшим значенням в j-му рядку $S^{|Q|\times|B|}$.

3.2 На основі локальних дескрипторів

В якості локальних дескрипторів було обрано DISK та DeDoDe v2 (далі – DeDoDe)[42]. Обидва дескриптори розраховували по 6000 ключових точок. Розмір дескриптора для DISK – 128, а для DeDoDe – 256. Для DISK параметр checkpoint був обраний epipolar. Для DeDoDe ваги детектора – L-C4-v2, а дескриптора – G-upright. Для зіставлення дескрипторів було використано SMNN з $r = 0.98$, далі для ключових точок, отриманих пар, було проведено геометричну верифікацію за допомогою сучасної модифікації RANSAC – MAGSAC++. Кількість пар після геометричної верифікації була взята в ролі елемента матриці подібності.

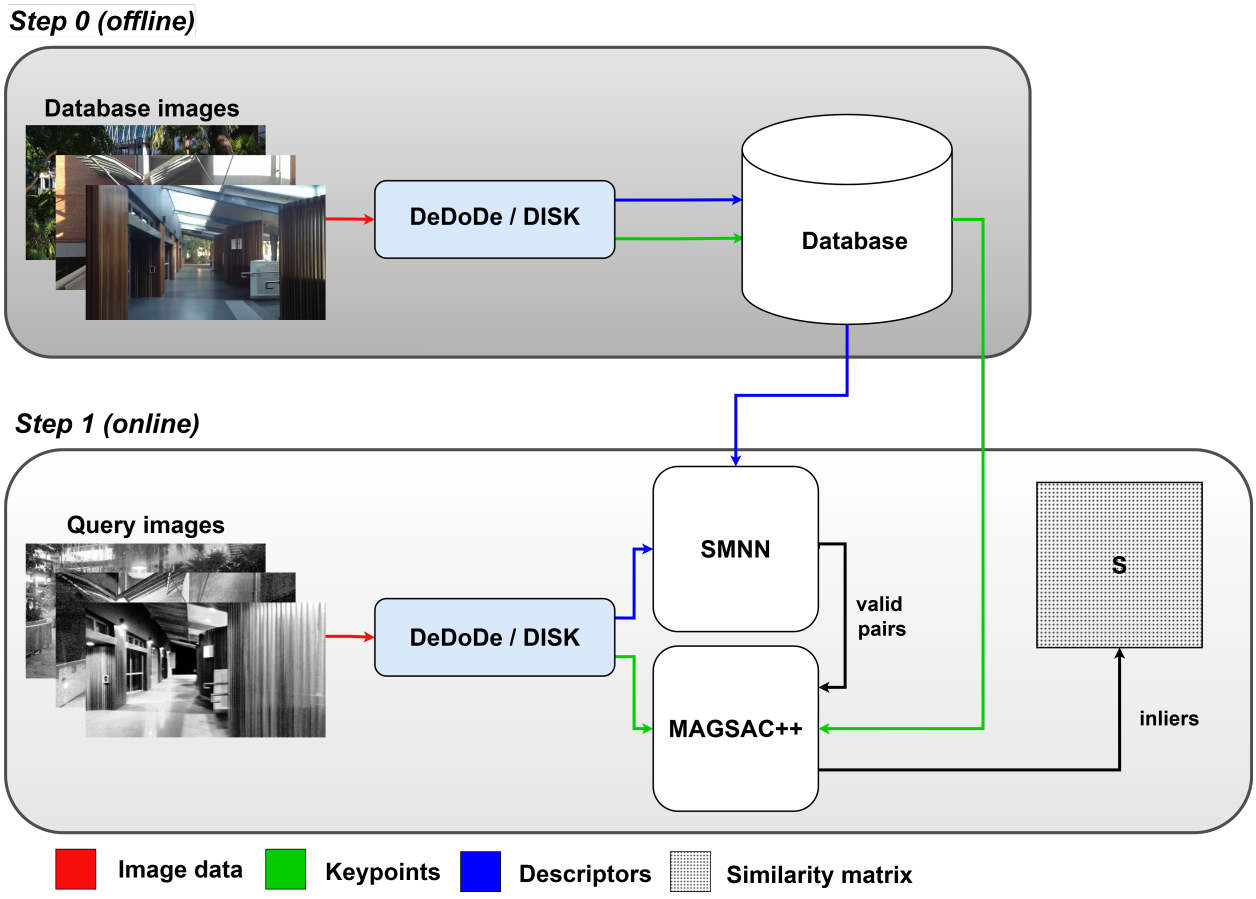


Рис. 3.2. Візуалізація підходу

Так само, як і для минулого випадку, розрахунок дескрипторів для

бази даних проводиться один раз перед початком роботи системи (offline). Для вибору кращого кандидата з DB для запиту $I_j \in Q$ просто обирається елемент з найбільшим значенням в j-му рядку $S^{|Q|\times|B|}$.

3.3 Гібридний підхід

Як видно з попередніх розділів, обчислювальні витрати алгоритмів із локальними дескрипторами значно перевищують ті, що потрібні для глобальних дескрипторів. Так, в даному прикладі глобальний дескриптор має розмір лише 512 чи 2048, тоді як локальні – до 6000 пар (2 вимірна ключова точка + вектор розмірності 128 або 256). До того ж стадії зіставлення локальних дескрипторів і геометричної верифікації набагато більш затратні, ніж простий розрахунок косинусної подібності, тому загальна складність VPR на основі локальних ознак істотно вища.

Гібридний підхід (Рис. 3.3) дозволяє зменшити кількість інформації, яку потрібно обробляти локальним дескриптором.

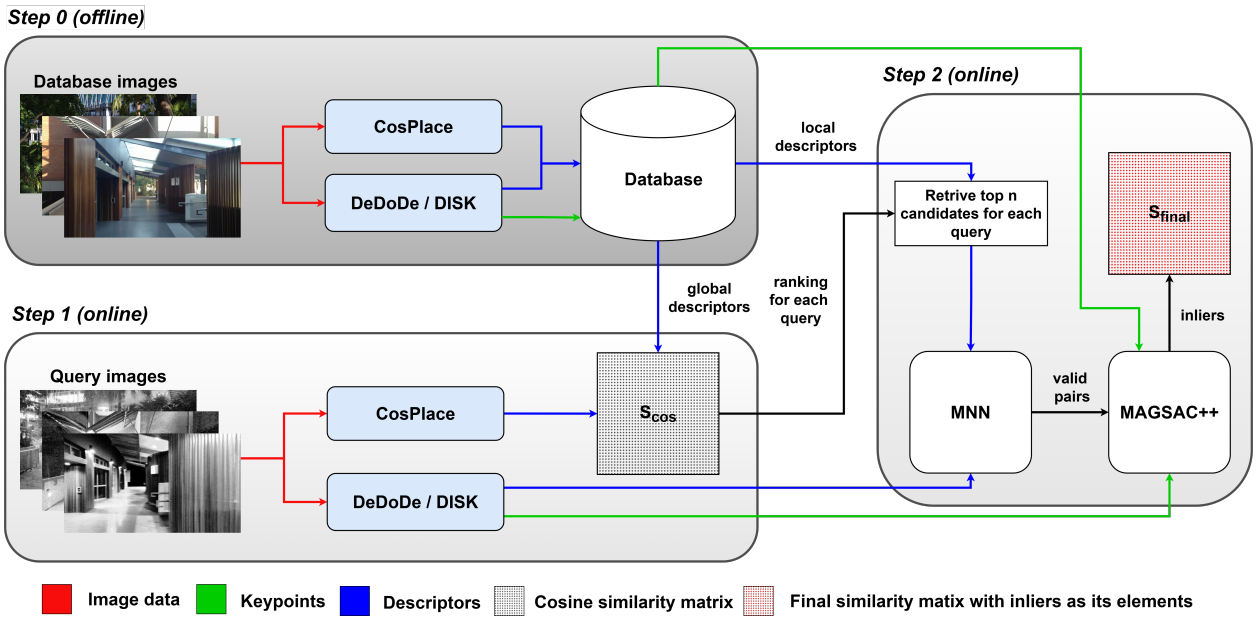


Рис. 3.3. Візуалізація гібридного підходу. S_{cos} – матриця подібності для глобальних дескрипторів, S_{final} – фінальна матриця подібності, на основі якої обираються кінцеві кандидати для запитів.

Основна ідея полягає в тому, що ми використовуємо швидкі глобальні дескриптори, щоб відібрати $n < |DB|$ найкращих кандидатів для кожно-

го запиту. Далі використовуємо локальні дескриптори, але замість того, щоб розглядати кожне зображення з DB, для кожного запиту ми будемо дивитися лише на його n найкращих кандидатів. Тобто фактично гібридний підхід комбінує глобальні та локальні дескриптори в один алгоритм. Наслідком такого підходу є те, що при сталому n швидкість роботи локального дескриптора залежить лише від кількості запитів, а початковий розмір бази даних вже не впливає на швидкість роботи. Це є дуже важливим досягненням, адже розміри баз даних можуть бути дуже великими. В гібридному підході глобальні дескриптори фактично опрацьовують усю DB та передають локальним дескрипторам лише малі фрагменти цієї бази даних для кожного запиту. Важливо розуміти, що серед цих n кандидатів, які були отримані на основі глобальних дескрипторів, не завжди будуть потрапляти правильні пари для кожного запиту, тому важливо правильно обирати число n. Відповідно, другий наслідок – якість результату 2 етапу (локальний дескриптор) обмежена тим, наскільки добре відпрацював 1 етап (глобальний дескриптор).

Глобальні та локальні дескриптори будуть обиратися такі самі, як і в минулих алгоритмах. Додатково було додано, як другий етап неявний локальний дескриптор LoFTR, або як їх називають в літературі – dense matcher. Вони працюють трохи інакше, тому розглянемо іншу діаграму (Рис. 3.4).

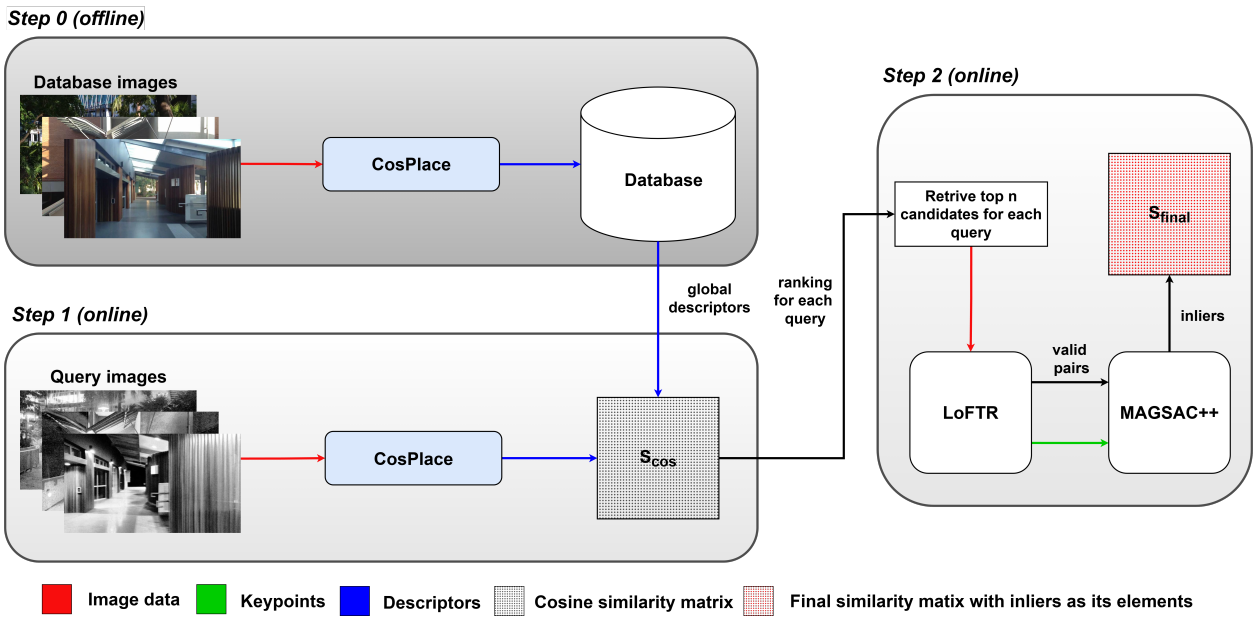


Рис. 3.4. Візуалізація гібридного підходу з LoFTR.

Dense matcher приймає на вхід пару зображень $(I_Q, I_{DB}) \in Q \times DB$ та повертає набір співпадаючих ключових точок між ними. Він одночасно виконує роль локального дескриптора й співставника. Як видно з 2 кроку діаграми (Рис. 3.4), ми не зберігаємо локальних дескрипторів чи ключових точок в базі даних. Це відбувається через те, що LoFTR встановлює відповідності, використовуючи контекст обох зображень; ключові точки та їх дескриптори формуються під час зіставлення й не є стабільними ознаками лише одного зображення. Через це необхідно витрачати багато ресурсів на кожну пару, і саме тому LoFTR не буде розглядатись окремо від гібридного підходу, як інші локальні дескриптори.

Для порівняння глобальних дескрипторів було використано косинус подібності, для зіставлення локальних – MNN, а для геометричної верифікації – MAGSAC++. Кількість кандидатів, для переранжування n було обрано 10.

РОЗДІЛ 4

РЕЗУЛЬТАТИ ЕКСПЕРИМЕНТІВ

Загалом було проведено тести на трьох наборах даних. На кожному наборі було протестовано алгоритми, описані в главі 3, а саме (Табл. 4.1):

№	Глобальний дескриптор	Локальний дескриптор	Зіставлення
1	CosPlace (512)	-	-
2	CosPlace (2048)	-	-
3	CosPlace (512)	LoFTR+MAGSAC	
4	CosPlace (2048)	LoFTR+MAGSAC	
5	-	DeDoDe	SMNN+MAGSAC
6	CosPlace (512)	DeDoDe	MNN+MAGSAC
7	CosPlace (2048)	DeDoDe	MNN+MAGSAC
8	-	DISK	SMNN+MAGSAC
9	CosPlace (512)	DISK	MNN+MAGSAC
10	CosPlace (2048)	DISK	MNN+MAGSAC

Табл. 4.1. Розглянуті алгоритми VPR

4.1 Дані та методика експерименту

Для оцінки алгоритмів було обрано 2 набори даних, а саме Gardens Point [43] та SPED [44].

Перші два тести проводились на наборі даних Gardens Point. У першому тесті (Табл. 4.2) Q складалася з нічного проходу парком правою стороною, а DB – з денного проходу тією ж стороною (Рис. 4.1). У другому тесті (Табл. 4.3) Q залишалася незмінною, а DB містила денний прохід парком лівою стороною (Рис. 4.2). Всі множини містять по 200 зображень та $I_j \in Q$ відповідає локації $I_i \in DB$, коли $|j - i| \leq 1$. Такий вибір пов'язаний з тим, що послідовні фото мають дуже високий візуальний перетин та знаходяться досить близько одне до одного. Третій тест (Табл. 4.4) проводився на

наборі даних SPED. У ньому множини Q та DB містять по 607 зображень різних місць зі змінами в освітленні та сезоні (Рис. 4.3). Для SPED $I_j \in Q$ відповідає локації $I_i \in DB$, коли $j = i$.

Для оцінки результатів використовувалась метрика Recall@K для значень K: 1, 5, 10 та час (загальна сума часу виконання online кроків).

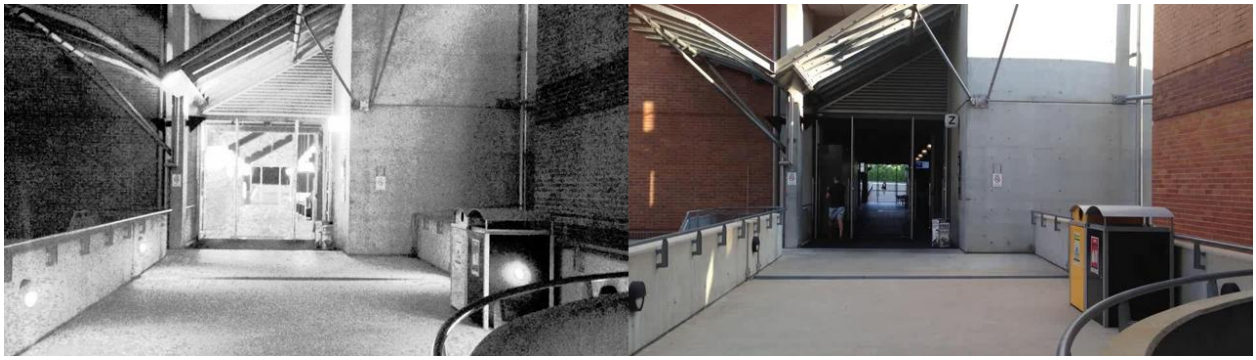


Рис. 4.1. Приклад правильної пари зображень (Gardens Point №1)

Рис. 4.2. Приклад правильної пари зображень (Gardens Point №2)

Рис. 4.3. Приклад правильної пари зображень (SPED)

4.2 Аналіз результатів

Розглянемо результати описаних алгоритмів (Табл. 4.1) на трьох тестах. Варто зауважити, що в тестах розрахунок дескрипторів та зіставлення відбувалися на GPU, а геометрична верифікація – на CPU.

4.2.1 Тест 1

Тест демонстрував проблему зміни освітлення. Запити були отримані нічним проходом парком, а DB – денним проходом за тією ж траєкторією.

№	**Алгоритм**	**Recall@1,** %	**Recall@5,** %	**Recall@10,** %	**Час, с.**
1	CosPlace (512)	79.5	98.5	100	1.15
2	CosPlace (2048)	75.5	97	98.5	1.98
3	CosPlace (512) + LoFTR	94.5	99.5	100	158.18
4	CosPlace (2048) + LoFTR	92.5	98.5	98.5	152.18
5	DeDoDe	91.5	99	99	363.04
6	CosPlace (512) + DeDoDe	**97**	100	100	49.16
7	CosPlace (2048) + DeDoDe	95	98.5	98.5	45.36
8	DISK	94.5	99.5	99.5	334.18
9	CosPlace (512) + DISK	94.5	100	100	50.36
10	CosPlace (2048) + DISK	93.5	98.5	98.5	50.71

Табл. 4.2. Результати тесту №1

З таблиці 4.2 можемо зробити декілька основних спостережень стосовно якості та швидкості алгоритмів.

Швидкість

Глобальні дескриптори (№ 1, 2) є значно швидшими за всі інші підходи, локальні дескриптори (№ 5, 8) витратили найбільше часу. Гібридні

підходи (№ 6, 7, 9, 10) витрачають значно менше часу за відповідні локальні дескриптори (в 6-8 разів швидші), але вони також і значно повільніші за відповідні глобальні дескриптори (в 20-40 разів повільніші). Алгоритми з LoFTR (№ 3, 4) виявились найбільш повільними серед гібридних підходів (в 3 рази повільніші).

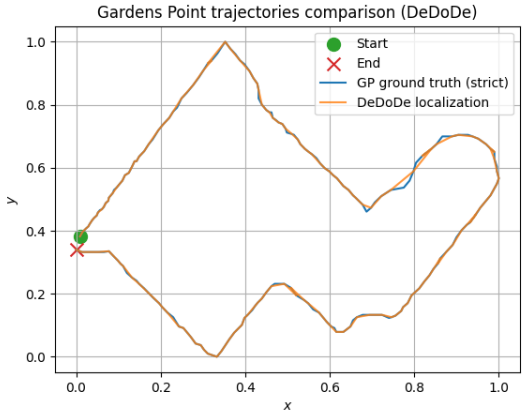
Якість результатів

Глобальні дескриптори показали непоганий результат, але помітно гірший в порівнянні з локальними дескрипторами. Гібридні алгоритми з DeDoDe v2 (№ 6, 7) показали найвищі результати за Recall@1 (Precision) – 97% та 95% відповідно. Це помітне покращення в порівнянні з поодиноким використанням DeDoDe v2 (№ 5), яке досягає результату в 91.5%. Результати гібридного підходу з DISK (№ 9, 10) майже не відрізняються від поодинокого використання DISK (№ 8) та приблизно на рівні з алгоритмами, які використовують LoFTR (№ 3, 4).

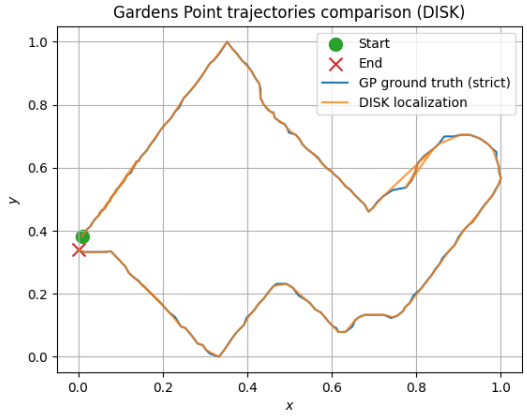
Варто зауважити, що, як зазначалося у попередньому розділі, для гібридних підходів з глобальними дескрипторами попередньо відбираються 10 найкращих кандидатів для переранжування. Це означає, що значення Recall@10 для відповідного глобального дескриптора є теоретично максимальним порогом для всіх показників Recall@K гібридного методу. Оскільки локальні дескриптори застосовуються лише до цих 10 зображень, частка запитів, у яких правильна відповідність присутня серед топ-10, не може збільшитися після повторного ранжування, оскільки змінюється лише порядок, а не склад кандидатів.

Іншою цікавою особливістю є те, що в усіх експериментальних сценаріях глобальний дескриптор CosPlace на основі глибшої архітектури ResNet152 з розміром вектора 2048 (№ 2, 4, 7, 10) демонстрував гірші результати порівняно з компактнішим CosPlace (512), побудованим на ResNet50 (№ 1, 3, 6, 9).

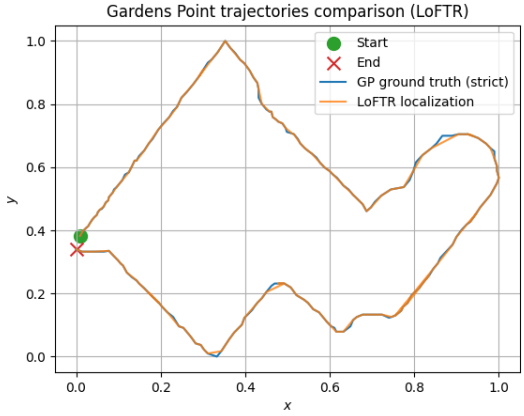
Оскільки обраний набір даних описує траєкторію проходу, то є можливість використовувати надані координати для візуалізації отриманих результатів у вигляді траєкторій (Рис. 4.4).



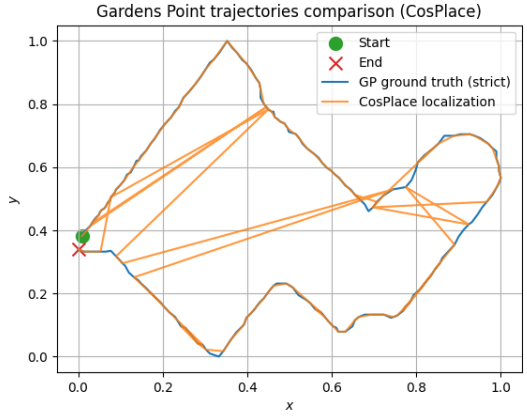
(а) CosPlace(512)+DeDoDe



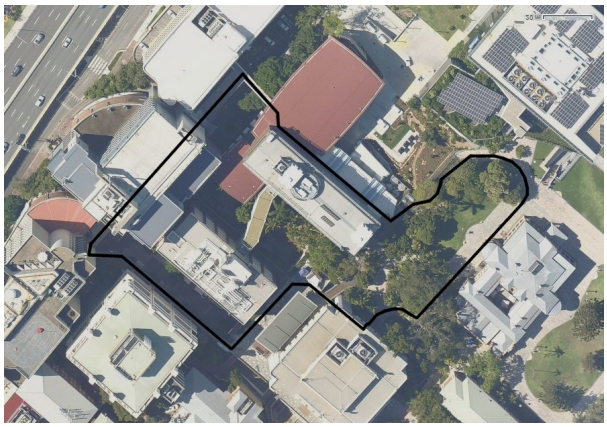
(б) CosPlace(512)+DISK



(в) CosPlace(512)+LoFTR



(г) CosPlace(512)

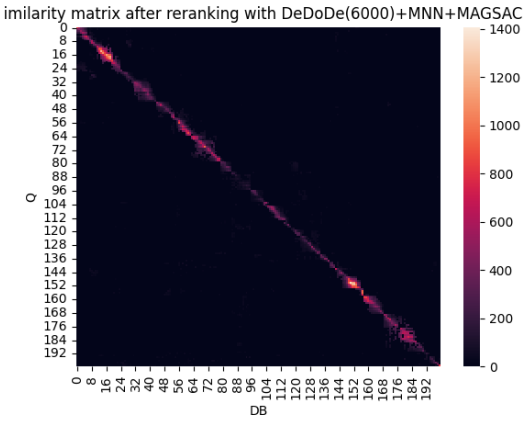


(ґ) Ground truth

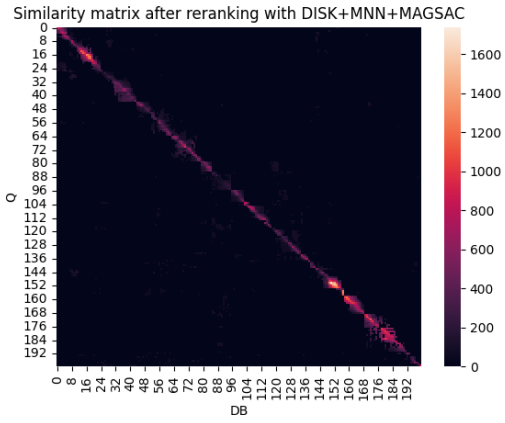
Рис. 4.4. Результати локалізації для різних алгоритмів, де синя траєкторія – істинна (Ground truth), а помаранчева – локалізація алгоритму

Можемо спостерігати, що гібридні підходи дійсно помітно покращили результат глобального дескриптора. Також видно, що більшість помилок у всіх гібридних алгоритмах відбулася в приблизно одних і тих же місцях.

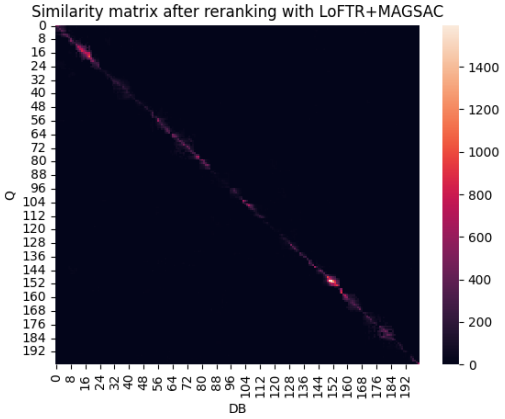
Розглянемо матриці подібності S для різних підходів (Рис. 4.5).



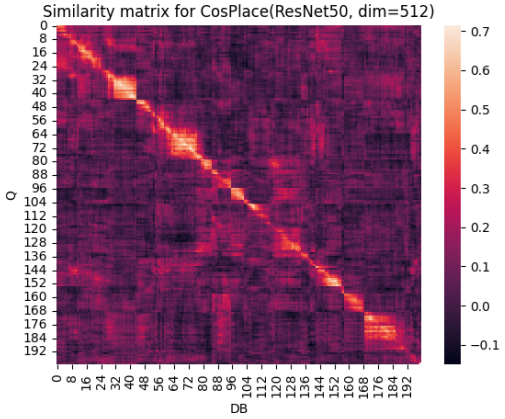
(а) CosPlace(512)+DeDoDe



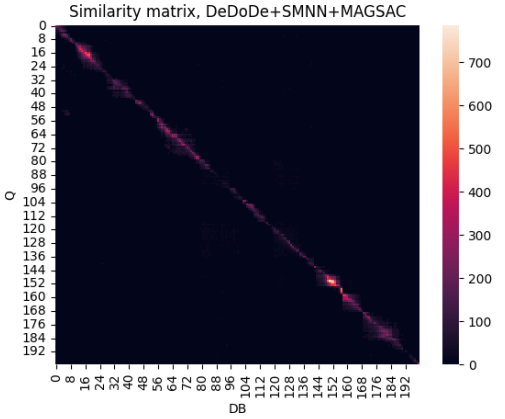
(б) CosPlace(512)+DISK



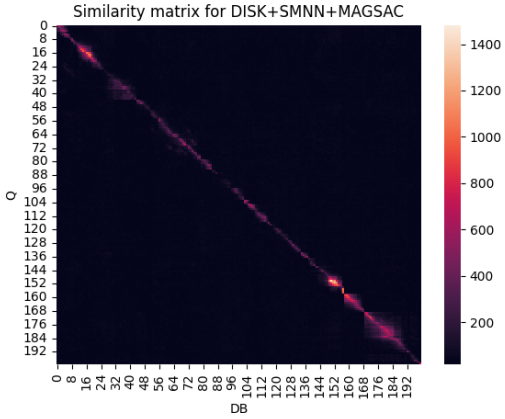
(в) CosPlace(512)+LoFTR



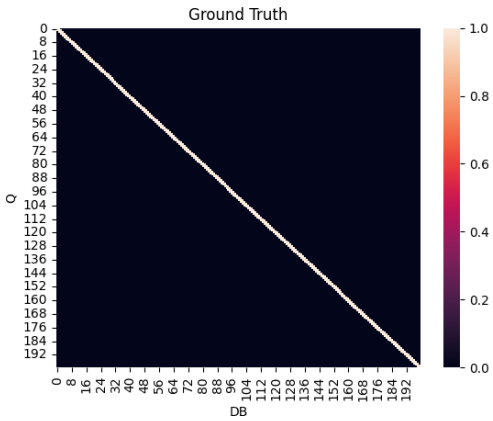
(г) CosPlace(512)



(г) DeDoDe



(д) DISK



(е) Ground truth

Рис. 4.5. Результуючі матриці подібності S

Можемо спостерігати, що всі значення матриць після локальних дескрипторів сконцентровані на головній діагоналі, і відповідно виглядають, як матриця істинних відповідностей (Рис. 4.5e). Для глобального дескриптора (Рис. 4.9г) результат є менш точним, але загалом найбільші значення також сконцентровані поблизу головної діагоналі.

4.2.2 Тест 2

В цьому тесті в додаток до зміни освітлення додалась зміна точки спостереження. Тобто тепер замість обходу парку по правій стороні як у Q, так і в DB, тепер DB – це обхід по лівій стороні.

№	**Алгоритм**	**Recall@1,** %	**Recall@5,** %	**Recall@10,** %	**Час, с.**
1	CosPlace (512)	60	94	98	1.16
2	CosPlace (2048)	57.5	93.5	97	1.98
3	CosPlace (512) + LoFTR	74.5	95	98	151.71
4	CosPlace (2048) + LoFTR	76.5	95.5	97	152.59
5	DeDoDe	71.5	95	96	372.60
6	CosPlace (512) + DeDoDe	**77**	97	98	51.25
7	CosPlace (2048) + DeDoDe	76.5	96.5	97	52.08
8	DISK	66	92.5	94.5	325.68
9	CosPlace (512) + DISK	70.5	94	98	50.65
10	CosPlace (2048) + DISK	67.5	95.5	97	52.05

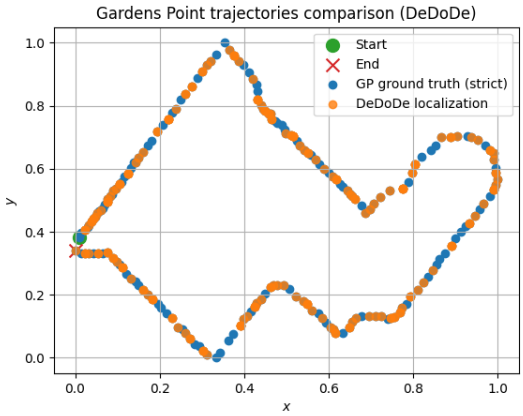
Табл. 4.3. Результати тесту №2

Швидкість

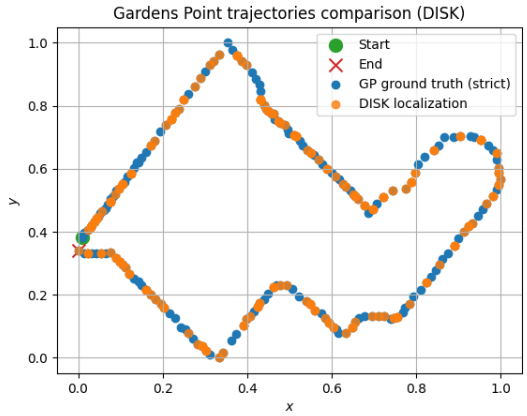
Показники витраченого часу збігаються з тим, що ми спостерігали в минулому тесті, тому одразу перейдемо до порівняння алгоритмів за якістю результатів.

Якість результатів

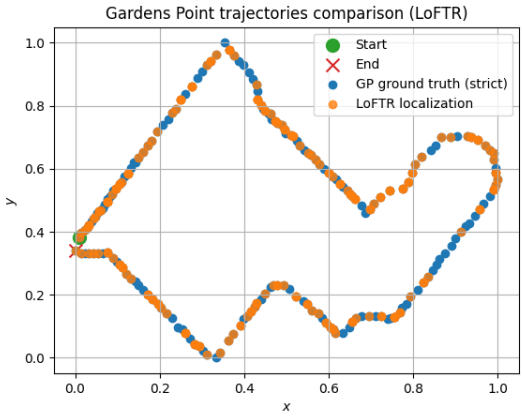
Як видно з результатів (Табл. 4.3), всі значення Recall@1 помітно знизились, що зумовлено ускладненням задачі через різні точки спостереження у множинах Q та DB. Тут також мають місце всі спостереження з першого тесту. Розглянемо отримані результати у вигляді траєкторій (Рис. 4.6).



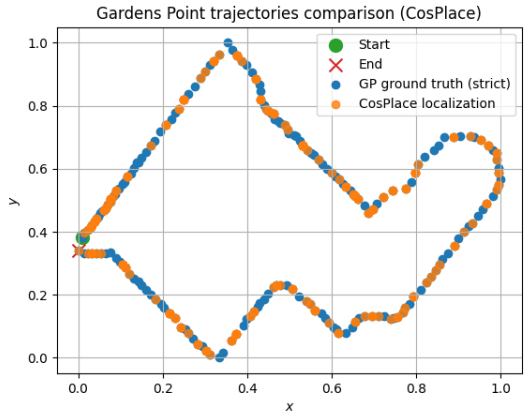
(а) CosPlace(512)+DeDoDe



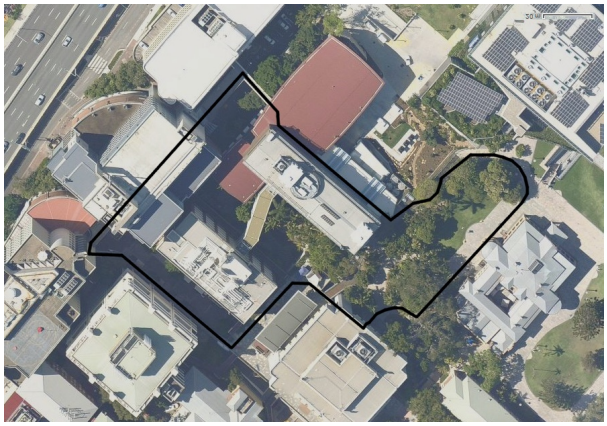
(б) CosPlace(512)+DISK



(в) CosPlace(512)+LoFTR

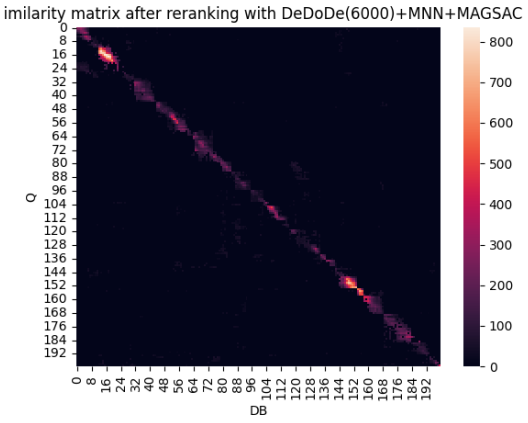


(г) CosPlace(512)

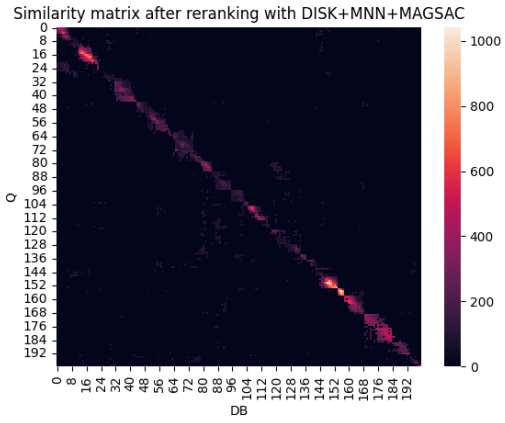


(ґ) Ground truth

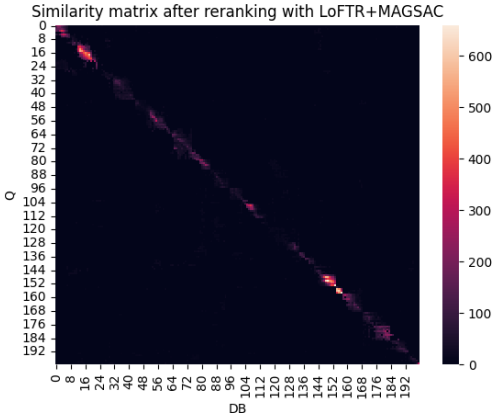
Рис. 4.6. Результати локалізації для різних алгоритмів



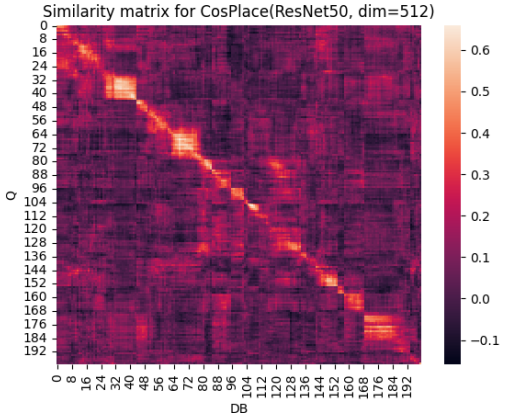
(а) CosPlace(512)+DeDoDe



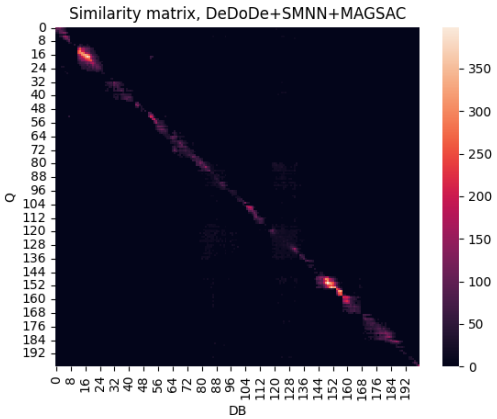
(б) CosPlace(512)+DISK



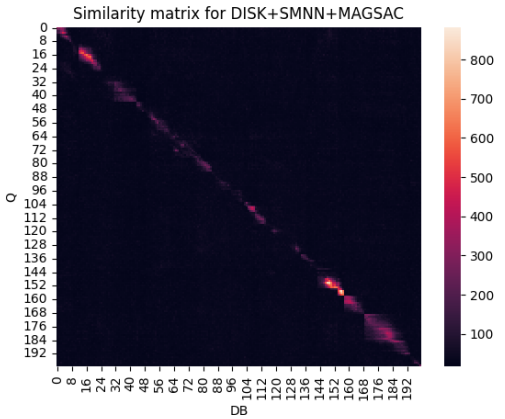
(в) CosPlace(512)+LoFTR



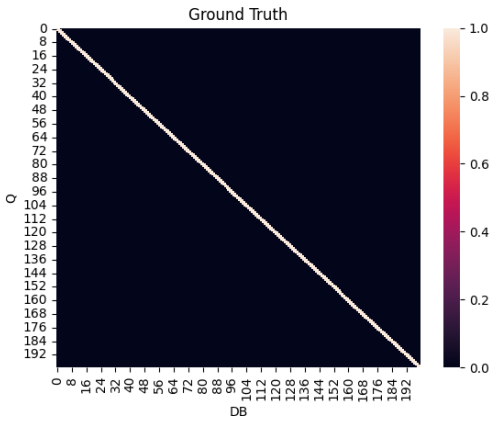
(г) CosPlace(512)



(г) DeDoDe



(д) DISK



(е) Ground truth

Рис. 4.7. Результуючі матриці подібності S

Загалом у цьому випадку результати демонструють подібну картину, проте варто відзначити, що більшість помилок локалізації – як у цьому тесті, так і в попередньому – виникає на ділянках з великими інтервалами, на яких значна частина зображень має мінімальні візуальні відмінності. Це призводить до проблеми перцептивного дублювання (perceptual aliasing).

Матриці подібності (Рис. 4.7), як і в першому тесті, досить близькі до діагональних, що ще раз дозволяє впевнитися в адекватності результатів.

4.2.3 Тест 3

В цьому тесті зображення – це набір не пов'язаних між собою локацій, тобто ми не можемо говорити про траєкторію, та послідовні зображення не мають ніякого візуального перетину. В цьому тесті присутні зміни погоди, освітлення та зміни середовища плином часу.

№	**Алгоритм**	**Recall@1**, %	**Recall@5**, %	**Recall@10**, %	**Час, с.**
1	CosPlace (512)	79.1	89.6	93.1	2.81
2	CosPlace (2048)	79.2	88.8	91.8	6.03
3	CosPlace (512) + LoFTR	**89**	92.8	93.1	449.37
4	CosPlace (2048) + LoFTR	87.8	91.3	91.8	461.47
5	DeDoDe	80.4	88.3	91.1	2954.84
6	CosPlace (512) + DeDoDe	83.7	92.6	93.1	127.31
7	CosPlace (2048) + DeDoDe	83.2	91.3	91.8	126.35
8	DISK	85.7	92.4	93.4	2368.76
9	CosPlace (512) + DISK	85.7	91.9	93.1	156.85
10	CosPlace (2048) + DISK	84.3	89.8	91.8	157.57

Табл. 4.4. Результати тесту №3

Швидкість

Значення часу, отримані у цьому тесті (Рис. 4.4), суттєво вищі порівняно з попередніми, що зумовлено втричі більшим розміром набору даних. Для глобальних дескрипторів і гібридних підходів час обчислення також збільшився приблизно в 3 рази, що є очікуваним. Натомість для локальних дескрипторів спостерігається квадратичне зростання часу — приблизно у 9 разів (3^2), що пояснюється їх складністю: для кожної пари зображень виконується співставлення великої кількості ключових точок.

Якість результатів

У цьому тесті перевага локальних дескрипторів над глобальними практично зникла. Гібридний підхід загалом демонструє лише незначні покращення або навіть невеликі погіршення порівняно з відповідними глобальними дескрипторами. Чітким лідером став гібридний підхід із використанням LoFTR (Рис. 4.8) – найімовірніше, через його здатність виконувати щільне зіставлення, не покладаючись на наперед визначені ключові точки, що дозволяє краще адаптуватися до складних змін візуального середовища.

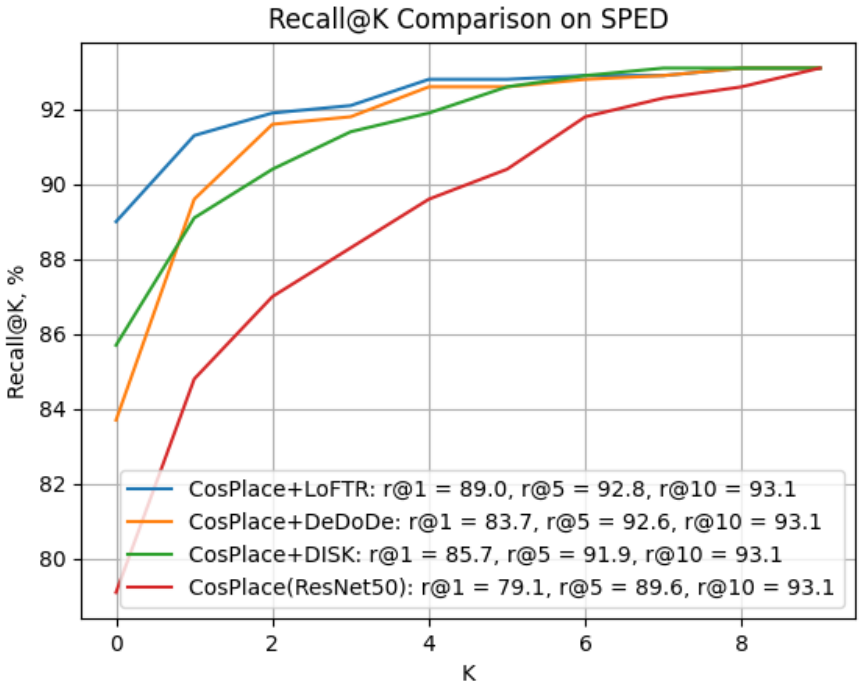
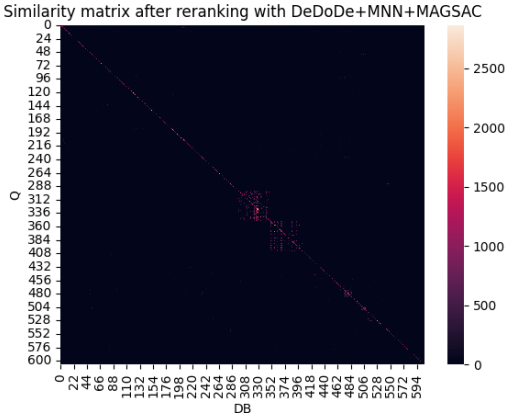
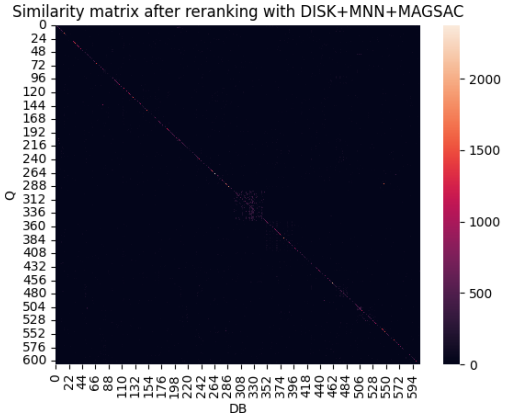


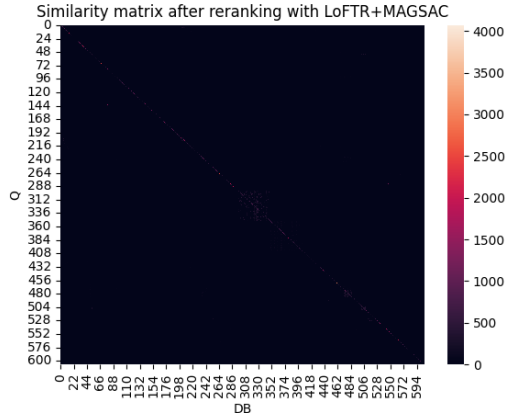
Рис. 4.8. Графіки Recall@K для гібридних підходів та глобального дескриптора



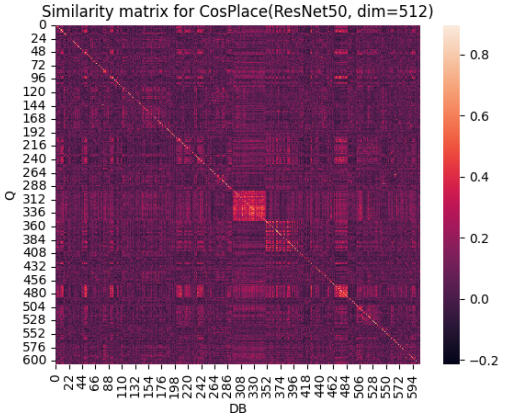
(а) CosPlace(512)+DeDoDe



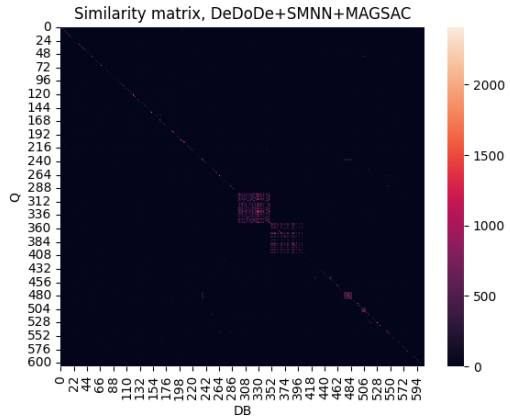
(б) CosPlace(512)+DISK



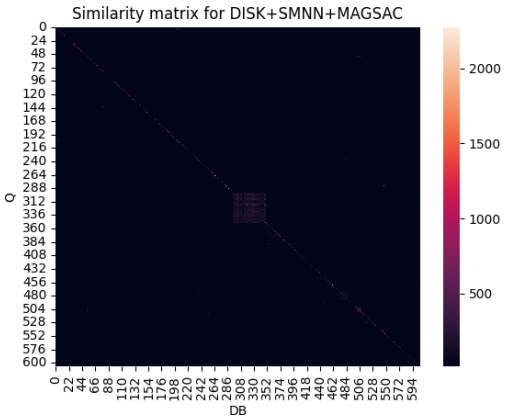
(в) CosPlace(512)+LoFTR



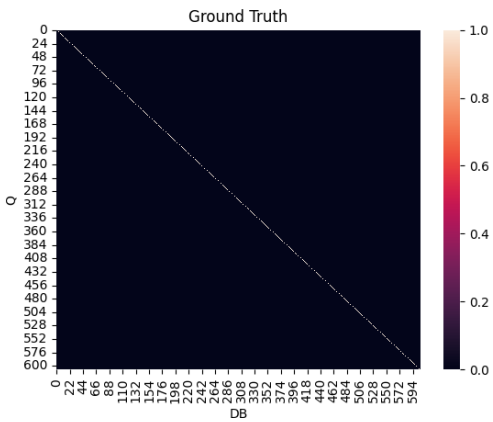
(г) CosPlace(512)



(г) DeDoDe



(д) DISK



(е) Ground truth

Рис. 4.9. Результуючі матриці подібності S

Загалом можна спостерігати, що всі отримані матриці подібності мають загальний вигляд, близький до матриці істинних відповідностей. Водночас помітними є характерні квадратні області вздовж головної діагоналі. Їх поява переважно зумовлена високою візуальною схожістю зображень у певних послідовностях. Наприклад, у сегменті з індексами 295–349 значну частину кадру займає небо, що спричиняє підвищену подібність між зображеннями, навіть за відсутності фактичної відповідності. Подібні ефекти спостерігалися і в попередніх тестах, проте були характерними здебільшого для матриць подібності, побудованих на основі глобальних дескрипторів. Єдиним підходом, якому вдалося уникнути таких артефактів на даному наборі даних, виявився гібридний метод із використанням LoFTR.

ВИСНОВОК

У цій роботі було досліджено задачу візуального виявлення місць. Розглянуто основні складнощі, що виникають під час її розв'язання, а також сучасні підходи до побудови систем VPR. Експериментальні результати показали, що гібридний підхід дозволяє ефективно поєднати високу точність локальних дескрипторів із швидкодією глобальних. Така комбінація забезпечує вдалий компроміс між якістю та швидкістю: час виконання суттєво зменшується порівняно з повністю локальними методами, а точність у більшості випадків залишається на тому ж рівні або навіть перевищує її. Крім того, на відміну від чисто локальних підходів, час роботи гібридних алгоритмів майже не залежить від розміру бази даних завдяки попередньому відбору кандидатів глобальними дескрипторами та сталої кількості локальних зіставлень. Було також відзначено вразливість до перцептивного дублювання, що зумовлена наявністю візуально схожих місць у даних.

Серед протестованих підходів найкращі результати продемонстрував алгоритм CosPlace (512) + LoFTR. Він ефективно впорався з проблемами перцептивного дублювання та забезпечив високу якість локалізації у різних умовах, хоча й був найповільнішим серед гібридних методів. Це свідчить про перевагу трансформерної архітектури щільного зіставника LoFTR як неявного локального дескриптора й підтверджує її перспективність для задач, що потребують високої точності.

Загалом результати підтверджують, що навіть за наявності відносно невеликої бази даних гібридні підходи дають змогу суттєво прискорити роботу локальних дескрипторів без втрати якості, а в більшості випадків – із її покращенням.

Додаток А

Сертифікат за участь у конференції

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

СЕРТИФІКАТ

ФНКТ-0425-16
засвідчує, що
Данило ЖИКУЛ
був учасником II (VIII) міжнародної науково-практичної конференції здобувачів вищої освіти і молодих учених **«Інформаційні технології: теорія та практика»**
2 - 4 квітня 2025 р.
Запоріжжя-Харків-Дніпро
обсягом 24 години

В.о. начальника науково-дослідної частини — *Наталя Висоцька*

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЗАПОРІЗЬКА ПОЛІТЕХНІКА" НАУКОВО-ДОСЛІДНА ЧАСТИНА

СПИСОК ЛІТЕРАТУРИ

1. Lowry, S. *та ін.* Visual Place Recognition: A Survey. *IEEE Transactions on Robotics* **32,** 1—19. ISSN: 1941-0468 (лют. 2016).
2. *Міжнародна науково-практична конференція здобувачів вищої освіти і молодих учених «Інформаційні технології: теорія і практика»* `https://zp.edu.ua/uploads/dept_s&r/2025/conf/4.2/Prohrama_konferentsiyi_ITTP-2025_ukr.pdf`.
3. Arandjelovic, R., Gronát, P., Torii, A., Pajdla, T. & Sivic, J. NetVLAD: CNN Architecture for Weakly Supervised Place Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **40,** 1437—1451 (2018).
4. Se, S., Lowe, D. G. & Little, J. Mobile Robot Localization and Mapping with Uncertainty using Scale-Invariant Visual Landmarks. *Int. J. Robotics Res.* **21,** 735—760 (2002).
5. Oliva, A. & Torralba, A. Building the gist of a scene: the role of global image features in recognition. *Progress in brain research* **155,** 23—36. `https://api.semanticscholar.org/CorpusID:2432623` (2006).
6. Sivic & Zisserman. *Video Google: a text retrieval approach to object matching in videos* в (IEEE, Nice, France, 2003), 1470—1477 vol.2. ISBN: 0-7695-1950-4.
7. Cummins, M. & Newman, P. M. Appearance-only SLAM at large scale with FAB-MAP 2.0. *Int. J. Robotics Res.* **30,** 1100—1123 (2011).
8. Lowe, D. G. Distinctive Image Features from Scale-Invariant Keypoints. *Int. J. Comput. Vis.* **60,** 91—110 (2004).
9. Bay, H., Tuytelaars, T. & Gool, L. V. *SURF: Speeded Up Robust Features* в *Computer Vision - ECCV 2006, 9th European Conference on Computer Vision, Graz, Austria, May 7-13, 2006, Proceedings, Part I* (ред. Leonardis, A., Bischof, H. & Pinz, A.) **3951** (Springer, 2006), 404—417.
10. Krizhevsky, A., Sutskever, I. & Hinton, G. E. ImageNet classification with deep convolutional neural networks. *Commun. ACM* **60,** 84—90 (2017).

11. Chen, Z., Lam, O., Jacobson, A. & Milford, M. Convolutional Neural Network-based Place Recognition. *CoRR* **abs/1411.1509.** arXiv: 1411.1509. http://arxiv.org/abs/1411.1509 (2014).

12. Sermanet, P. *та ін. OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks* в *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings* (ред. Bengio, Y. & LeCun, Y.) (2014). http://arxiv.org/abs/1312.6229.

13. He, K., Zhang, X., Ren, S. & Sun, J. Deep Residual Learning for Image Recognition. *CoRR* **abs/1512.03385.** arXiv: 1512.03385. http://arxiv.org/abs/1512.03385 (2015).

14. Simonyan, K. & Zisserman, A. *Very Deep Convolutional Networks for Large-Scale Image Recognition* в *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings* (ред. Bengio, Y. & LeCun, Y.) (2015). http://arxiv.org/abs/1409.1556.

15. Jégou, H., Douze, M., Schmid, C. & Pérez, P. *Aggregating local descriptors into a compact image representation* в (IEEE, San Francisco, CA, USA, 2010), 3304—3311. ISBN: 978-1-4244-6983-3.

16. Torii, A., Arandjelovic, R., Sivic, J., Okutomi, M. & Pajdla, T. 24/7 Place Recognition by View Synthesis. *IEEE Trans. Pattern Anal. Mach. Intell.* **40,** 257—271 (2018).

17. Ali-Bey, A., Chaib-Draa, B. & Giguére, P. *MixVPR: Feature Mixing for Visual Place Recognition* в (IEEE, Waikoloa, HI, USA, 2023), 2997—3006. ISBN: 978-1-6654-9347-5.

18. Sun, J., Shen, Z., Wang, Y., Bao, H. & Zhou, X. *LoFTR: Detector-Free Local Feature Matching with Transformers* в (IEEE, Nashville, TN, USA, 2021), 8918—8927. ISBN: 978-1-6654-4510-8.

19. Barbarani, G. *та ін.* Are Local Features All You Need for Cross-Domain Visual Place Recognition? *CoRR* **abs/2304.05887.** arXiv: 2304.05887 (2023).

20. Berton, G., Masone, C. & Caputo, B. *Rethinking Visual Geo-localization for Large-Scale Applications* в (IEEE, New Orleans, LA, USA, 2022), 4868—4878. ISBN: 978-1-6654-6947-0.

21. DeTone, D., Malisiewicz, T. & Rabinovich, A. *SuperPoint: Self-Supervised Interest Point Detection and Description* в (IEEE, Salt Lake City, UT, USA, 2018), 337—33712. ISBN: 978-1-5386-6101-7.

22. Tyszkiewicz, M. J., Fua, P. & Trulls, E. DISK: Learning local features with policy gradient. *CoRR* **abs/2006.13566.** arXiv: 2006.13566. https://arxiv.org/abs/2006.13566 (2020).

23. Edstedt, J., Bökman, G., Wadenbäck, M. & Felsberg, M. DeDoDe: Detect, Don't Describe - Describe, Don't Detect for Local Feature Matching. *CoRR* **abs/2308.08479.** arXiv: 2308.08479 (2023).

24. Edstedt, J., Sun, Q., Bökman, G., Wadenbäck, M. & Felsberg, M. RoMa: Revisiting Robust Losses for Dense Feature Matching. *CoRR* **abs/2305.15404.** arXiv: 2305.15404 (2023).

25. Schubert, S., Neubert, P., Garg, S., Milford, M. & Fischer, T. Visual Place Recognition: A Tutorial [Tutorial]. *IEEE Robotics Autom. Mag.* **31,** 139—153 (2024).

26. Schleiss, M., Rouatbi, F. & Cremers, D. VPAIR - Aerial Visual Place Recognition and Localization in Large-scale Outdoor Environments. *CoRR* **abs/2205.11567.** arXiv: 2205.11567 (2022).

27. Garg, S., Fischer, T. & Milford, M. Where is your place, Visual Place Recognition? *CoRR* **abs/2103.06443.** arXiv: 2103.06443. https://arxiv.org/abs/2103.06443 (2021).

28. Li, J., Eustice, R. M. & Johnson-Roberson, M. *High-level visual features for underwater place recognition* в (IEEE, Seattle, WA, 2015), 3652—3659. ISBN: 978-1-4799-6921-0.

29. Maddern, W., Pascoe, G., Linegar, C. & Newman, P. 1 year, 1000 km: The Oxford RobotCar dataset. *Int. J. Robotics Res.* **36,** 3—15 (2017).

30. Lu, F. *ma in.* Deep Homography Estimation for Visual Place Recognition. *CoRR* **abs/2402.16086.** arXiv: 2402.16086 (2024).

31. Naseer, T., Burgard, W. & Stachniss, C. Robust Visual Localization Across Seasons. *IEEE Transactions on Robotics* **34,** 289—302. ISSN: 1941-0468 (2 2018).

32. Google LLC. *Google Street View* https://www.google.com/maps/streetview. [Online; accessed 29-May-2025]. 2025.

33. Harris, C. G. & Stephens, M. *A Combined Corner and Edge Detector* в *Proceedings of the Alvey Vision Conference, AVC 1988, Manchester, UK, September, 1988* (ред. Taylor, C. J.) (Alvey Vision Club, 1988), 1—6.

34. Rosten, E., Porter, R. & Drummond, T. Faster and Better: A Machine Learning Approach to Corner Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **32,** 105—119. ISSN: 1939-3539 (1 січ. 2010).

35. Rublee, E., Rabaud, V., Konolige, K. & Bradski, G. *ORB: An efficient alternative to SIFT or SURF* в (IEEE, Barcelona, Spain, 2011), 2564—2571. ISBN: 978-1-4577-1100-8.

36. Alcantarilla, P. F., Nuevo, J. & Bartoli, A. *Fast Explicit Diffusion for Accelerated Features in Nonlinear Scale Spaces* в *British Machine Vision Conference, BMVC 2013, Bristol, UK, September 9-13, 2013* (ред. Burghardt, T., Damen, D., Mayol-Cuevas, W. W. & Mirmehdi, M.) (BMVA Press, 2013).

37. Leutenegger, S., Chli, M. & Siegwart, R. *BRISK: Binary Robust invariant scalable keypoints* в *IEEE International Conference on Computer Vision, ICCV 2011, Barcelona, Spain, November 6-13, 2011* (ред. Metaxas, D. N., Quan, L., Sanfeliu, A. & Gool, L. V.) (IEEE Computer Society, 2011), 2548—2555.

38. MacQueen, J. B. *Some Methods for Classification and Analysis of MultiVariate Observations* (ред. Cam, L. M. L. & Neyman, J.) (University of California Press, 1967).

39. Lindenberger, P., Sarlin, P.-E. & Pollefeys, M. *LightGlue: Local Feature Matching at Light Speed* в (IEEE, Paris, France, 2023), 17581—17592. ISBN: 979-8-3503-0719-1.

40. Sarlin, P.-E., DeTone, D., Malisiewicz, T. & Rabinovich, A. *SuperGlue: Learning Feature Matching With Graph Neural Networks* в (IEEE, Seattle, WA, USA, 2020), 4937—4946. ISBN: 978-1-7281-7169-2.

41. Fischler, M. A. & Bolles, R. C. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Commun. ACM* **24,** 381—395 (1981).

42. Edstedt, J., Bökman, G. & Zhao, Z. *DeDoDe v2: Analyzing and Improving the DeDoDe Keypoint Detector* в (IEEE, Seattle, WA, USA, 2024), 4245—4253. ISBN: 979-8-3503-6548-1.

43. Sünderhauf, N., Dayoub, F., Shirazi, S., Upcroft, B. & Milford, M. On the Performance of ConvNet Features for Place Recognition. *CoRR* **abs/1501.04158.** arXiv: 1501.04158. http://arxiv.org/abs/1501.04158 (2015).

44. Chen, Z., Liu, L., Sa, I., Ge, Z. & Chli, M. Learning Context Flexible Attention Model for Long-Term Visual Place Recognition. *IEEE Robotics and Automation Letters* **3,** 4015—4022. ISSN: 2377-3774 (4 жовт. 2018).