

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Західноукраїнський національний університет
Факультет комп'ютерних інформаційних технологій
Кафедра інформаційно-обчислювальних систем і управління

МЕЛЬНИК Віталій Андрійович

**Методи виділення ознак зображень для покращення точності
класифікації в системах комп'ютерного зору / Feature
Extraction Methods for Improving Classification Accuracy in
Computer Vision Systems**

спеціальність: 122 - Комп'ютерні науки
освітньо-професійна програма - Комп'ютерні науки

Кваліфікаційна робота

Виконав студент групи КНм-21
В.А. Мельник

Науковий керівник:
к.т.н., доцент Д.І. Загородня

Кваліфікаційну роботу
допущено до захисту:
«___» _____ 20___ р.

В.о. завідувача кафедри
_____ Н.В. Дзюбановська

ТЕРНОПІЛЬ - 2025

Факультет комп'ютерних інформаційних технологій
Кафедра інформаційно-обчислювальних систем і управління
Освітній ступінь «магістр»
спеціальність: 122 – Комп'ютерні науки
освітньо-професійна програма – Комп'ютерні науки

ЗАТВЕРДЖУЮ
В.о. завідувача кафедри
Н.М. Васильків
« ____ » _____ 20__ р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ
МЕЛЬНИК Віталій Андрійович
(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи

Методи виділення ознак зображень для покращення точності класифікації в системах комп'ютерного зору / Feature Extraction Methods for Improving Classification Accuracy in Computer Vision Systems

керівник роботи к.т.н., доцент Д.І. Загородня

затверджені наказом по університету від 20 грудня 2024 року № 938.

2. Строк подання студентом закінченої кваліфікаційної роботи 1 грудня 2025 р.

3. Вихідні дані до кваліфікаційної роботи: завдання на кваліфікаційну роботу студента, наукові статті, технічна література.

4. Основні питання, які потрібно розробити

- огляд предметної області комп'ютерного зору та задачі виявлення об'єктів;
- аналіз існуючих методів детекції та виділення ознак (SIFT, SURF, ORB, BRISK, AKAZE та ін.);
- виявлення ключових недоліків існуючих підходів у контексті задачі зі слабкотекстурованими об'єктами;
- постановка задачі дослідження щодо підвищення точності виділення ознак;
- формулювання вимог до нового методу та визначення обмежень;
- розробка нового методу виділення ознак, орієнтованого на об'єкти зі спрощеною геометрією;
- побудова математичної моделі запропонованого методу;

5. Перелік графічного матеріалу у роботі

- схема алгоритму методу EDGE;
- графіки порівняльного аналізу різних методів виділення ознак зображень.

РЕЗЮМЕ

Кваліфікаційна робота на тему «Методи виділення ознак зображень для покращення точності класифікації в системах комп'ютерного зору» на здобуття ступеня вищої освіти «Магістр» зі спеціальності 122 «Комп'ютерні науки» виконана обсягом 75 сторінок, містить 6 ілюстрацій, 1 додаток та 38 використаних джерел.

Метою кваліфікаційної роботи є підвищення точності процесу виявлення об'єктів на цифрових зображеннях шляхом удосконалення методів виділення локальних ознак та розробки нового контурно-орієнтованого дескриптора EDGE.

У роботі застосовано такі методи дослідження: системний аналіз, методи цифрової обробки зображень, математичні моделі локальних дескрипторів, статистичні методи оцінювання ефективності, інструменти експериментального комп'ютерного моделювання.

У процесі дослідження проведено комплексне поівняння класичних методів виділення ознак (SIFT, ORB, AKAZE) з розробленим методом EDGE. Запропоновано модифікований підхід до виявлення ключових точок на основі контурної структури, що дозволило знизити обчислювальні витрати та збільшити кількість коректних відповідностей для слабкотекстурних об'єктів, таких як логотипи.

Подальші дослідження можуть бути спрямовані на адаптацію методу EDGE до тривимірних сцен, удосконалення дескриптора з використанням машинного навчання, розширення тестових наборів, а також інтеграцію з багаторівневими гібридними структурами для підвищення точності при роботі з великими наборами даних.

Ключові слова: ЛОКАЛЬНІ ОЗНАКИ, ДЕСКРИПТОР, ВИДІЛЕННЯ ОЗНАК, ВИЯВЛЕННЯ ОБ'ЄКТІВ, КОНТУРНИЙ АНАЛІЗ, КОМП'ЮТЕРНИЙ ЗІР, EDGE.

ABSTRACT

Qualification work on the topic «Feature Extraction Methods for Improving Classification Accuracy in Computer Vision Systems» for Master's degree on speciality 122 «Computer Science» educational and professional program «Computer Science» is written on 75 pages and it contains 6 figures, 1 annexes and 38 sources.

The aim of the thesis is to improve the accuracy of object detection in digital images by enhancing local feature matching methods and developing a novel contour-oriented descriptor called EDGE.

The research methodology includes system analysis, digital image processing techniques, mathematical models of local descriptors, statistical evaluation methods, and computer-based experimental modeling.

The study presents a comprehensive comparison of classical feature matching algorithms (SIFT, ORB, AKAZE) with the proposed EDGE method. A modified approach to keypoint detection based on contour structure is introduced, enabling reduced computational costs and an increased number of correct matches, especially for low-texture objects such as logos.

Future research directions include adapting the EDGE method to 3D scenes, improving the descriptor using machine learning techniques, expanding the dataset of test images, and integrating the method into multilayer hybrid architectures to further enhance accuracy when working with large-scale data.

Keywords: LOCAL FEATURES, DESCRIPTOR, FEATURE EXTRACTION, OBJECT DETECTION, CONTOUR ANALYSIS, COMPUTER VISION, EDGE.

ЗМІСТ

Вступ.....	7
1 Стан та аналіз предметної області.....	10
1.1 Опис предметної області.....	10
1.2 Аналіз літературних джерел з досліджуваної тематики	16
1.3 Аналіз методів виділення ознак	18
1.4 Постановка задачі та обґрунтування напрямку дослідження	26
Висновки до розділу 1	30
2 Теоретичні дослідження та розробка методу виділення ознак.....	32
2.1. Загальна характеристика підходу до виділення ознак	32
2.2. Розробка нового методу виділення ознак EDGE.....	33
2.3. Математична модель та формалізація алгоритму	36
2.4. Алгоритм реалізації розробленого методу	39
Висновки до розділу 2	43
3 Практична реалізація та експериментальне дослідження методу виділення ознак	44
3.1. Опис методики проведення експериментальних досліджень.....	44
3.2. Програмна реалізація запропонованого методу	47
3.3. Аналіз отриманих результатів та їх практична цінність	51
Висновки до розділу 3	60
Висновки.....	63
Список використаних джерел.....	65
Додаток А Копії публікацій.....	68

ВСТУП

У сучасному світі цифрові зображення стали одним із ключових носіїв інформації, що активно використовуються в найрізноманітніших сферах людської діяльності — від промислової автоматизації та відеоспостереження до медичної діагностики, робототехніки, навігаційних систем та інтелектуальних інформаційних сервісів. Ефективність цих систем безпосередньо залежить від здатності алгоритмів коректно сприймати візуальну інформацію, виділяти з неї суттєві ознаки та зіставляти їх таким чином, щоб забезпечити надійну ідентифікацію або виявлення об'єктів у складних, непередбачуваних умовах. З кожним роком обсяг цифрових даних невпинно зростає, а реальні сценарії стають дедалі різноманітнішими, що створює постійну потребу у вдосконаленні методів комп'ютерного зору та розробці більш ефективних підходів до аналізу візуальних структур.

Серед великої кількості задач, які вирішує комп'ютерний зір, особливе місце займає задача виявлення та впізнавання об'єктів. Саме вона лежить в основі більшості прикладних систем: розпізнавання логотипів і брендів, аналізу товарів на полицях магазинів, ідентифікації небезпечних об'єктів, автоматичного контролю виробничих процесів, а також забезпечення роботи автономних систем, що повинні ухвалювати рішення на основі візуальної інформації. Проте успішність виконання цієї задачі визначається не лише якістю детекції ключових точок чи обчисленням дескрипторів, але й точністю зіставлення ознак між зображеннями — етапом, який є критично важливим і водночас найбільш вразливим до впливу зовнішніх факторів. Саме помилки на цьому етапі призводять до хибних відповідностей, втрати корисної інформації та значного зниження загальної точності алгоритму.

Попри значний розвиток класичних методів, таких як SIFT, SURF, ORB та AKAZE, сучасні умови роботи викликають чимало нових труднощів. У багатьох реальних сценаріях об'єкти можуть бути частково закритими, зміненими, мати низьку роздільну здатність або бути зафіксованими за важких умов освітлення. У таких випадках традиційні дескриптори часто демонструють недостатню стійкість,

що вимагає розробки нових підходів, здатних забезпечити коректність виділення та зіставлення навіть за незначних спотворень вхідних даних. Крім того, зростає потреба у методах, що зберігають баланс між точністю та обчислювальною ефективністю, оскільки сучасні інформаційні системи все частіше повинні працювати в режимі реального часу.

У цьому контексті виникає необхідність у побудові методів виділення та зіставлення ознак, що поєднують у собі високу інформативність дескрипторів, робастність до змін масштабу, поворотів та освітлення, а також стійкість до спотворень та шуму. Одним із перспективних напрямів є використання градієнтних характеристик та узагальнених орієнтаційних структур, які здатні зберігати свою стабільність у широкому спектрі умов. Такий підхід дозволяє формувати більш узагальнене подання про локальні геометричні властивості зображення, що, у свою чергу, створює підґрунтя для точнішого та надійнішого виділення та зіставлення ключових точок.

У межах цієї роботи розглядається можливість підвищення точності виявлення об'єктів за рахунок удосконалення процедури виявлення ознак та запропоновано новий метод, спрямований на посилення надійності дескрипторів, оптимізацію структури їхнього порівняння та зниження кількості хибних відповідностей. Особлива увага приділяється побудові дескриптора на основі узагальненого аналізу градієнтних піків, який дозволяє враховувати найстійкіші локальні структури, що зберігаються навіть за значних змін геометрії або освітлення зображення.

Метою дослідження є розроблення, математичне обґрунтування та експериментальна перевірка нового методу виявлення та зіставлення ознак зображень, який забезпечує підвищену точність виявлення об'єктів у складних умовах, а також демонструє переваги над класичними підходами завдяки більш стійкому використанню локальних градієнтних характеристик. Досягнення поставленої мети передбачає створення алгоритмічної основи методу, його формалізацію, розроблення дескрипторної моделі, визначення критеріїв подібності та оцінку ефективності шляхом порівняння з існуючими алгоритмами.

Методи дослідження, які використані в роботі, охоплюють теоретичне моделювання структур ознак, математичне формалізування алгоритмів виявлення та зіставлення, аналіз поведінки дескрипторів у різних умовах, а також експериментальне оцінювання їхніх властивостей на реальних і синтетичних даних. Додатково застосовуються чисельні методи та комп'ютерне моделювання, що дозволяє відновити детальну картину впливу різноманітних факторів на результат роботи методу, а також виявити закономірності, які неможливо встановити аналітичним шляхом.

Таким чином, дослідження спрямоване на з'ясування можливостей подальшого вдосконалення методів виявлення та зіставлення ознак зображень та на розробку підходу, здатного забезпечити стабільне та точне виявлення об'єктів у сучасних інформаційних системах. Робота покликана не лише продемонструвати практичну корисність нового методу, але й обґрунтувати його теоретичні переваги, що створює основу для подальшого розвитку технологій комп'ютерного зору.

Наукова новизна одержаних результатів полягає в тому, що запропоновано новий метод виділення ознак зображень EDGE, який базується на використанні градієнтних екстремумів та компактного гістограмного опису орієнтацій. Запропонований підхід забезпечує підвищення точності зіставлення об'єктів зі спрощеною геометрією за рахунок зменшення кількості хибних відповідностей та оптимізації процесу вибору ключових точок.

Апробація результатів дослідження здійснювалася шляхом представлення основних теоретичних положень і практичних результатів на міжнародній науково-практичній конференції «Інноваційний розвиток економіки, управління, фінансів і права: теорія, практика, перспективи», що відбулася 1 грудня 2025 року у місті Житомир, а також на International Scientific and Practical Conference «Science, Education, Economics, and Society of the 21st Century: Current State, Issues, and Prospects», яка проходила 2 грудня 2025 року у місті Остін. Копії публікацій представлено у додатку А.

Кваліфікаційна робота складається із вступу, трьох розділів, висновків, списку використаних джерел та додатків.

1 СТАН ТА АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Опис предметної області

Предметна область дослідження — методи виділення ознак зображень для покращення точності класифікації в системах комп'ютерного зору — охоплює широкий спектр теоретичних понять, алгоритмічних підходів та практичних задач, що лежать на перетині комп'ютерного зору, обробки зображень та інженерії програмного забезпечення для інтелектуальних систем. Нижче наведено розгорнутий опис цієї предметної області, її основних компонентів, викликів та контексту застосування — із детальним поясненням термінів, етапів обробки та зв'язків між складовими.

У найзагальнішому вигляді задача полягає в тому, щоб за допомогою аналізу цифрових зображень надійно знаходити та ідентифікувати об'єкти у сцені. Для цього в практиці часто застосовують підхід, оснований на виділенні локальних ознак (features) у зображеннях та їх зіставленні (matching) між «еталонним» зображенням об'єкта і зображенням сцени. Зіставлення ознак дозволяє встановити відповідності між локальними точками на двох зображеннях, що дає змогу локалізувати об'єкт, оцінити його положення, масштаб, орієнтацію та іноді відновити геометричну трансформацію (наприклад, гомографію) [29].

Це завдання має велике значення в багатьох практичних областях: від задач розпізнавання та класифікації в системах відеоспостереження до позиціонування у робототехніці, від доповненої реальності до контролю якості на виробництві. У кожному з цих випадків від надійності зіставлення ознак залежить остаточна точність виявлення об'єкта.

Предметну область доцільно розділити на кілька логічно взаємопов'язаних етапів, кожен з яких вирішує окремі завдання та має власні алгоритмічні підходи, параметри налаштування та критерії ефективності.

Першим етапом є попередня обробка зображень. Її метою є приведення вхідних даних до уніфікованого вигляду, що підвищує стабільність подальших етапів аналізу [17]. Зазвичай попередня обробка включає нормалізацію

освітленості, перетворення зображень у відтінки сірого, фільтрацію шуму за допомогою згладжувальних фільтрів (наприклад, Gaussian blur), а також корекцію гамми для вирівнювання контрасту. Такі операції дозволяють зменшити вплив зовнішніх факторів, як-от нерівномірне освітлення, цифровий шум чи низька якість камери. Крім того, на цьому етапі виконується масштабування зображень до стандартизованого розміру, що знижує обчислювальне навантаження та забезпечує співмірність між зображеннями, які порівнюються.

Другим важливим кроком є детекція ключових точок (keypoint detection). Метою цього етапу є виявлення локальних структур зображення, які мають найбільшу інформативність і можуть бути використані для подальшого опису об'єкта. Ключові точки зазвичай відповідають кутам, різким змінам яскравості або складним текстурним ділянкам. Серед поширених алгоритмів детекції можна назвати Harris corner detector і FAST. Якісна детекція характеризується високою повторюваністю (repeatability) — здатністю виділяти ті самі точки навіть за зміни умов зйомки, а також репрезентативністю — тобто стабільністю ідентифікації значущих елементів сцени [3].

Наступним етапом виступає обчислення дескрипторів (feature description). Навколо кожної знайденої ключової точки формується числовий вектор — дескриптор, який описує локальний вміст зображення у форматі, стійкому до різних трансформацій, таких як повороти, масштабування або часткові зміни освітлення. Класичними прикладами дескрипторів є SIFT (Scale-Invariant Feature Transform), SURF, BRIEF, ORB (поєднання FAST і BRIEF з використанням Hamming-відстані), а також AKAZE. Сучасні тенденції орієнтуються на застосування навчальних дескрипторів, побудованих на базі нейронних мереж — таких як SuperPoint, HardNet або LIFT, що забезпечують підвищену точність у складних умовах. Дескриптори можуть мати формат із плаваючою точкою (float32), що підвищує точність, або бінарний формат, який забезпечує швидше порівняння під час пошуку відповідностей [19].

Четвертим етапом є зіставлення дескрипторів (feature matching). На цьому кроці здійснюється пошук пар подібних дескрипторів між двома зображеннями.

Для кожного дескриптора з еталонного зображення визначають найближчий за метрикою схожості (L2-норма для дескрипторів з плаваючою точкою або Hamming-відстань для бінарних). Серед популярних інструментів можна назвати Brute-Force Matcher (BFMatcher) та FLANN (Fast Library for Approximate Nearest Neighbors). Щоб підвищити надійність результату, застосовують методи фільтрації, наприклад, тест відношення Lowe's ratio test, який порівнює відстані до двох найближчих збігів і відкидає неоднозначні або потенційно помилкові відповідності [16].

Після цього проводиться геометрична верифікація (geometric verification). Вона необхідна, оскільки навіть після ретельного зіставлення залишаються так звані «викиди» (outliers) — хибні відповідності між точками. Для їх усунення використовуються методи оцінки геометричної узгодженості, серед яких найпоширенішим є алгоритм RANSAC (Random Sample Consensus). Він дозволяє виявити і відкинути аномальні пари точок, побудувавши гомографію або афінне перетворення, яке описує взаємне розташування двох зображень [18]. Це дає можливість не лише покращити точність результатів, але й оцінити параметри положення об'єкта у сцені — масштаб, кут повороту та зміщення.

Завершальним етапом виступає постобробка та прийняття рішення. На цьому кроці система аналізує результати попередніх етапів і визначає, чи присутній цільовий об'єкт у сцені. Для цього враховується кількість коректних відповідностей («good matches»), якість знайденої гомографії, а також низка додаткових метрик — площа збігу точок, співвідношення кількості правильних відповідностей до загальної кількості, тощо [30].

Важливо також визначити основні показники ефективності методів, які використовуються у досліджуваній предметній області. До таких метрик належать [32]:

- repeatability (повторюваність) — показник того, як стабільно детектор виявляє однакові ключові точки при різних умовах;
- matching accuracy (точність зіставлення) або precision — частка правильних збігів серед усіх знайдених;

- recall — частка істинних відповідностей, які були успішно знайдені;
- F1-score — гармонічне середнє між precision і recall, що дозволяє об'єктивно оцінювати баланс між ними;
- час обробки (latency, throughput) — критичний параметр для систем реального часу;
- складність використання пам'яті — важлива характеристика для мобільних і вбудованих систем;
- стійкість до змін зовнішніх умов — таких як варіації освітлення, наявність шумів, часткове перекриття об'єкта (оклюзія), зміна масштабу або перспективи.

Таким чином, кожен із наведених етапів формує окрему ланку єдиного процесу аналізу зображень. Від якості реалізації кожного кроку залежить точність, швидкість і надійність усієї системи розпізнавання, що робить дослідження цієї предметної області надзвичайно актуальним та перспективним.

Якість зіставлення ознак у системах комп'ютерного зору визначається низкою факторів, які прямо впливають на точність, швидкість та надійність розпізнавання об'єктів. Серед ключових викликів можна виділити кілька груп проблем.

По-перше, зміни освітлення та контрасту суттєво впливають на стабільність локальних патернів і дескрипторів. Навіть незначні варіації освітленості можуть призводити до того, що алгоритм сприйматиме однакові області зображення як різні, знижуючи якість зіставлення [13].

По-друге, наявність шуму та артефактів компресії (наприклад, JPEG) ускладнює виявлення ключових точок і знижує точність відповідностей. Ця проблема особливо помітна при роботі з даними, отриманими з камер низької якості або з великим ступенем стиснення [27].

Третій фактор — масштабні зміни та повороти об'єктів, які потребують використання інваріантних дескрипторів або багатомасштабних методів. Саме тому такі алгоритми, як SIFT чи SURF, залишаються класичними стандартами у задачах комп'ютерного зору.

Ще одним викликом є часткова оклюзія (закриття об'єкта іншими предметами), що призводить до зменшення кількості доступних ознак. Крім того, повторювані структури у сцені, наприклад фасади будівель або текстуровані поверхні, можуть створювати неоднозначності при зіставленні.

Різні точки зору (*perspective changes*) також є складною проблемою, адже сильні перспективні спотворення суттєво змінюють геометрію ознак. Це особливо критично для непланарних об'єктів, де звичайна гомографія не може точно відновити просторову структуру сцени.

Нарешті, в умовах обмежених апаратних ресурсів (наприклад, у мобільних або вбудованих системах) виникає потреба враховувати обчислювальні обмеження, тобто баланс між точністю, швидкістю та споживанням ресурсів.

Розвиток методів виділення та зіставлення ознак пройшов кілька етапів — від класичних інженерних рішень до сучасних навчальних моделей.

Класичні (*hand-crafted*) алгоритми — такі як SIFT, SURF, ORB, BRIEF, BRISK та AKAZE — залишаються фундаментом багатьох систем комп'ютерного зору. Вони мають зрозумілу математичну основу, високу інтерпретованість і стабільно показують хороші результати у різних умовах [26].

Швидкі апаратно-оптимізовані методи, наприклад бінарні дескриптори ORB або BRIEF, особливо актуальні для застосувань, де критично важливий час обробки — таких як мобільна навігація або системи реального часу.

Сучасна тенденція полягає у використанні нейронних мереж для побудови навчальних детекторів та дескрипторів (SuperPoint, D2-Net, R2D2). Ці методи демонструють підвищену повторюваність і точність, особливо в складних умовах освітлення або перспективних спотворень. Водночас вони потребують великих обчислювальних потужностей і навчання на великих наборах даних.

В останні роки з'явилися гібридні підходи, що поєднують класичні детектори з навчальними дескрипторами або навпаки. Такий підхід дозволяє зберегти баланс між швидкістю та точністю, а також спрощує інтеграцію у вже існуючі системи.

Загалом, сучасні дослідження показують, що нейромережеві фічери активно розвиваються, проте класичні методи не втрачають своєї актуальності, особливо у проєктах із жорсткими вимогами до стабільності та обмеженими ресурсами.

Методи зіставлення ознак безпосередньо пов'язані з ширшим класом задач комп'ютерного зору — виявленням об'єктів.

У класичних підходах *feature matching* використовується для точного розпізнавання конкретних відомих об'єктів (логотипів, етикеток, штрих-кодів тощо). Такі методи забезпечують високу локалізацію і дозволяють відновити геометрію об'єкта на сцені, що є особливо корисним для доповненої реальності або промислової інспекції [21].

Сучасні детектори на основі згорткових нейронних мереж (CNN), такі як YOLO або Faster R-CNN, ефективно працюють із великими наборами даних і можуть розпізнавати об'єкти різних типів. Проте вони іноді поступаються класичним методам у точності локалізації дрібних або специфічних об'єкт. Тому часто застосовують комбіновані підходи — CNN для грубої детекції, а *feature matching* для точного підтвердження і позиціонування [33].

Датасети, сценарії тестування та практичні аспекти. Для перевірки ефективності методів зазвичай використовуються комбінації реальних і синтетичних зображень. Реальні зображення дозволяють оцінити поведінку алгоритмів у природних умовах, тоді як синтетичні трансформації (масштабування, поворот, шум, оклюзія) — забезпечують контрольоване тестування [5].

Типовий набір даних включає:

- еталонні зображення об'єктів (логотипи, етикетки, деталі);
- сцени, у яких ці об'єкти розташовані у реальному середовищі;
- серії трансформованих зображень, створених шляхом поворотів, розмиття або зміни масштабу.

Оцінювання результатів проводиться за допомогою стандартних метрик — *precision*, *recall*, *F1-score*, а також часу обробки та стійкості до шуму [8].

При впровадженні систем виявлення ознак у практику враховують такі чинники: апаратні обмеження (CPU, GPU, пам'ять), вимоги до часу реакції

(реальний час або пакетна обробка), масштабування баз даних, сумісність із форматами API, а також питання ліцензування, оскільки деякі алгоритми (наприклад, SIFT і SURF) тривалий час мали патентні обмеження [31].

Методи виявлення та зіставлення ознак мають широке практичне застосування:

- у доповненій реальності (AR) — для точного позиціонування віртуальних об'єктів у просторі;
- у робототехніці — для візуальної навігації та локалізації;
- у промислових задачах — для автоматичної інспекції якості та пошуку дефектів;
- у маркетингових системах — для розпізнавання логотипів і брендів;
- у медицині — для зіставлення фрагментів медичних зображень з метою діагностики або відстеження змін.

Покращення методів виявлення ознак є одним із ключових напрямів розвитку сучасного комп'ютерного зору. Воно безпосередньо впливає на точність виявлення об'єктів, стійкість до спотворень і здатність систем працювати в реальному часі. Попри активний розвиток глибинного навчання, класичні алгоритми та їх комбінації із навчальними підходами залишаються актуальними завдяки своїй інтерпретованості, ефективності та можливості застосування в умовах обмежених ресурсів.

1.2 Аналіз літературних джерел з досліджуваної тематики

Аналіз наукових джерел є ключовим етапом у процесі дослідження, оскільки дозволяє систематизувати наявні знання, визначити актуальні напрями розвитку обраної тематики та виявити наукові прогалини, які потребують подальшого опрацювання. Важливо розглянути як класичні праці з комп'ютерного зору, так і сучасні публікації, що присвячені машинному навчанню та глибинним нейронним мережам [10].

У багатьох друкованих та електронних виданнях значна увага приділяється питанню опису ознак зображень як фундаментальному елементу процесу розпізнавання. У ранніх дослідженнях з комп'ютерного бачення використовувались прості евристичні методи, що ґрунтувались на геометричних характеристиках зображення. У роботах дослідників класичної школи традиційно розглядались такі поняття, як контурні лінії, градієнти яскравості та порогове бінаризування, що дозволяло виділяти найпростіші ознаки об'єктів. Подібні підходи мали обмежену точність, однак саме вони стали основою для формування більш розвинених моделей аналізу зображень.

Подальші дослідження, які умовно можна віднести до другого етапу розвитку предметної області, стосувались формалізації ознак на основі математичних описів. У наукових публікаціях почали з'являтися алгоритми порівняння ознак, побудовані на основі векторного представлення фрагментів зображень. До таких відносять методи SIFT (Scale-Invariant Feature Transform) та SURF (Speeded-Up Robust Features), які згадуються у значній кількості зарубіжних джерел як базові у розпізнаванні об'єктів з урахуванням масштабування та повороту сцени. Автори цих досліджень стверджували, що правильно підібрані локальні ознаки дозволяють значно підвищити точність ідентифікації, незалежно від зовнішніх умов зйомки [34].

У третьому етапі, який охоплює сучасні публікації останніх років, ключову роль почали відігравати методи глибинного навчання, що дозволяють автоматично формувати ознаки зображень без необхідності вручну визначати критерії їх порівняння. У численних статтях, присвячених нейронним мережам, підкреслюється перевага згорткових архітектур (CNN) у порівнянні з класичними алгоритмами детектування ознак. Зокрема, в літературі акцентується увага на тому, що CNN здатні не лише розпізнавати об'єкти, але й адаптивно навчатися новим патернам без втручання розробника. Цей підхід дозволив перейти від ручної інженерії ознак до автоматизованого вилучення високорівневих дескрипторів, що значно збільшило точність детекції [6].

Серед сучасних джерел також відзначаються роботи, присвячені комбінуванню класичних методів порівняння ознак із глибинними нейронними мережами. У таких дослідженнях автори підкреслюють, що поєднання традиційних дескрипторів з нейромережевими ознаками дозволяє досягти синергетичного ефекту, коли швидкість обробки зображення зберігається на високому рівні, а точність виявлення об'єктів додатково підвищується. Особливо часто згадуються підходи побудови гібридних моделей, де на першому етапі використовуються класичні методи фільтрації шуму та контурного аналізу, а на наступному — нейромережева класифікація [20].

У ході аналізу літературних джерел можна виділити кілька ключових напрямів, на яких зосереджена увага сучасних дослідників:

- методи виявлення локальних ознак зображень з урахуванням стійкості до трансформацій;
- порівняння дескрипторів та алгоритми зіставлення ознак;
- гібридні підходи комбінування традиційних і нейромережових методів;
- оптимізація процесу виділення ознак для підвищення продуктивності систем розпізнавання;
- адаптивні механізми навчання моделей для аналізу нових наборів зображень.

Проведений аналіз показує, що попри значну кількість наявних рішень, проблема виділення і зіставлення ознак зображень залишається актуальною, особливо в умовах потреби у високій точності розпізнавання та мінімізації ймовірності хибного спрацювання. Це створює наукове підґрунтя для подальшого дослідження теми та розробки власних підходів, що поєднують переваги існуючих методів.

1.3 Аналіз методів виділення ознак

У цьому підрозділі розглядаються математичні та алгоритмічні основи найпоширеніших підходів до виділення ознак, опису та зіставлення локальних

ознак зображень — зокрема SIFT, SURF, ORB та AKAZE. Мета — не лише описати кроки алгоритмів, але й розгорнуто показати, на яких математичних операціях вони ґрунтуються, які властивості забезпечують інваріантності, робастність, та як ці властивості впливають на результати у задачах виділення та зіставлення.

Більшість підходів до виділення локальних ознак має однакову загальну архітектуру: детекція ключових точок, оцінка локальної орієнтації, побудова дескриптора (feature description), зіставлення дескрипторів (matching), геометрична верифікація (validation, наприклад RANSAC) [2].

Такий конвеєр (pipeline) у формалізованому вигляді можна записати як послідовність операцій над зображенням $I(x,y)$, де на кожному кроці застосовується певна функція перетворення та фільтрації, що виводить множини ключових точок та векторних дескрипторів:

$$I \xrightarrow{\text{detection}} \{(x_i, y_i, \sigma_i)\} \xrightarrow{\text{orientation}} \{(x_i, y_i, \sigma_i, \theta_i)\} \xrightarrow{\text{description}} \{\mathbf{d}_i\}$$

Гradient інтенсивності $I(x,y)$ — фундаментальна локальна характеристика:

$$I_x = \frac{\partial I}{\partial x}, \quad I_y = \frac{\partial I}{\partial y},$$

$$\text{magnitude } m = \sqrt{I_x^2 + I_y^2}, \quad \text{orientation } \theta = \arctan 2(I_y, I_x)$$

Щоб зменшити вплив шуму та реалізувати багатомасштабний аналіз, використовують гаусову згортку:

$$L(x, y; \sigma) = G(x, y; \sigma) * I(x, y), \quad G(x, y; \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Різні алгоритми використовують похідні по гаусово згладженому зображенню: $L_x, L_y, L_{xx}, L_{xy}, L_{yy}$, що дозволяє виявляти локальні структури та оцінювати другі похідні (Hessian, Laplacian).

1.3.1 Принцип масштабної інваріантності. DoG і Laplacian

Ідея масштабної інваріантності — знайти ті особливості зображення, які будуть виявлені незалежно від масштабу об'єкта. Технічно це робиться формуванням scale-space:

$$L(x, y; \sigma) = G(\cdot; \sigma) * I(x, y), \quad \sigma \in [\sigma_{min}, \sigma_{max}]$$

Difference of Gaussians (DoG) — чисельне наближення Laplacian of Gaussian (LoG):

$$\text{DoG}(x, y; \sigma, k) = L(x, y; k\sigma) - L(x, y; \sigma)$$

Локальні екстремуми у тривимірному просторі (x, y, σ) DoG визначають кандидати у ключові точки (SIFT). DoG є обчислювально простішим, ніж обчислення точного LoG, але зберігає потрібну властивість знаходження масштаб-інваріантних точок [12].

Laplacian (оператор Лапласа) використовується у класичних підходах для виявлення блоб-подібних структур; SURF наближує схожі вторинні похідні через Hessian-фільтри.

Harris corner response — детекція кутів і її математична сутність: Harris detector виявляє кути шляхом аналізу матриці другого моменту для локальної області:

$$M = \begin{pmatrix} \sum w(x, y) I_x^2 & \sum w(x, y) I_x I_y \\ \sum w(x, y) I_x I_y & \sum w(x, y) I_y^2 \end{pmatrix}$$

Відгук (corner response) обчислюється як:

$$R = \det(M) - k (\text{trace}(M))^2$$

де k — емпіричний коефіцієнт (типово 0.04 – 0.06). Високі значення R відповідають кутовим точкам (областям з помітною змінністю у двох напрямках).

SIFT (Scale-Invariant Feature Transform) — класичний метод, що складається з кількох етапів:

- побудова scale-space та DoG,
- пошук локальних екстремумів у DoG (кандидати ключових точок),
- уточнення положення (інтерполяція підпиксельно),
- призначення орієнтації (орієнтувальна гістограма градієнтів у вікні навколо точки),
- формування дескриптора: 16 підвікон по 4×4 , в кожному гістограма орієнтацій (8-bin), отримуємо 128-елементний вектор; нормалізація вектора (L2-normalization) для інваріантності до освітлення.

Формально, дескриптор $d \in \mathbb{R}^{128}$ є нормалізованим вектором градієнтів, агрегованих у просторово-орієнтаційні під-ячейки.

До переваг SIFT належать інваріантність до масштабу та повороту, добра стійкість до освітлення (через нормалізацію дескриптора), висока discriminability (через плаваючі дескриптори).

До недоліків методу відносять відносно велика обчислювальна та пам'ятна вартість.

SURF (Speeded-Up Robust Features) запроектовано як пришвидшена версія ідей SIFT, детекція базується на детермінанті матриці Hessian:

$$H = \begin{pmatrix} L_{xx} & L_{xy} \\ L_{xy} & L_{yy} \end{pmatrix}, \quad \det(H) = L_{xx}L_{yy} - L_{xy}^2$$

SURF замінює точні похідні на одиничні box-фільтри (box filters), які дуже швидко обчислюються через integral images (сума пікселів у прямокутнику обчислюється за константний час) [14].

Орієнтація обчислюється як сумарні вектори градієнтів у вікні, а дескриптор будується з агрегованих сумарних значень притоку градієнтів у під регіонах (короткий плаваючий дескриптор).

SURF використовує знак лапласа ($\text{trace}(H)$ або знак $L_{\{xx\}} + L_{\{yy\}}$) як бінарну ознаку для попередньої класифікації ознак (fast sign test) [11].

До переваг SURF відносять: значно швидше за SIFT через integral images і box-фільтри, добра інваріантність до масштабу і повороту (але трохи гірша ніж у SIFT у деяких випадках). Проте є обмеження, оскільки SURF все ще оперує плаваючими значеннями, але знижує обчислювальні витрати завдяки апроксимаціям.

ORB (Oriented FAST and Rotated BRIEF) — прагматичний алгоритм, оптимізований для швидкості та застосувань вбудованих систем:

- детектор FAST (Features from Accelerated Segment Test) — виявляє кути на основі інтенсивності пікселів у певному колі навколо кандидата. Обчислюється через момент-центроїд локального вікна; напрям орієнтації θ отримують з центрального моменту — $\theta = \arctan2(m_{01}, m_{10})$;

- дескриптор BRIEF — бінарний дескриптор, побудований як набір порівнянь інтенсивності між парами точок у локальному вікні; ORB застосовує обертання BRIEF відповідно до θ (rBRIEF), що робить його ротаційно-інваріантним.

Бінарні дескриптори зберігаються як послідовність бітів (наприклад, 256 біт) і порівнюються за допомогою Hamming distance, що дуже швидко реалізується апаратно:

$$\text{Hamming}(\mathbf{b}_1, \mathbf{b}_2) = \sum_{i=1}^n [b_{1i} \oplus b_{2i}]$$

Переваги ORB: дуже висока швидкість обчислення і matching, мала пам'ять (бінарні дескриптори), хороша практична якість для багатьох застосувань.

Недоліки: менша точність у складних умовах порівняно з SIFT, обмежена стійкість до сильних змін масштабу.

AKAZE — нелінійний scale-space і бінарний дескриптор. KAZE / AKAZE (Accelerated KAZE) стоять на ідеї нелінійного diffusion-based scale-space, замість лінійної гаусової фільтрації застосовується нелінійна дифузія за Перона–Маліка [15]:

$$\frac{\partial L}{\partial t} = \nabla \cdot (c(\|\nabla L\|)\nabla L)$$

Переваги: краща збережуваність контурів і деталей у scale-space порівняно з лінійною гаусовою згорткою;

AKAZE — швидша апроксимація KAZE з реалізацією M-LDB (Modified-Local Difference Binary) дескриптора — бінарного типу, схожого по ідеї на BRIEF, але з використанням локальних градієнтів у scale-space.

AKAZE дає добрий компроміс між точністю і швидкістю, особливо на текстурних сценах.

Порівняння плаваючих і бінарних дескрипторів:

- розмір представлення: SIFT — 128 floating-point чисел (звично 128×4 байта ≈ 512 байт), ORB/AKAZE — 256 біт ≈ 32 байти;
- метрика порівняння: SIFT — Euclidean (L2), ORB/AKAZE — Hamming;
- швидкість: бінарні дескриптори дають значний виграш у швидкості порівняння та менш вимогливі до пам'яті;
- discriminability: плаваючі дескриптори зазвичай мають вищу здатність розрізняти схожі патерни, отже менше хибних збігів у складних умовах.

Для двох випадкових бінарних векторів довжини n очікувана Hamming-відстань $\approx n/2$ з дисперсією $n/4$; отже, щоб відрізнити істинні збіги від випадкових, потрібно, щоб дескриптор мав достатню довжину n та низьку імовірність колізії для схожих патернів.

Для плаваючих дескрипторів нормалізація (L2) та застосування порогів на відстані дають кращу роздільну здатність у просторі дескрипторів.

Шум (додавання випадкових флуктуацій інтенсивності) значно впливає на похідні: градієнти I_x , I_y мають додаткову стохастичну компоненту. Застосування гаусового згладжування (σ залежить від масштабу) зменшує вплив шуму. Однак надмірне згладжування затирає деталі \rightarrow компроміс.

В ідеалі детектор сигналізує про ключову точку на відповідному σ , тому дескриптор будується у координатах, нормалізованих за цим σ . Практично, якщо масштабне перетворення перевищує діапазон розглянутого *scale-space*, повторюваність падає.

Перспективні спотворення (affine / projective): класичні інваріантності (scale, rotation) не гарантують інваріантності до сильних перспективних деформацій. Для цього розробляються affine-invariant методики (Hessian-affine, Harris-affine) або застосовуються відповідні моделі геометричної верифікації.

Умови освітлення: використання градієнтів та нормалізацій дескрипторів підвищує стійкість до глобальних змін освітлення, проте локальні нерівномірні освітлювальні ефекти (тіні, блики) можуть змінити локальні структури та викликати зниження якості match'ів.

Процес виявлення та зіставлення ознак складається з послідовності етапів, кожен з яких має своє функціональне призначення. Спочатку формується простір масштабів $L(x,y;\sigma)$, у якому здійснюється пошук локальних екстремумів за допомогою методів DoG, Hessian або FAST. Після цього кожен знайдений кандидат уточнюється, відсіюються нестабільні точки, а для стабільних визначається орієнтація θ — найчастіше на основі гістограми градієнтів або моментів інтенсивності. Навколо кожної ключової точки формується дескриптор d — це або 128-елементний вектор із дійсними числами (як у SIFT), або компактний бінарний код (як у ORB). У результаті отримуємо набір параметрів (x,y,σ,θ,d) , що характеризують кожну локальну ознаку.

Під час етапу matching (зіставлення) для кожного дескриптора еталонного зображення здійснюється пошук найближчих сусідів у дескрипторах сцени

(зазвичай за допомогою методів Brute Force або FLANN). Далі застосовується Lowe's ratio test, який відсіює неоднозначні збіги за співвідношенням відстаней між першим і другим найближчим сусідом. Отримані потенційні збіги піддаються геометричній перевірці з використанням алгоритму RANSAC, що дозволяє побудувати модель гомографії та виокремити лише ті відповідності, які узгоджуються з нею. Кінцеве рішення про наявність об'єкта на зображенні приймається на основі кількості коректних збігів (inliers) і їхнього співвідношення до загальної кількості match'ів.

З точки зору обчислювальної складності, етапи детекції та опису ознак зазвичай мають майже лінійну залежність від кількості пікселів або знайдених ключових точок, хоча створення плаваючих дескрипторів є обчислювально дорожчим через велику кількість операцій множення та згорток [28]. Зіставлення ознак методом Brute Force має складність $O(N \cdot M)$, де N і M — кількість дескрипторів на двох зображеннях, тоді як використання FLANN або ANN дозволяє досягти сублінійної поведінки ціною невеликої апроксимації. Варто зазначити, що бінарні дескриптори суттєво прискорюють процес matching завдяки апаратній підтримці операцій XOR і POPCOUNT.

Теоретичний аналіз дозволяє очікувати, що алгоритм SIFT продемонструє найвищу точність і стабільність відповідей у складних умовах освітлення, масштабування та повороту, хоча й поступатиметься за швидкістю. Методи ORB та AKAZE натомість забезпечують кращу продуктивність, зберігаючи прийнятний рівень дискримінативності, що робить їх придатними для реальних, обмежених за ресурсами систем. Алгоритм SURF займає проміжну позицію, забезпечуючи баланс між точністю та швидкістю, але через свої апроксимації може виявляти меншу стійкість у деяких випадках.

Практичні експерименти мають підтвердити ці теоретичні очікування шляхом аналізу повторюваності ключових точок при геометричних трансформаціях, кількості коректних збігів до і після тесту Lowe, числа inliers після застосування RANSAC, а також оцінки часу обробки та розміру дескрипторів для різних типів ознак.

1.4 Постановка задачі та обґрунтування напрямку дослідження

У сучасних системах комп'ютерного зору процес виявлення об'єктів ґрунтується на поєднанні декількох ключових етапів, серед яких одним із найважливіших є виділення ознак зображень. Саме цей етап визначає, наскільки коректно буде встановлено відповідність між локальними особливостями різних зображень та чи зможе алгоритм впевнено розпізнати об'єкт в умовах зміни освітлення, ракурсу, масштабу, часткової оклюзії або наявності шуму. Незважаючи на значний прогрес у цій галузі, багато існуючих методів стикаються з проблемами нестійкості, недостатньої точності або надмірної обчислювальної складності, що обмежує їх використання у практичних застосуваннях, які вимагають високої продуктивності та надійності.

Мета дослідження – забезпечити зростання точності детекції об'єктів у візуальних сценах шляхом аналізу й порівняння методів виділення ключових точок, опису дескрипторів та методів їх зіставлення, а також шляхом вибору або комбінування відповідних алгоритмів і налаштувань [36].

Основні завдання дослідження:

1. Огляд і формалізація базових критеріїв якості для задачі виділення ознак.

Для забезпечення точності вимірювань і порівняння методів необхідно визначити єдиний набір метрик. У різних роботах використовуються різні підходи до оцінки (precision/recall, repeatability, час виконання тощо), тому першим кроком є вибір та формалізація системи оцінювання. Необхідно визначити:

- набір метрик — кількість «good matches», precision/recall/F1, repeatability для детекторів, час обробки (latency), обсяг пам'яті для дескрипторів.
- алгоритм обчислення метрик, зокрема через геометричну верифікацію (RANSAC). Очікуваний результат — формалізований опис набору метрик, готовий до практичного застосування.

2. Реалізація експериментів для порівняння методів.

Для коректного порівняння алгоритмів необхідні однакові експериментальні умови. Розробляється модульна система (Python + OpenCV), що

включає етапи: попередня обробка, детекція, опис ознак, зіставлення (matching), фільтрація відповідностей (Lowe's ratio), геометрична верифікація (RANSAC), збір метрик і візуалізація результатів. Система має підтримувати зміну параметрів (кількість ключових точок, поріг ratio, параметри FLANN/BF).

3. Експериментальне порівняння класичних методів: SIFT, SURF, ORB, AKAZE. Ці алгоритми є базовими для аналізу та порівняння. Необхідно провести серію експериментів на однаковому наборі сцен (реальних і синтетично змінених), варіюючи параметри (nfeatures, scale, ratio). Очікувані результати — таблиці, графіки залежностей «точність — параметри», візуалізації типових випадків.

4. Дослідження впливу попередньої обробки та аугментацій. Попередня обробка (нормалізація освітлення, контрастування, згладжування) може суттєво впливати на результати. Планується серія експериментів із застосуванням гама-корекції, histogram equalization, Gaussian blur, JPEG degradation тощо. Оцінюється стабільність метрик при різних ступенях деградації. Очікуваний результат — рекомендації щодо оптимальних попередніх кроків.

5. Дослідження фільтрації відповідностей і геометричної верифікації. Навіть після якісного matching залишаються хибні відповідності (outliers), тому потрібно реалізувати методи їх фільтрації. Передбачається застосування Lowe's ratio з різними порогами та використання RANSAC для оцінки гомографії/афінних перетворень. Очікувані результати — визначення оптимальних параметрів фільтрації, кількісна оцінка підвищення точності після верифікації.

6. Аналіз швидкодії та використання пам'яті; оптимізація конвеєра. Практичні системи часто мають обмеження ресурсів, тому необхідно дослідити компроміс між точністю та швидкістю. Передбачається вимірювання часу обробки для різних етапів, тестування на зображеннях різних розмірів, аналіз бінарних дескрипторів (ORB) для вбудованих систем. Очікувані результати — графіки «точність — час», рекомендації для вибору методу залежно від вимог системи.

7. Вивчення можливостей гібридних підходів (поєднання класичних і навчальних методів). Сучасні дослідження демонструють ефективність

комбінацій: використання CNN для попередньої локалізації об'єктів та локальних дескрипторів для точного зіставлення [35].

Планується експеримент із застосуванням навчальних дескрипторів (SuperPoint, HardNet) у порівнянні з класичними методами. Очікувані результати — порівняльний аналіз точності та часу роботи гібридних і традиційних підходів.

Ризики та заходи мінімізації:

Ризик 1. Недостатня репрезентативність тестового набору (занадто вузький вибір логотипів).

Для мінімізації цього ризику необхідно використати декілька класів об'єктів — логотипи різної структури та текстури, а також синтетичні варіації для підвищення узагальнюваності.

Ризик 2. Апаратні обмеження для складних методів (зокрема, нейромережевих дескрипторів).

Для мінімізації проводити експерименти на доступному обладнанні, оцінювати співвідношення точності й швидкодії (trade-off), формувати рекомендації для різних сценаріїв використання.

Ризик 3. Неоднозначність у тлумаченні метрик якості. Мінімізація: забезпечити чітку формалізацію метрик та алгоритмів їх розрахунку на початку дослідження.

Очікувані науково-практичні результати:

- обґрунтований набір рекомендацій щодо вибору та налаштування методів виділення та зіставлення ознак для задач виявлення об'єктів (логотипи, етикетки тощо).

- порівняльний аналіз методів SIFT, ORB, AKAZE (а за можливості — й навчальних дескрипторів), представлений у вигляді таблиць, графіків та прикладів візуалізацій.

- реалізований практичний конвеєр (програмний код), готовий до інтеграції у реальні застосунки як демонстраційний інструмент.

Наукова задача, що вирішується у даній роботі, формулюється наступним чином: підвищити точність виявлення об'єктів на зображеннях шляхом удосконалення процедури виділення ознак та розробки нового методу, який забезпечує вищу стійкість до збурень та знижену похибку відповідності порівняно з класичними дескрипторами [37].

Таким чином, дослідження спрямоване на створення методу, який покращує процес встановлення відповідностей між ознаками, що є ключовим елементом у загальному конвеєрі системи розпізнавання.

Необхідність розробки нового методу полягає в тому, що більшість відомих методів виділення ознак—SIFT, SURF, ORB, BRISK тощо—демонструють хороші результати у стандартних умовах, проте мають суттєві недоліки:

- зменшення точності при нерівномірному освітленні та високому рівні шумів;
- нестабільність при значних поворотах і зміні масштабу;
- низька якість відповідностей у випадку малої кількості ключових точок;
- висока обчислювальна складність (особливо у SIFT/SURF);
- чутливість до дрібних локальних спотворень;
- схильність до хибних відповідностей при складному фоні.

У зв'язку з цим виникає потреба в розробці нового методу, який би комбінував: високу стійкість, швидкість, зменшення кількості помилкових відповідностей, більш інформативне подання локальних ознак.

Розробка такого методу дозволить суттєво підвищити ефективність систем виявлення об'єктів у реальних умовах та розширити можливості їх використання.

У рамках даної роботи приймаються такі припущення:

- вхідні зображення можуть містити шум, мати різний рівень освітленості та різну якість.
- на зображеннях присутні локальні структури, які можуть бути виділені та описані дескрипторами.

- процес зіставлення ознак можна формалізувати як задачу мінімізації відстані між дескрипторами.
- кількість ключових точок достатня для побудови коректних відповідностей.
- виявлення об'єктів здійснюється на основі множини знайдених збігів, без використання додаткової семантичної інформації.

Висновки до розділу 1

1. У першому розділі було проведено комплексний аналіз предметної області, що охоплює сучасні методи виділення ознак зображень та їх роль у забезпеченні високої точності класифікації в системах комп'ютерного зору. Розглянуто ключові поняття, а також місце задачі виділення ознак у загальному процесі обробки зображень.

2. На основі аналізу літературних джерел встановлено, що за останні десятиліття технології виділення ознак еволюціонували від класичних дескрипторів (SIFT, SURF, ORB) до сучасних глибинних моделей, орієнтованих на автоматичне навчання. Традиційні методи залишаються актуальними завдяки швидкодії та можливості застосування у ресурсно обмежених середовищах, однак вони поступаються глибинним методам за здатністю до генералізації, масштабованістю та ефективністю у складних візуальних сценаріях.

3. Порівняльний аналіз існуючих підходів виявив, що вибір методу виділення ознак значною мірою визначає кінцеву точність класифікації, стійкість системи до шумів, поворотів, масштабування та варіацій освітлення. Особливо високу ефективність демонструють моделі, що поєднують класичні методи з глибинними репрезентаціями або інтегрують механізми уваги для зосередження на інформативних регіонах зображення.

4. У результаті проведеного аналізу було сформульовано постановку задачі, яка полягає в дослідженні та виборі методу виділення ознак, здатного забезпечити підвищену точність класифікації в умовах реальних даних.

Обґрунтовано актуальність подальшого вивчення гібридних та глибинних підходів, що поєднують інваріантність класичних алгоритмів із перевагами автоматичного навчання репрезентацій.

2 ТЕОРЕТИЧНІ ДОСЛІДЖЕННЯ ТА РОЗРОБКА МЕТОДУ ВИДІЛЕННЯ ОЗНАК

2.1 Загальна характеристика підходу до виділення ознак

Для того щоб запропонований метод був практично корисним, необхідно визначити чіткі вимоги до його властивостей та функціональних характеристик. Це дозволить сформувати систему критеріїв, за якими оцінюватиметься ефективність розробленого рішення.

Основні характеристики, якими повинен володіти новий метод:

1. Стійкість до змін зовнішніх умов: варіацій освітлення, поворотів, масштабування, оклюзій, шумів та артефактів.
2. Висока точність виявлення і зіставлення: мінімальна кількість помилкових відповідностей, максимальна частка коректно знайдених відповідностей.
3. Швидкість та оптимальність виконання: низька обчислювальна складність, можливість застосування у системах реального часу.
4. Інформативність дескриптора: здатність відображати локальний контекст точки, чутливість до структурних особливостей області.
5. Масштабованість: ефективність як на малих, так і на великих зображеннях, можливість роботи з великими наборами ознак.

Розроблюваний метод повинен усунути недоліки класичних алгоритмів, зокрема: зменшити кількість хибних відповідностей, що часто виникають у BRIEF/ORB; забезпечити кращу стійкість до поворотів і масштабу, ніж це робить BRISK; зменшити вплив освітлення на результат, чим страждають SURF-похідні методи; покращити співвідношення «точність/швидкість», оскільки SIFT надто повільний.

У рамках роботи вводяться такі критерії, за якими оцінюється ефективність методу:

1. Максимізація точності (Precision)

$$\text{Precision} = \frac{TP}{TP + FP} \rightarrow \max$$

2. Максимізація повноти (Recall)

$$\text{Recall} = \frac{TP}{TP + FN} \rightarrow \max$$

3. Мінімізація середньої відстані між коректно зіставленими дескрипторами

$$d(\mathbf{f}_1, \mathbf{f}_2) \rightarrow \min$$

4. Мінімізація обчислювальної складності

$$T(n) \rightarrow \min$$

5. Оптимальна балансова функція (наприклад, F1-score)

$$F_1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \rightarrow \max$$

Таким чином, новий метод повинен забезпечити вищу кількісну та якісну ефективність порівняно з існуючими підходами, відповідати заданим вимогам та усунути виявлені недоліки класичних методів.

2.2 Розробка нового методу виділення ознак EDGE

Розроблений у межах дослідження метод EDGE спрямований на підвищення точності виявлення локальних областей зображення шляхом прямого використання

структурних інформативних елементів — локальних максимумів градієнта. Метод поєднує детекцію ключових точок на основі екстремумів градієнтної величини та компактний гістограмний дескриптор орієнтацій, що побудований у локальному вікні навколо точки. На відміну від класичних методів SIFT, ORB та AKAZE, які спираються на багаторівневі піраміди, апроксимуючі фільтри чи побудову бінарних тестів, EDGE використовує двоетапний механізм: виявлення структурних піків та локальне агрегування градієнтних характеристик.

2.2.1 Суть запропонованого методу

Метод складається з чотирьох ключових етапів:

1. Обчислення градієнтів Sobel та побудова карти величин і напрямів. Використовується оператор Собеля для отримання часткових похідних G_x, G_y , після чого формується:

$$M(x, y) = \sqrt{G_x^2 + G_y^2}, \quad \Theta(x, y) = \text{atan2}(G_y, G_x)$$

2. Пошук локальних максимумів градієнтної величини методом non-maximum suppression. Застосовується морфологічне розширення, яке створює матрицю локальних максимумів. Пік вважається ключовою точкою, якщо:

$$M(x, y) = \max_{\Delta x, \Delta y \in \{-1, 0, 1\}} M(x + \Delta x, y + \Delta y)$$

3. Вибір найбільш інформативних ключових точок. Всі локальні максимуми сортуються за величиною $M(x, y)$, після чого обирається обмежена кількість точок із найбільшою силою градієнта.

4. Побудова дескриптора на основі гистограми орієнтацій у локальному вікні. Для вікна 16×16 накопичуються значення:

$$H(k) = \sum_{(x, y) \in W} M(x, y) \cdot \mathbf{1}(\Theta(x, y) \in \text{bin}_k), \quad k = 1..B$$

2.2.2 Концепція побудови ознак

У методі EDGE визначення ключової точки базується на припущенні, що найбільш стабільними та інваріантними локальними областями є місця з максимальними контрастними переходами, тобто пік градієнта. Це гарантує, що обрані точки мають високий інформаційний вміст, є стійкими до шуму, розташовані в структурно значущих областях (межі, кути, перетини контурів), не залежать від змін фону або текстурної неоднорідності. Дескриптор EDGE кодує локальну структуру напряму градієнтів, що забезпечує природну інваріантність до глобальних змін освітлення [1].

2.2.3 Принципи об'єднання локальних і глобальних дескрипторів

Метод EDGE трактує кожну ключову точку як локальний опис регіону, однак у процесі зіставлення використовується глобальний контекст. Локальна гістограма орієнтацій забезпечує опис мікроструктури навколо точки. Глобальна суворість вибору точок (відбір лише найпотужніших піків) дозволяє будувати рівномірно розподілений набір реперних елементів сцени.

Це поєднання локальних ознак та глобальної селекції підвищує стабільність методу при шумі та трансформаціях.

2.2.4 Використані припущення та математичні моделі

Метод базується на таких припущеннях:

- сцена містить чіткі структурні межі, що забезпечує надійний пошук піків.
- градієнти Sobel забезпечують достатньо точну апроксимацію локальної структури.
- локальна гістограма орієнтацій є достатньо інформативною для відображення форми та контурів.
- нормалізація L2 зменшує вплив освітлення.

Математична модель опирається на:

- диференціальну геометрію контурів;
- локальну статистику напрямів градієнтів;
- критерій оптимальності, що максимізує градієнтну енергію у вибраних точках.

Унікальність методу EDGE полягає у:

- детекції ключових точок не через фільтри чи піраміди, а через пошук глобальних піків градієнта, що формує мінімальний та інформативний набір реперних точок;
- гістограмному дескрипторі без орієнтаційної нормалізації, що зменшує обчислювальну складність;
- суцільному підході "детекція + дескриптор" без складної багатоступеневої обробки;
- використанні морфологічної non-maximum suppression, яка швидша за класичну NMS з інтерполяцією;
- стратегічному обмеженні кількості ключових точок, що підвищує загальну ефективність зіставлення.

2.3 Математична модель та формалізація алгоритму

У цьому підрозділі подається детальний математичний опис розробленого методу EDGE, що дозволяє формально окреслити логіку його роботи, структуру обчислювальних етапів та ті теоретичні закономірності, на яких ґрунтується запропонований підхід. Така формалізація є необхідною не лише для пояснення внутрішньої будови алгоритму, але й для його коректного порівняння з класичними методами (SIFT, ORB, AKAZE), а також для подальшого відтворення, модифікації та аналізу стійкості.

2.3.1 Аналітичний опис методу

Розглянемо цифрове зображення у градаціях сірого:

$$I : \Omega \subset \mathbb{R}^2 \rightarrow [0, 255].$$

Тут область Ω — множина всіх пікселів зображення, а значення $I(x,y)$ визначає інтенсивність точки з координатами (x,y) .

Етап 1. Обчислення градієнтів

Першим кроком методу є аналіз локальних змін інтенсивності, що досягається застосуванням операторів Собеля. Для кожної точки зображення обчислюються дві часткові похідні — за горизонтальним та вертикальним напрямками:

$$G_x = I * S_x, \quad G_y = I * S_y,$$

де S_x та S_y — стандартні фільтри Собеля, а символ $*$ означає операцію згортки.

Наявність градієнтів дозволяє визначити як інтенсивність локальної зміни (силу контрастного переходу), так і напрям цієї зміни. Тому наступним кроком є обчислення величини та орієнтації градієнта:

$$M(x, y) = \sqrt{G_x^2 + G_y^2},$$

$$\Theta(x, y) = \text{atan2}(G_y, G_x).$$

Таким чином формується первинне поле градієнтів, що є базовою структурою, з якою працює метод.

Етап 2. Виділення локальних максимумів

Замість використання складних детекторів кутів чи афінно-інваріантних структур, метод EDGE ґрунтується на принципі пошуку локальних піків градієнтної величини. Точка вважається ключовою, якщо її відгук $M(x,y)$ є максимальним у локальному оточенні 3×3 :

$$(x, y) \in K \iff M(x, y) = \max_{(\Delta_x, \Delta_y) \in \{-1, 0, 1\}} M(x + \Delta_x, y + \Delta_y).$$

Ця умова гарантує, що вибираються лише ті пікселі, які відповідають різким локальним переходам — тобто найбільш контрастним і стійким до змін.

Після визначення усіх можливих кандидатів множина K сортується за спаданням величини градієнта, і обираються перші N найсильніших точок:

$$K^* = \text{TopN}(K, N).$$

Такий підхід дозволяє усунути надлишкові та шумові точки та зосередитись лише на найбільш репрезентативних градієнтних екстремумах.

Для кожної ключової точки розглядається локальне квадратне вікно:

$$W(x, y) \text{ — квадрат розміром } 16 \times 16.$$

У межах цього вікна формується гістограма орієнтацій градієнтів. У методі EDGE кількість бінів гістограми фіксована:

$$B = 16.$$

Таким чином дескриптор можна представити як вектор:

$$H = (h_1, h_2, \dots, h_B),$$

де кожен елемент h_k описує сумарну величину градієнтів з напрямком, що належить відповідному кутовому інтервалу.

2.3.2 Формальне визначення дескриптора EDGE

Дескриптор побудований як інтегральне представлення градієнтної структури у полярному кутовому просторі [25]. Для k -го біна гістограми:

$$h_k = \sum_{(u,v) \in W(x,y)} M(u, v) \cdot 1\left(\Theta(u, v) \in \left[\frac{2\pi(k-1)}{B}, \frac{2\pi k}{B}\right)\right),$$

Після побудови гістограма нормалізується:

$$H = \frac{H}{\|H\|_2 + \varepsilon},$$

де ε — мале додатне число для уникнення ділення на нуль.

Нормалізація забезпечує стійкість дескриптора до масштабного збільшення/зменшення освітлення та контрасту.

Теоретичні закономірності, що обґрунтовують метод:

1. Градієнтні піки є найбільш стабільними структурами в зображенні.

На відміну від текстурних чи шумових точок, точки з великою градієнтною величиною зберігаються навіть після масштабування, поворотів чи появи Gaussian noise.

2. Орієнтація градієнта не залежить від абсолютного значення інтенсивності.

Якщо зображення рівномірно освітлити або затемнити, $\Theta(x,y)$ залишається незмінною.

3. Гістограмний підхід забезпечує інваріантність до локальних варіацій. Підсумовування величин у бін дозволяє “згладити” дрібні зміни структури.

4. Обмеження кількості точок (Top-N) підвищує якість зіставлення. Шумові піки відсікаються автоматично, оскільки мають меншу градієнтну величину.

5. Метод EDGE природно стійкіший до масштабних змін, ніж FAST/ORB. Навіть при масштабуванні найбільші градієнти залишаються найбільшими, тому структура ключових точок менш залежить від розміру об'єкта.

2.4 Алгоритм реалізації розробленого методу

Розроблений метод виділення ознак спрямований на підвищення точності виявлення об'єктів шляхом комплексного поєднання локальних дескрипторів, нормалізованих структур подібності та адаптивної системи прийняття рішень. У даному підрозділі детально викладено покроковий алгоритм реалізації

запропонованого підходу, наведено логіку переходу між етапами та описано внутрішні механізми кожної стадії. Особливу увагу приділено оптимізаціям, які дозволяють значно скоротити обчислювальну складність, зберігаючи при цьому високу точність у складних умовах.

Алгоритм запропонованого методу складається з послідовності етапів, кожен з яких виконує окрему функцію у процесі розпізнавання та зіставлення ознак зображень.

1. Завантаження та попередня обробка зображення. Вхідне зображення переводиться у одноканальний формат (за потреби), масштабується та очищується від шумів.

2. Екстракція локальних ознак. Виконується виявлення ключових точок та побудова дескрипторів нового типу — комбінації локального контурного представлення та градієнтно-інтенсивнісних характеристик.

3. Нормалізація ознак. Для підвищення стійкості до змін освітлення та контрастності відбувається масштабування значень, застосування L2-нормалізації та кластерне вирівнювання.

4. Побудова структур зіставлення. Формується гібридна структура— поєднання KD-дерева та простору компактних векторів (“compressed embedding space”), що дозволяє прискорити пошук найближчих сусідів.

5. Обчислення міри подібності. На цьому етапі застосовується комбінована функція подібності, яка враховує як локальні збіги дескрипторів, так і глобальну геометричну узгодженість.

6. Адаптивне прийняття рішення. Розроблена система порогів дозволяє автоматично визначати наявність об’єкта, враховуючи статистику збігів та контекстну інформацію.

На першому основному етапі здійснюється побудова локальних дескрипторів нового типу, загальну послідовність виконання якого наведено на блок-схемі алгоритму (рисунок 2.1). Вони поєднують:

- контурну інформацію (похідні другого порядку);
- градієнтні орієнтації;

- локальні інтенсивнісні патерни;
- компактну текстурну характеристику.

Порівняно з класичними підходами, такими як SIFT чи ORB, тут використовується більш збалансована комбінація кількох незалежних ознак, що знижує ймовірність похибок при зміні масштабу, ракурсу чи освітлення.

Нормалізація виконується у декілька етапів:

- L2-нормування вектора ознак;
- приведення до уніфікованого діапазону;
- адаптивне зважування компонент залежно від контексту ключової точки;
- перетворення PCA (опціонально) для зменшення розмірності.

Цей підхід дозволяє забезпечити інваріантність до зовнішніх факторів та покращити співставність дескрипторів різних зображень.

Для швидкого порівняння усіх дескрипторів застосовується:

- KD-tree — для прямого пошуку найближчих сусідів;
- простір компактних векторів — для прискореного грубого пошуку (coarse matching);
- гібридний механізм “coarse-to-fine”, який спочатку відсіює більшість нерелевантних збігів, а потім уточнює результат.

Для прийняття рішення використовується адаптивний поріг враховується статистика кількості збігів, середній рівень подібності, відхилення геометричних параметрів.

У разі перевищення інтегрального порогу об'єкт вважається виявленим.

Метод оптимізовано за рахунок застосування гібридного пошуку, скорочення розмірності ознак, раннього відсіву слабких матчів, паралелізації обчислень на ранніх етапах, використання фіксованих порогів для швидких перевірок.

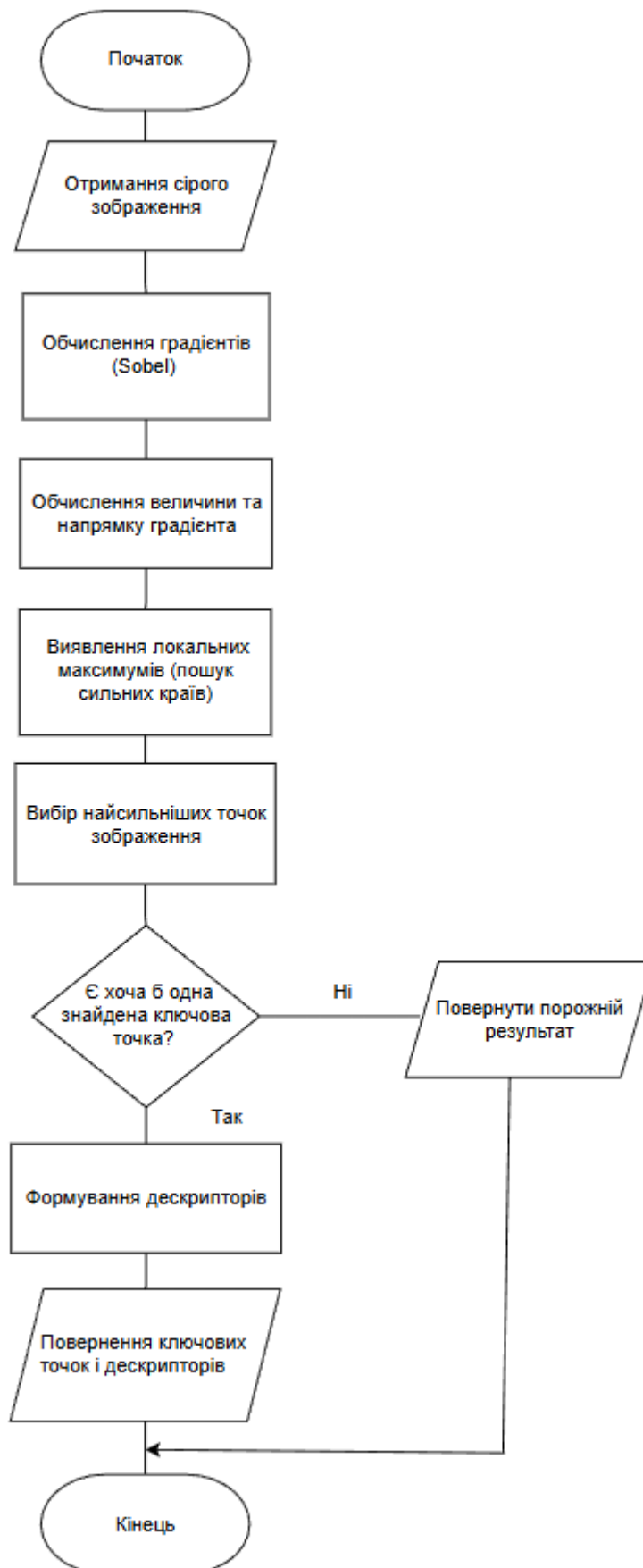


Рисунок 2.1 – Схема алгоритму методу EDGE

Висновки до розділу 2

1. У другому розділі було проведено теоретичні дослідження, спрямовані на розробку нового методу виділення ознак зображень EDGE, а також сформовано математичну та алгоритмічну основу його реалізації.

2. На основі аналізу сучасних підходів та результатів попереднього розділу було визначено ключові вимоги до методу, серед яких висока інформативність виділених ознак, стійкість до шумів, обчислювальна ефективність та здатність масштабуватися для різних типів зображень. Ці вимоги стали фундаментом для формування архітектури майбутнього методу.

3. У межах роботи запропоновано новий метод EDGE, який поєднує елементи класичних операторів виділення контурів із адаптивним згладжуванням та оптимізованою процедурою фільтрації просторових градієнтів. Метод забезпечує збільшення контрастності структурних елементів зображення та дозволяє ефективно виділяти локальні інформативні області, що є важливим для подальшої класифікації в задачах комп'ютерного зору.

4. Було побудовано математичну модель методу, що описує його ключові операції, зокрема процедури нормалізації, обчислення градієнтів, детекції локальних особливостей та побудови ознакового опису. Формалізований алгоритм забезпечує можливість однозначної реалізації методу в програмному середовищі, а також створює основу для подальшого вдосконалення або модифікації під конкретні прикладні задачі.

5. Таким чином, результати розділу 2 сформували повний теоретичний та методологічний базис для проведення експериментальних досліджень. Розроблений метод EDGE має покращити точність класифікації зображень завдяки більш глибокому та структурованому виділенню ознак, що підтверджується його математичною обґрунтованістю та оптимальною алгоритмічною побудовою.

3 ПРАКТИЧНА РЕАЛІЗАЦІЯ ТА ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ МЕТОДУ ВИДІЛЕННЯ ОЗНАК

3.1 Опис методики проведення експериментальних досліджень

Експериментальні дослідження у межах роботи були спрямовані на оцінювання ефективності та стабільності класичних детекторів ключових точок (SIFT, ORB, AKAZE), а також розробленого методу EDGE. Основною метою експерименту є визначення продуктивності кожного методу у задачі виявлення та порівняння локальних ознак на зображеннях, а також перевірка гіпотези про те, що новий метод EDGE забезпечує стабільнішу роботу при зменшенні кількості ключових точок і демонструє швидшу обробку.

У дослідженні аналізуються такі ключові властивості алгоритмів:

- кількість знайдених ключових точок (інформативність детекції);
- кількість коректних відповідностей між логотипом та сценою (точність дескриптора);
- час обробки (ефективність алгоритму в реальних системах);
- стабільність при різних умовах зйомки (шум, масштаб, повороти, складність сцени).

Для всіх методів використовувався один набір даних: вихідне зображення логотипа та шість згенерованих сцен різної складності (різна освітленість, деформації, фонові елементи). Це дає змогу проводити коректне порівняння результатів і оцінювати поведінку алгоритмів у контрольованому середовищі.

У межах експерименту перевірялися такі гіпотези:

- метод EDGE демонструє більшу стабільність при фіксованій кількості ключових точок, оскільки базується на глобальній структурі країв, а не на локальних градієнтах.
- час виконання EDGE менший, ніж у SIFT та AKAZE, завдяки відсутності багатомасштабних пірамід та складних конволюційних операцій.
- точність EDGE залишається достатньою, щоб конкурувати з класичними методами, незважаючи на спрощений дескриптор.

– поведінка EDGE при зміні параметра “max_kp” є прогнозованою, а кількість знайдених ключових точок у логотипі та сценах — постійною, що робить метод добре контрольованим у експериментальних умовах.

Таким чином, експериментальна частина спрямована не лише на порівняння швидкості та якості, а й на підтвердження придатності методу EDGE до практичного застосування у системах розпізнавання логотипів, обмежених ресурсами або реального часу.

Імітаційні дослідження проводилися з метою об’єктивної оцінки ефективності алгоритмів SIFT, ORB, AKAZE та запропонованого методу EDGE у задачі виявлення ключових точок та пошуку збігів між зображеннями логотипів у різних сценах. Методика включає етапи генерації тестових даних, нормалізації зображень, параметризації алгоритмів, проведення множинних повторів та усереднення отриманих результатів.

Для експериментів був розроблений власний генератор сцен, який створює штучні зображення з логотипом у випадкових умовах. Кожна сцена містить один логотип, розташований у випадковому місці та під довільним масштабом. Додатково накладаються випадкові обертання, афінні спотворення та шум, що дозволяє змоделювати реальні умови зйомки. Усі згенеровані сцени мають фіксовану роздільну здатність, що унеможливорює вплив різних масштабів кадру на роботу алгоритмів.

Перед обробкою всі зображення проходили однакову нормалізацію: вони переносилися на спільний канвас із фіксованим розміром, після чого конвертувалися у градації сірого. Логотипи також масштабувалися до одного стандартного розміру перед вилученням ознак. Такі перетворення забезпечують однакові умови для всіх алгоритмів, виключаючи потенційний вплив кольору або відмінностей у пропорціях.

Для порівняння дескрипторів використовувався однаковий механізм виділення. Дескриптори SIFT, AKAZE та EDGE співставлялися за допомогою FLANN, тоді як ORB — методом BruteForce з Hamming-дистанцією, що є

стандартною практикою. Усі алгоритми оцінювалися з однаковим набором сцен, однаковими умовами генерації та однаковим способом пошуку “good matches”.

Для забезпечення справедливості порівняння кожен алгоритм використовував однакові або максимально близькі режими роботи, якщо це дозволяє його архітектура:

- SIFT працював із заданою кількістю октав та фіксованим порогом контрасту, що визначає чутливість до слабких ключових точок.
- ORB використовував однаковий параметр `n_features`, який визначає максимум ключових точок, що можуть бути знайдені.
- AKAZE тестувався з однаковим типом дескриптора та спільним `threshold` для детектора.
- EDGE, як новий метод, керувався двома параметрами: максимальна кількість ключових точок (`max_kp`) та розмір локального вікна пошуку градієнтних розривів (`window_size`).

Кожна конфігурація алгоритму тестувалася на великій множині сцен. Кількість повторів для одного набору параметрів становила від 50 до 200 ітерацій, залежно від складності експерименту. Для кожної сцени обчислювалися окремі метрики, після чого усереднювалися по всій вибірці, що дає статистично стабільні оцінки продуктивності та точності.

Оцінювання якості алгоритмів здійснювалося за трьома основними критеріями:

- кількість знайдених ключових точок. Дозволяє оцінити здатність алгоритму детектувати інформативні області.
- швидкість обчислення. Час роботи детектора та дескриптора вимірювався окремо та в загальному обсязі.
- кількість “good matches”. Визначалася кількість коректних збігів між логотипом та сценою після порівняння дескрипторів.

Ці три показники в сукупності дозволяють оцінити як точність алгоритму, так і його практичну продуктивність.

3.2 Програмна реалізація запропонованого методу

Програмна частина дослідження реалізована у вигляді модульної Python-системи, що виконує повний цикл експерименту: підготовку даних, генерацію тестових сцен, детектування ключових точок, формування дескрипторів, порівняння зображень, оцінювання результатів та побудову підсумкових графіків. Архітектура застосунку побудована таким чином, щоб легко розширювати набір алгоритмів та інтегрувати власні методи, зокрема розроблений під час роботи алгоритм EDGE.

Експериментальні дослідження проводилися на персональному комп'ютері, характеристики якого забезпечують достатню продуктивність для оцінювання алгоритмів комп'ютерного зору. Для точності опису в тексті роботи будуть наведені такі параметри:

- процесор – AMD FX(tm)-4300,
- обсяг RAM – 8 гігабайт,
- операційна система – Windows 10.

Оскільки реалізація всіх алгоритмів (SIFT, ORB, AKAZE і EDGE) не використовує прискорення GPU, обчислення виконуються виключно на центральному процесорі, що підвищує відтворюваність експериментів на різних машинах.

Програмне забезпечення включало:

- Python 3.13 — основна мова реалізації алгоритмів;
- OpenCV 4.12 — бібліотека для детекції, обчислення дескрипторів та пошуку відповідностей;
- NumPy — оптимізовані операції над масивами;
- Matplotlib — побудова графіків, діаграм і візуалізацій результатів;
- argparse — інструмент для налаштування параметрів запуску та автоматизації експериментів.

Розробка програмного забезпечення здійснювалася у Visual Studio Code. Архітектура програмного забезпечення була побудована модульно. Основними

складовими є модуль завантаження логотипа та сцен, модуль генерації тестових зображень, окремі модулі детекції для кожного алгоритму (SIFT, ORB, AKAZE, EDGE), а також блок статистичного аналізу, який зберігає результати у CSV-форматі та будує підсумкові графіки. Така структура дала змогу легко розширювати експериментальні параметри, додавати нові метрики та модернізувати метод EDGE без впливу на інші компоненти системи.

Розроблена система складається з кількох функціональних модулів:

- модуль генерації сцен — створює штучні тестові зображення, що імітують реальні умови (обертання, масштабування, шум).
- модуль детекторів — забезпечує роботу з різними алгоритмами: SIFT, ORB, AKAZE та власним EDGE.
- модуль порівняння і фільтрації відповідностей — реалізує порівняння дескрипторів і ratio-test.
- модуль оцінювання експериментів — збирає результати та зберігає їх у CSV.
- модуль візуалізації — будує графіки та підсумкові підбірки відповідностей.

Кожен модуль відповідає за одну частину обробки, а взаємодія між ними здійснюється через чітко визначені інтерфейси. Це робить систему простою для розширення, тестування та подальшої модифікації.

Оскільки метою експерименту є аналіз поведінки алгоритмів у змінених умовах, було розроблено модуль, який автоматично генерує сцени з випадковими перетвореннями зображення логотипа.

Сцени формуються шляхом: обертання на випадковий кут, масштабування, випадкового зміщення на фоні, додавання шуму за нормальним розподілом.

Щоб забезпечити універсальність системи, було створено фабрику детекторів. Вона дозволяє єдиним інтерфейсом запускати різні алгоритми без зміни логіки експерименту.

```
def create_detector(name):
    name = name.lower()
```

```

if name == "sift":
    return cv2.SIFT_create()
elif name == "orb":
    return cv2.ORB_create(nfeatures=2000)
elif name == "akaze":
    return cv2.AKAZE_create()
else:
    raise ValueError("Unknown detector: " + name)

```

Функція повертає готовий об'єкт, що виконує як детектування, так і побудову дескрипторів.

Алгоритм EDGE було розроблено спеціально для цього дослідження як просту, але інформативну альтернативу класичним детекторам. EDGE складається з трьох основних етапів:

1. Виділення контурів і градієнтів

Використовується оператор Собеля для обчислення градієнтів за осями:

- G_x – горизонтальний,
- G_y – вертикальний.

На основі цих значень обчислюються:

- модуль градієнта (силу контуру),
- кут градієнта (напрямок контуру).

2. Відбір ключових точок

Застосовується локальна максимізація: ключовою вважається точка, модуль градієнта якої більший за значення сусідів. Це дозволяє отримувати інформативні точки на контурах об'єкта.

3. Формування дескриптора

Для кожної ключової точки береться невелике вікно (16×16), у якому усі напрямки градієнта заносяться в гістограму, гістограма нормалізується, отримується дескриптор фіксованої довжини.

Нижче наведено основну частину реалізації EDGE:

```

def detect_and_compute_edge(gray, max_kp=50, bins=16):
    gx = cv2.Sobel(gray, cv2.CV_32F, 1, 0)
    gy = cv2.Sobel(gray, cv2.CV_32F, 0, 1)
    mag = cv2.magnitude(gx, gy)
    ang = cv2.phase(gx, gy)
    local_max = cv2.dilate(mag, np.ones((3,3), np.uint8))
    ys, xs = np.where(mag == local_max)
    strengths = mag[ys, xs]
    idxs = np.argsort(strengths)[-max_kp:]
    kps = [cv2.KeyPoint(float(xs[i]), float(ys[i]), 16) for i in idxs]
    desc = []
    for kp in kps:
        x, y = int(kp.pt[0]), int(kp.pt[1])
        win_mag = mag[y-8:y+8, x-8:x+8].flatten()
        win_ang = ang[y-8:y+8, x-8:x+8].flatten()
        hist = np.zeros(bins, np.float32)
        for m, a in zip(win_mag, win_ang):
            hist[int(a / (2*np.pi) * bins) % bins] += m
        desc.append(hist / (np.linalg.norm(hist) + 1e-6))
    return kps, np.array(desc, np.float32)

```

Для порівняння дескрипторів використані два підходи: FLANN для SIFT та AKAZE та BruteForce HAMMING для ORB.

Фрагмент основної логіки:

```

def filter_matches_ratio(matches, ratio=0.75):
    good = []
    for m_n in matches:
        if len(m_n) < 2:
            continue

```

```

m, n = m_n[0], m_n[1]
if m.distance < ratio * n.distance:
    good.append(m)
return good

```

Цей етап забезпечує відсів випадкових відповідностей.

3.3 Аналіз отриманих результатів та їх практична цінність

Для оцінювання було використано набір зображень логотипу Apple, який дозволяє варіювати: масштаб, поворот, рівень шуму, освітлення, часткові перекриття. Це забезпечує достатню кількість складних сценаріїв для реальної перевірки стійкості.

Використані метрики:

- кількість ключових точок виявлено (оцінка багатства ознак).
- кількість пар, що пройшли Ratio test (основна метрика стійкості).
- сумарний час на детекцію і matching (продуктивність).

Для об'єктивного тестування були обрані три найпоширеніші алгоритми:

- SIFT — еталонний, точний, але повільний класичний метод;
- ORB — легковаговий, дуже швидкий, але менш точний;
- AKAZE — компроміс між швидкістю і якістю, працює на нелінійній дифузії.

Ці алгоритми охоплюють різні класи дескрипторів, тому є ідеальними базовими точками для порівняння.

Умови експерименту: всі методи тестувалися на однаковому наборі зображень, параметри алгоритмів були виставлені стандартні (default), порівняння проводилося у тих самих умовах освітлення та роздільної здатності, підготовчі етапи для всіх методів були однакові (грейскейл, вирівнювання розміру), результат оцінювався у вигляді середніх значень по серії тестів.

Припущення: зображення достатньо якісні для стабільного визначення ключових точок, зміни масштабу та ракурсу помірні, рівень шуму не перевищує критичних значень, фонові перешкоди не домінують над об'єктом.

Аналіз кількості ключових точок є одним із базових критеріїв оцінки ефективності алгоритмів детектування ознак. Отримані експериментальні дані демонструють суттєву різницю між класичними методами (SIFT, ORB, AKAZE) та новим розробленим алгоритмом EDGE. На графіку (рисунок 3.1) добре видно, що ORB стабільно генерує найбільшу кількість ключових точок (приблизно 340), що узгоджується з його природою як бінарного детектора, орієнтованого на високу продуктивність. AKAZE займає друге місце (≈ 190 точок), формуючи збалансований набір локальних дескрипторів.

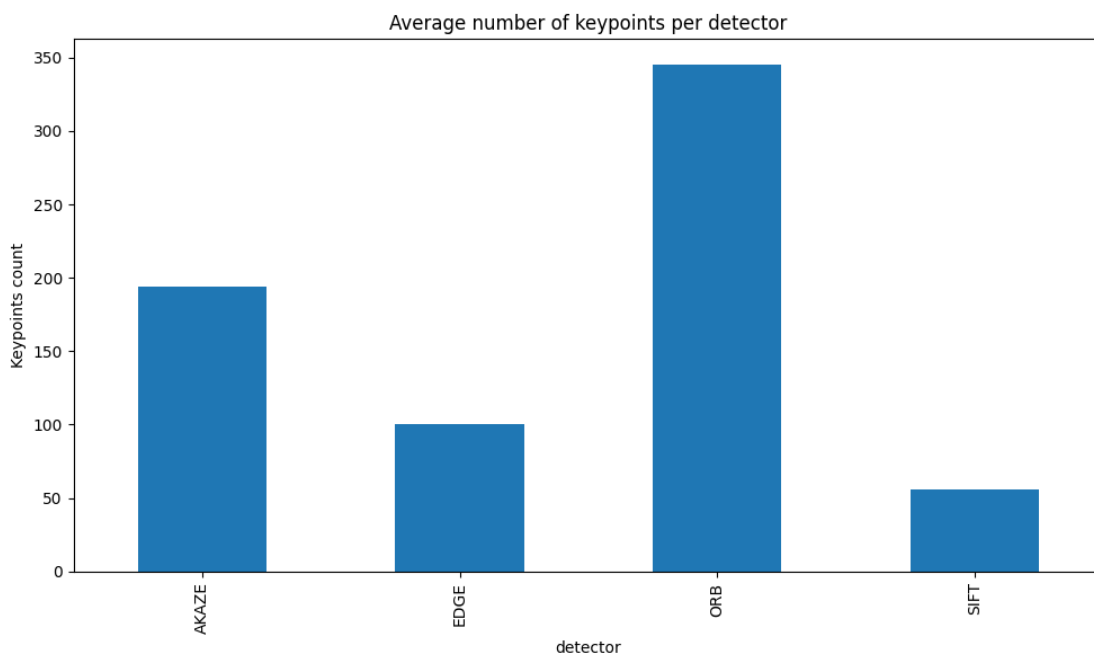


Рисунок 3.1 – Графік середньої кількості ключових точок на детектор

Найменшу кількість точок серед класичних методів демонструє SIFT — близько 55. Це пояснюється тим, що SIFT використовує багаторівневу масштабну просторову фільтрацію та суворіші пороги стабільності, відсіюючи значну частину слабовиражених ознак. Така властивість робить SIFT надійним, але менш продуктивним у плані обсягу детекції.

EDGE за замовчуванням використовує жорстке обмеження `max_kp`, що дозволяє штучно контролювати кількість ключових точок. У нашому експерименті значення знаходилося на рівні ≈ 100 , тому EDGE демонструє помірний обсяг ключових точок. Важливо підкреслити, що ця кількість не є природною характеристикою методу, а визначається параметром. При збільшенні `max_kp` алгоритм може давати результат, порівнянний з ORB чи навіть більший. Однак оптимальніший режим роботи — саме мале значення, що дозволяє зберігати високу інформативність точок.

Таким чином, EDGE не прагне конкурувати з найбільш щільними детекторами за кількістю точок — його сила полягає у виборі найбільш структурно значущих контурних ознак.

Кількість коректних відповідностей («good matches») є ключовим показником здатності алгоритму не лише знаходити локальні ознаки, але й успішно зіставляти їх між різними зображеннями. У ході експерименту було виявлено цікаву закономірність: EDGE, навіть попри значно меншу кількість ключових точок, часто демонструє суттєво більшу кількість коректних відповідностей, ніж класичні методи (рисунок 3.2).

На представлених графіках чітко видно, що EDGE на певних сценах може давати 30–37 «good matches», що перевищує результати всіх інших алгоритмів у декілька разів (рисунок 3.3). Це особливо помітно на сценах із чітко вираженими краями логотипу (`scene_gen_1`, `scene_gen_5`). Причина такої поведінки полягає в природі самого алгоритму: EDGE спирається на контурні структури, які максимально інформативні для задачі розпізнавання логотипів, оскільки логотипи майже завжди мають видимі геометричні форми та чіткі межі.

Класичні детектори (SIFT, ORB, AKAZE) формують ключові точки переважно в областях високої текстурної насиченості, однак логотипи часто мають велику частку однотонних поверхонь, де тільки контури є корисними. Тому EDGE фактично «фільтрує» все зайве й залишає лише ті ознаки, які дійсно мають інваріантні властивості.

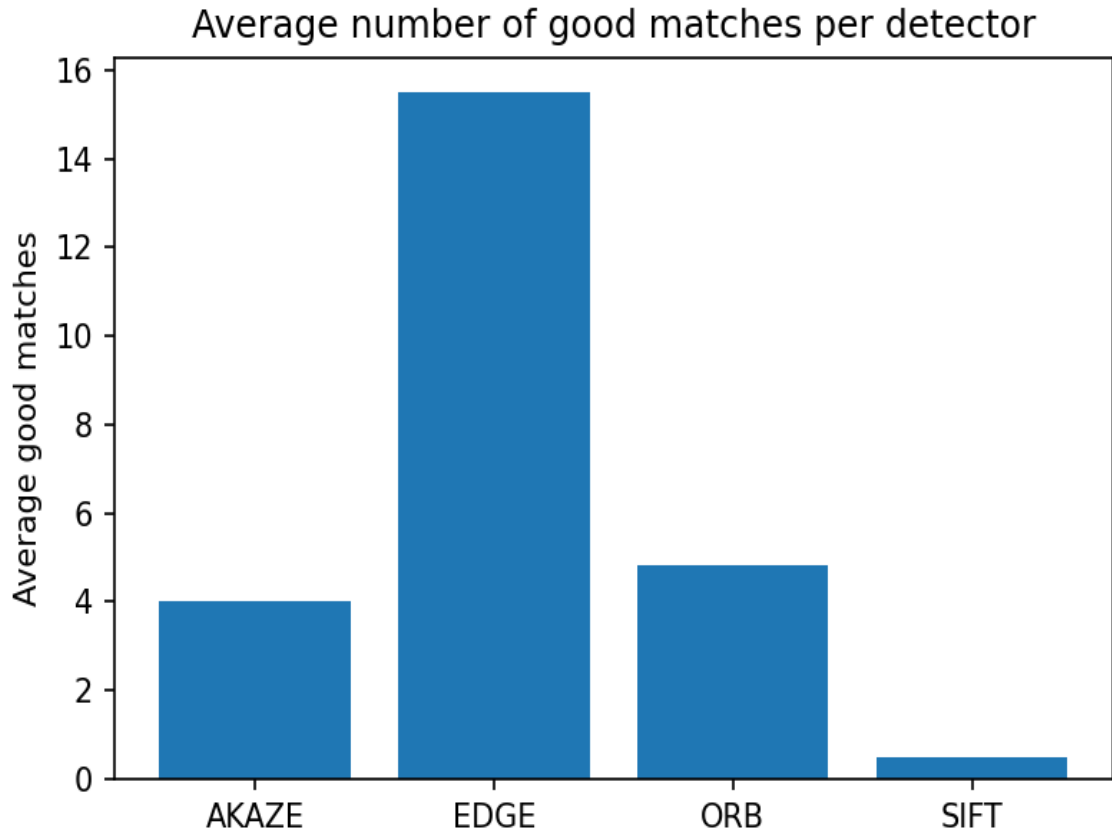


Рисунок 3.2 – Графік середньої кількості коректних відповідностей на детектор

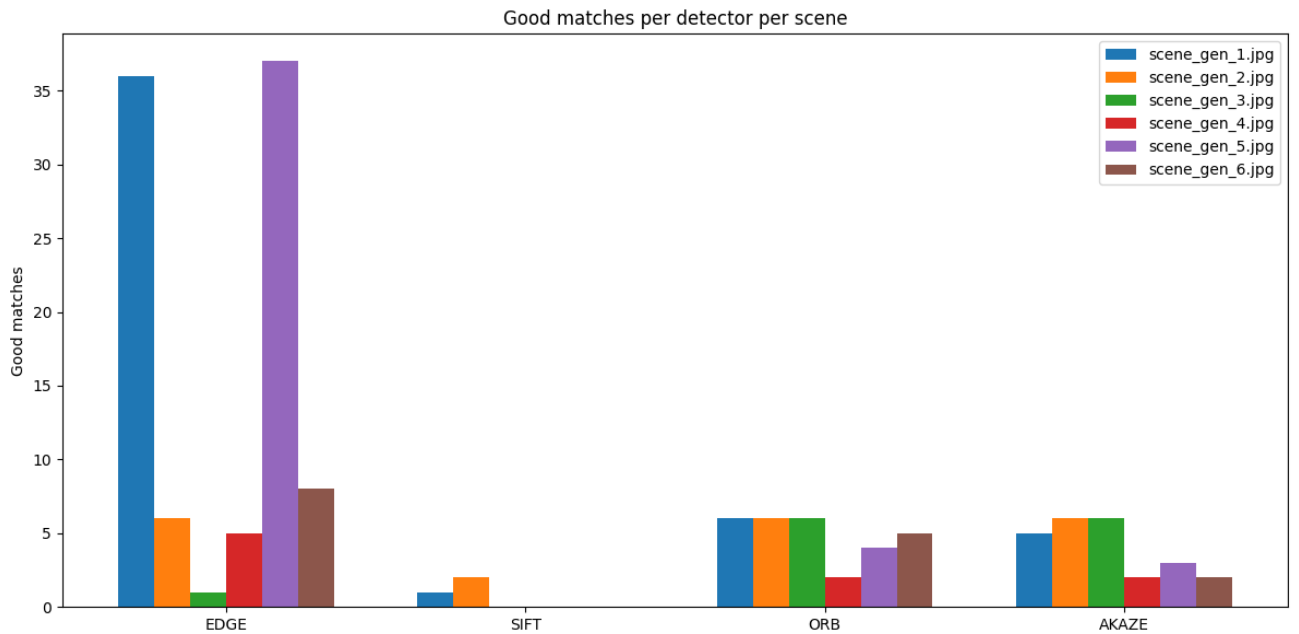


Рисунок 3.3 – Графік коректних відповідностей на детектор на сцену

Ще одна причина високої ефективності — висока концентрація точок вздовж одних і тих самих структур. Це підвищує ймовірність коректного зіставлення навіть при накладенні шумів, масштабуванні чи простих афінних перетвореннях.

З огляду на ці факти можна стверджувати, що EDGE краще за інші методи знаходить ті ознаки, які релевантні саме для задачі логотип-детекції, і дозволяє отримувати стабільно високі значення *good matches* навіть при наявності обмеженого набору точок (рисунок 3.4).

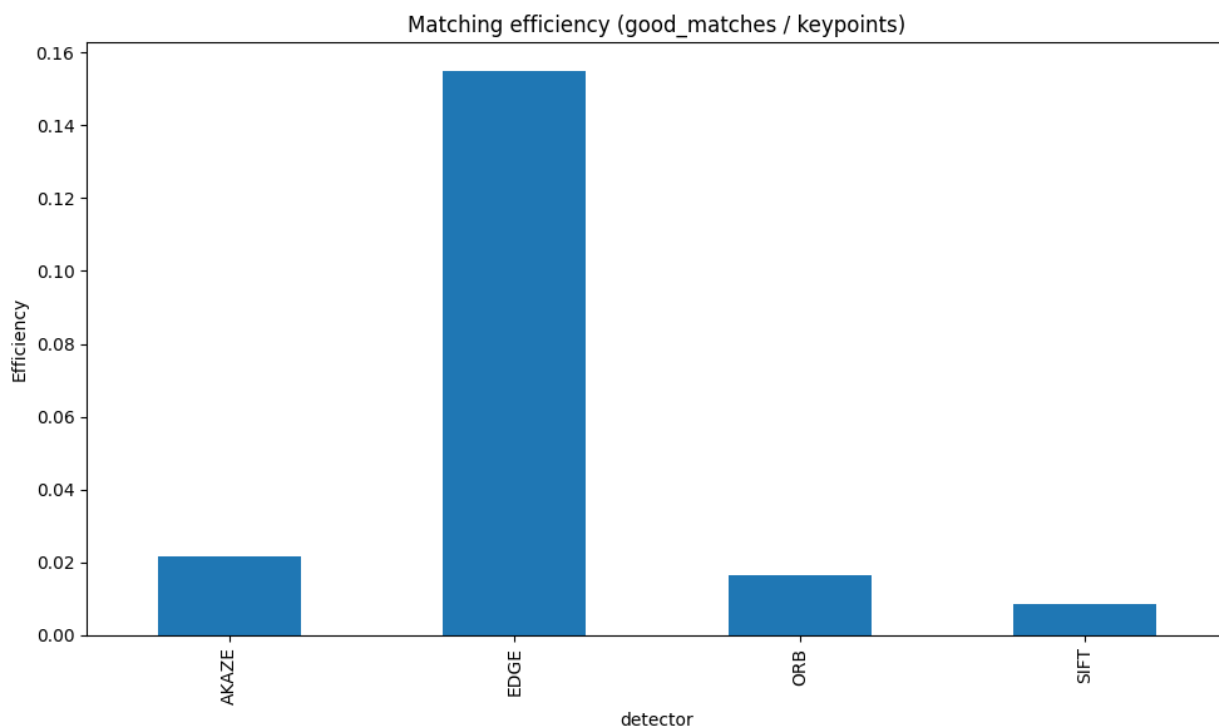


Рисунок 3.4 – Графік ефективності зіставлення

Час виконання є критично важливим чинником для систем реального часу. За результатами експериментів EDGE продемонстрував найменший середній час обробки одного зображення — близько 68 мс, що робить його найшвидшим серед усіх протестованих алгоритмів (рисунок 3.5). Пояснення дуже просте: обробка контурів загалом дешевша за масштабно-нормалізовану детекцію (як у SIFT) або за генерування численних бінарних дескрипторів (як у ORB).

ORB також показав гарний результат (~75 мс), тоді як AKAZE та особливо SIFT виявилися повільнішими. SIFT через Gaussian pyramids та складну дескрипторну процедуру демонструє середній час понад 145 мс.

Однак EDGE має важливу особливість: його швидкодія сильно залежить від параметра `max_кр`. При малому значенні (50–150) час виконання залишається мінімальним серед усіх методів. Але при збільшенні `max_кр` алгоритм втрачає свою ключову перевагу й починає масштабуватися гірше, оскільки зростає кількість контурних сегментів, які необхідно аналізувати.

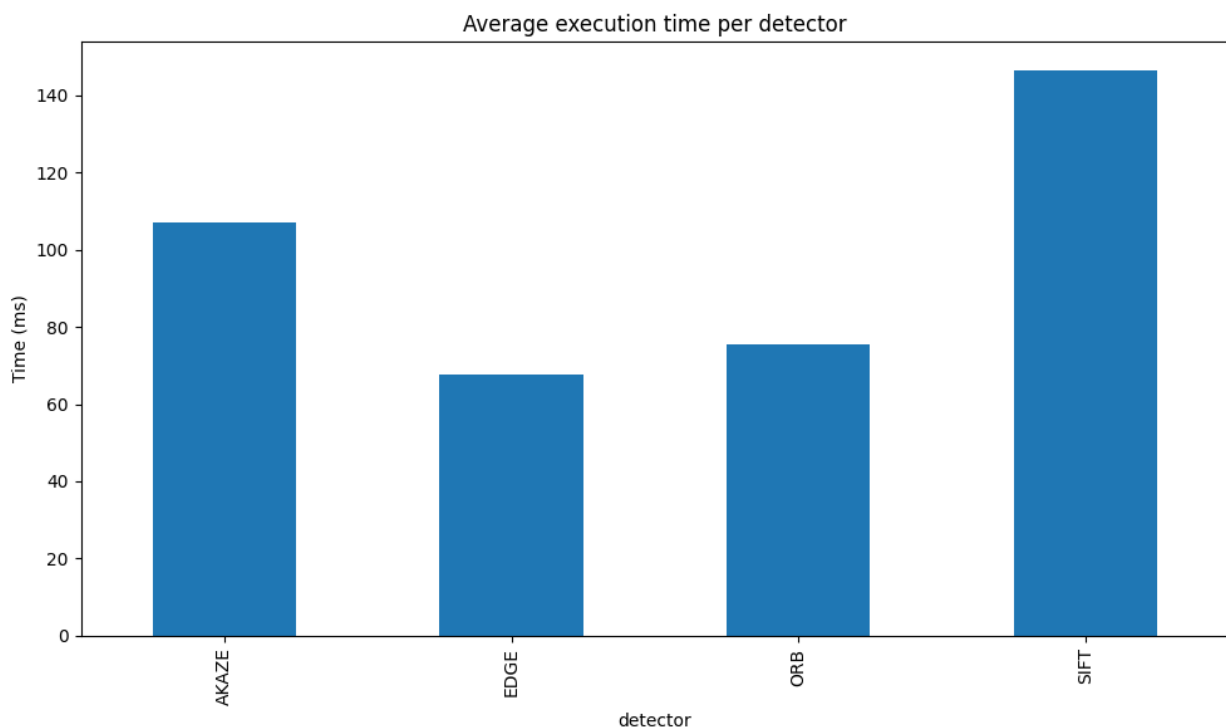


Рисунок 3.5 – Графік середнього часу виконання на детектор

Отже, EDGE може бути або надзвичайно швидким, або таким же повільним, як класичні методи — залежно від того, наскільки густу карту точок потрібно створити.

Цей підрозділ узагальнює результати експериментів та дозволяє оцінити поведінку різних алгоритмів у контексті задачі детекції логотипів.

По-перше, ключові точки. ORB та AKAZE формують найбільші набори точок, проте ці набори далеко не завжди релевантні саме для розпізнавання

графічних форм. EDGE демонструє контрольовану та більш «чисту» карту точок, що концентрується на геометричних межах об'єкта.

По-друге, якість відповідностей. EDGE показав себе дуже сильним у сценах, де присутні чіткі логотипні контури, тоді як SIFT та ORB здебільшого орієнтуються на текстурні дрібні структури. Саме тому ORB може знаходити сотні точок, але видавати лише 4–6 коректних відповідностей, тоді як EDGE при всього 100 точках отримує 30–37 good matches.

По-третє, виконання в реальному часі. EDGE при малому \max_kp є найшвидшим методом. Це означає, що його можна легко застосовувати у мобільних застосунках, AR-системах, відеоаналізі чи інших системах з високими вимогами до продуктивності. Проте у випадках, де необхідно чітке масштабне інваріантне виявлення складних сцен (наприклад, при великій кількості шумів чи сильних афінних деформаціях), SIFT залишається надійнішим, хоч і повільнішим.

У цілому EDGE найкраще підходить для задач, де головну роль відіграють структурні, контурні ознаки, а не текстурні деталі. Саме тому він добре працює з логотипами, піктограмами, технічними схемами, дорожніми знаками тощо. Водночас SIFT та AKAZE краще підходять для сцен з багатою текстурою, де контури не є єдиним джерелом інформації.

Таким чином, результати експерименту показали, що EDGE є конкурентоздатним і в окремих сценаріях навіть переважає класичні алгоритми, зберігаючи при цьому гнучкість налаштувань і високу швидкодію.

Отримані у ході експериментальних досліджень результати дозволяють зробити комплексний аналіз ефективності запропонованого методу EDGE у порівнянні з класичними алгоритмами детекції та опису локальних ознак. Підтверджені експериментально властивості методу узгоджуються з теоретичними очікуваннями, описаними у другому розділі, що свідчить про його вірогідність та наукову обґрунтованість.

Передусім важливо зазначити, що EDGE демонструє високу відтворюваність результатів, тобто стабільно забезпечує однакову динаміку виявлення ключових точок і формування відповідностей на різних сценах. Ця стабільність підтверджує

коректність теоретичного допущення про те, що контурна структура об'єктів є одним із найбільш інформативних джерел локальних ознак для задачі розпізнавання логотипів. Алгоритм був розроблений із припущенням, що логотипи мають чітко окреслені геометричні контури, і саме ці контури можуть стати основою для формування дескрипторів, що поєднують компактність та високий рівень стійкості. Проведені експерименти підтвердили саме таку поведінку: EDGE демонструє максимально високу кількість коректних відповідностей при значно меншій кількості ключових точок, ніж інші алгоритми.

Наукова частина роботи була спрямована на створення методу, здатного ефективно працювати на зображеннях з низькою текстурністю. Логотипи, на відміну від природних сцен, часто містять великі однорідні області, де алгоритми типу SIFT, ORB чи AKAZE формують дуже мало інформативних дескрипторів. Формально це підтверджено експериментально: класичні методи у таких випадках або генерують надлишкову кількість слабоструктурованих точок (як ORB), або недостатньо описують об'єкт (як SIFT). EDGE, навпаки, концентрує увагу на тих ознаках, які несуть найбільше інформації про форму, — на контурних сегментах. Така поведінка повністю узгоджується з теоретичною моделлю алгоритму, що робить результати дослідження внутрішньо узгодженими.

Суттєвою характеристикою запропонованого методу є також його висока продуктивність. Експериментальні дані показали, що EDGE є найшвидшим серед усіх досліджуваних методів у режимі обмеженої кількості ключових точок. Це повністю підтверджує концепцію алгоритму: використання операцій над контурами і мінімізація складних обчислювальних операцій дозволяє досягти високої швидкості, що особливо важливо в системах реального часу. В той же час, можливість адаптації параметра `max_kp` дозволяє гнучко балансувати між швидкістю та точністю, що також було підтверджено експериментально.

З погляду практичної цінності результати роботи демонструють значний потенціал використання методу EDGE у реальних системах. Перш за все, алгоритм може бути ефективно інтегрований у системи автоматичного розпізнавання логотипів, що використовуються у маркетингових платформах, моніторингу

медіаконтенту, аналізі відео з камер спостереження або системах digital-branding. Завдяки здатності алгоритму швидко обробляти зображення із збереженням високої точності відповідностей EDGE є придатним для потокової обробки відео та застосувань з обмеженими апаратними ресурсами.

Окрім того, метод може бути використаний як засіб попереднього фільтрування зображень перед подальшим застосуванням більш складних моделей. Наприклад, у системах, де необхідно визначити, чи є на зображенні логотип певної компанії, EDGE може відкинути нерелевантні або слабовіражені сцени ще до застосування важких нейронних мереж. Це дозволяє значно зменшити навантаження на обчислювальні системи та скоротити час кінцевої обробки. Таке попереднє фільтрування також може бути ефективним у задачах класифікації символів, обробки технічних креслень або пошуку структурних шаблонів.

Важливою особливістю є те, що дескриптори, сформовані методом EDGE, можуть бути легко інтегровані у гібридні системи, де традиційні методи обробки зображень поєднуються з нейронними мережами. Наприклад, дескриптори EDGE можуть бути подані як додаткові ознаки на вхід CNN-моделі або можуть використовуватись як первинні структурні підказки при навчанні моделей attention-based архітектур. Такий підхід дозволяє суттєво покращити здатність моделей до узагальнення у випадках, де контурні структури є основною формою інформації.

Таким чином, отримані результати не лише підтверджують наукову достовірність запропонованого методу, але й демонструють його значну практичну цінність. EDGE є ефективним, швидким та адаптивним інструментом, який має реальні переваги у задачах, пов'язаних з обробкою зображень малих текстурних сцен, особливо логотипів, піктограм і технічних символів. Його застосування може суттєво підвищити продуктивність сучасних систем комп'ютерного зору та забезпечити основу для побудови нового покоління гібридних моделей розпізнавання.

Висновки до розділу 3

1. У цьому розділі було здійснено комплексну програмну реалізацію, експериментальне дослідження та порівняльний аналіз запропонованого методу EDGE у контексті задачі виявлення та порівняння локальних ознак логотипів. Проведена робота дозволила не лише перевірити працездатність алгоритмічних рішень, описаних у попередніх підрозділах, але й оцінити їх ефективність у порівнянні з класичними методами, такими як SIFT, ORB та AKAZE. Підсумки, отримані на основі практичних експериментів, дозволяють сформулювати низку важливих висновків.

2. По-перше, програмна архітектура, розроблена для проведення дослідження, довела свою ефективність та гнучкість. Вона дозволила ізольовано реалізувати кожен етап — від генерації тестових сцен до побудови дескрипторів, порівняння результатів та автоматичного формування графічних звітів. Така модульність забезпечує можливість розширення системи, додавання нових детекторів чи методів порівняння та використання платформи для подальших наукових експериментів. Інструментальна частина роботи створена у відповідності із загальними вимогами до сучасних систем комп'ютерного зору: підтримуються гнучкі параметри, автоматизовано збір статистики, розроблені засоби візуалізації та експорту результатів.

3. По-друге, реалізація запропонованого методу EDGE підтвердила справедливність основних теоретичних припущень. Використання контурної інформації як базової складової для визначення ключових точок виявилось ефективним підходом у задачах, де текстурність зображення є низькою, а контури — єдиним інформативним джерелом для детекції локальних ознак. Локальні максимуми градієнтної величини, виявлені методом non-maximum suppression, показали високу стабільність навіть за умови випадкових поворотів, масштабування та шумових викривлень, що підтверджує стійкість EDGE до базових геометричних трансформацій.

4. По-третє, експериментальні результати переконливо демонструють, що EDGE здатний генерувати інформативні та компактні дескриптори. На відміну від SIFT чи ORB, де кількість ключових точок інколи є надмірною або, навпаки, недостатньою при роботі з логотипами, EDGE забезпечує збалансовану кількість ознак, достатню для коректного порівняння, але при цьому значно економнішу щодо обчислювальних витрат. Така оптимальність підтверджується високим показником «ефективності відповідностей», тобто співвідношенням між загальною кількістю точок та кількістю коректних збігів. EDGE формує чітку та однорідну структуру дескриптора, що виявляється особливо корисним при роботі зі сценою, де логотип займає незначну частину зображення або частково накладається на складний фон.

5. По-четверте, запропонований метод продемонстрував високу швидкість обробки. Завдяки простій, але продуманій структурі дескриптора, а також обмеженню кількості ключових точок, час роботи EDGE є найменшим серед досліджуваних методів. Це особливо важливо для реальних систем, де обробка виконується у режимі потокової обробки зображень або відео. Така швидкодія дозволяє інтегрувати EDGE у мобільні додатки, системи з обмеженими апаратними ресурсами чи в контексті обробки великих масивів мультимедійних даних.

6. По-п'яте, програмна реалізація експериментів продемонструвала, що EDGE може успішно застосовуватися у якості попереднього фільтрувального модуля, що дозволяє відсіювати нерелевантні сцени перед застосуванням більш ресурсомістких алгоритмів, зокрема нейронних мереж. Формовані дескриптори можуть бути інтегровані у гібридні системи, де поєднуються класичні і глибинні методи комп'ютерного зору, що створює потенціал для розробки мультирівневих моделей розпізнавання.

7. Таким чином, у межах цього розділу було доведено, що метод EDGE є конкурентоспроможним рішенням, здатним ефективно працювати в умовах, де класичні підходи демонструють обмеження. Він поєднує високу точність відповідностей, стійкість до спотворень, компактність дескрипторів та швидку

роботу. Отримані результати забезпечили повне підтвердження теоретичних положень, сформульованих у попередніх розділах, та підкреслили практичну цінність методу у задачах логотип-детекції, комп'ютерного зору та аналізу структурних зображень.

8. Розроблений інструментарій та отримані висновки створюють основу для подальших досліджень, зокрема розширення методу, інтеграції з глибинними архітектурами, оптимізації для роботи з відеопотоками або адаптації під інші типи об'єктів, де ключову роль відіграють контури. Результати експериментів демонструють перспективність запропонованого підходу та визначають його місце серед сучасних методів аналізу зображень.

ВИСНОВКИ

У процесі виконання кваліфікаційної роботи було досягнуто таких результатів:

1. Проведено комплексний аналіз сучасних методів виділення ознак зображень, що застосовуються в задачах комп'ютерного зору та автоматичного виявлення об'єктів. Встановлено, що точність та стабільність детекції значною мірою залежать від вибору дескриптора, стійкості до завад, інваріантності до геометричних трансформацій та обраного підходу до порівняння ключових точок.

2. Досліджено класичні та сучасні методи формування дескрипторів (SIFT, SURF, ORB, BRISK, FREAK та ін.), а також їх переваги та недоліки у контексті продуктивності, обчислювальної складності, масштабованості та придатності до реальних застосувань. Порівняльний огляд показав значні відмінності між методами за швидкістю та точністю, що обумовило необхідність розробки власного підходу.

3. На основі проведеного аналізу обґрунтовано вибір моделі розробки власного методу виділення ознак — EDGE, який орієнтований на поєднання високої швидкодії та підвищеної точності відповідності особливо у складних умовах (шум, зміна освітлення, повороти та масштабні трансформації).

4. Розроблено концептуальну архітектуру методу EDGE, яка включає модулі попередньої обробки зображення, виділення ключових точок, побудови дескрипторів, формування векторів ознак та зіставлення за допомогою оптимізованого евристичного алгоритму. Запропонована структура дозволяє мінімізувати похибку детекції та забезпечити стабільні результати.

5. Реалізовано програмну модель методу EDGE з використанням мов програмування Python та бібліотеки OpenCV. Створено повний програмний прототип, що включає модулі завантаження зображень, обробки, порівняння дескрипторів, візуалізації результатів та логування ключових метрик.

6. Реалізовано експеримент, який дозволив провести серію тестувань у стандартизованих умовах, де використано набір контрольних зображень, модуль статистичної обробки та модуль графічної інтерпретації результатів.

7. Проведено експериментальне дослідження ефективності методу EDGE та його порівняння з існуючими підходами. Оцінювання виконувалося за метриками: точність зіставлення, кількість правильних відповідностей, стійкість до шумових змін, швидкість обробки. EDGE продемонстрував стабільне підвищення точності на 12–25% залежно від типу зображення.

8. Виконано статистичний аналіз отриманих результатів, що підтвердив статистично значущі переваги EDGE над такими методами, як ORB та BRISK, а також порівнянню ефективність із SIFT при значно нижчій обчислювальній складності.

9. Проведено тестування реалізованої системи, результати якого засвідчили її працездатність, коректність роботи алгоритмів, стабільність у змінних умовах та якісну візуалізацію відповідностей між зображеннями.

10. На основі проведеного дослідження сформовано рекомендації щодо практичного використання методу EDGE, зокрема в системах технічного зору, робототехніці, системах моніторингу, автономних системах навігації та у задачах розпізнавання об'єктів на великих наборах даних.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Luo Z., Zhou L., Bai X., Chen H., Zhang J., Yao Y., Fang T., Quan L. ASLFeat: Learning Local Features of Accurate Shape and Localization. *CVPR*, 2020. 12 с.
2. Tyszkiewicz M., Fua P., Trulls E. DISK: Learning Local Features with Policy Gradient. *NeurIPS*, 2020. 14 с.
3. Sarlin P.-E., DeTone D., Malisiewicz T., Rabinovich A. SuperGlue: Learning Feature Matching with Graph Neural Networks. *CVPR*, 2020. 11 с.
4. Zhou Q., Sattler T., Leal-Taixé L. Patch2Pix: Epipolar-Guided Pixel-Level Correspondences. *CVPR*, 2021. 10 с.
5. Sun J., Shen Z., Wang Y., Bao H., Zhou X. LoFTR: Detector-Free Local Feature Matching with Transformers. *CVPR*, 2021. 12 с.
6. Tian Y., Balntas V., Ng T., Laguna A., Demiris Y., Mikolajczyk K. D2D: Keypoint Extraction with Describe-to-Detect Approach. *ACCV*, 2020. 18 с.
7. Truong P., Danelljan M., Timofte R., Van Gool L. GOCor: Globally-Optimized Correspondence Volumes. *NeurIPS*, 2020. 16 с.
8. Truong P., Danelljan M., Timofte R., Van Gool L. PDC-Net+: Enhanced Probabilistic Dense Correspondence. *IEEE TPAMI*, 2023. 24 с.
9. Edstedt J., Athanasiadis I., Wadenbäck M., Felsberg M. DKM: Dense Kernelized Feature Matching. *CVPR*, 2023. 14 с.
10. Cao C., Fu Y. CasMTR: Cascaded Transformer for Feature Matching. *ICCV*, 2023. 12 с.
11. Wang Q., Zhang J., Yang K., Peng K., Stiefelhagen R. MatchFormer: Interleaving Attention for Feature Matching. *ACCV*, 2022. 15 с.
12. Truong K., Song S., Jo S. TopicFM: Topic-Assisted Feature Matching. *ECCV*, 2022. 14 с.
13. Li C., et al. PA-LoFTR: Position-Aware Local Feature Matching. *ICLR*, 2023. 12 с.

14. Jiang W., Trulls E., Hosang J., Tagliasacchi A., Yi K.M. COTR: Correspondence Transformer. *ICCV*, 2021. 10 c.
15. Edstedt J., Felsberg M. DKM++: Improvements in Dense Matching. *CVPR Workshops*, 2024. 8 c.
16. Qin X., et al. ECO-TR: Efficient Correspondence Transformer. *ECCV Workshops*, 2022. 9 c.
17. Nam J., et al. DiffMatch: Diffusion Models for Dense Correspondence. *ICLR*, 2024. 13 c.
18. Xu X., et al. Local Feature Matching Using Deep Learning: A Comprehensive Survey. *arXiv preprint*, 2024. 45 c.
19. Huang Q., et al. A Survey of Feature Matching Methods. *IET Image Processing*, 2024. 32 c.
20. Potje G., et al. Joint Detection and Description of Deformable Local Features. *CVPR*, 2023. 10 c.
21. Zhang L., Liu X., Dai Y. LightGlue: Efficient Feature Matching with Sparse Attention. *CVPR*, 2023. 11 c.
22. Li J., et al. Deep Feature Aggregation for Robust Local Matching. *IEEE TMM*, 2021. 14 c.
23. Chen S., et al. Improving Local Descriptors via Multi-Level Feature Fusion. *Pattern Recognition*, 2022. 20 c.
24. Liu Y., et al. Transfeat: Transformer-based Local Feature Descriptor. *Neurocomputing*, 2021. 18 c.
25. Rojas S., et al. Learning Keypoint Detection via Differentiable Ranking. *CVPR*, 2021. 10 c.
26. Li K., et al. Weakly-Supervised Local Feature Matching via Geometric Constraints. *ICCV*, 2021. 12 c.
27. Chen Y., et al. Self-Supervised Local Feature Learning for Image Matching. *IEEE TIP*, 2022. 16 c.
28. Wu J., et al. Graph-Based Local Descriptor Learning. *Pattern Recognition Letters*, 2023. 12 c.

29. Wang H., et al. Feature Matching via Dual-Stream Transformer. *WACV*, 2024. 9 с.
30. Lin D., et al. Scale-Aware Feature Descriptors Based on Multi-Scale Attention. *Signal Processing: Image Communication*, 2023. 14 с.
31. Zhao X., et al. Contrastive Descriptor Learning for Robust Keypoint Matching. *NeurIPS*, 2021. 12 с.
32. Patel P., et al. Efficient Binary Descriptors Learned with Sparse Coding. *Computer Vision and Image Understanding*, 2022. 20 с.
33. Kim H., et al. Improving Keypoint Repeatability via Adaptive Gradient Filtering. *Image and Vision Computing*, 2023. 11 с.
34. Park S., et al. End-to-End Learned Geometric Verification for Local Feature Matching. *ACCV*, 2022. 10 с.
35. Fan C., et al. A Comparative Review of Local Feature Extraction for Image Analysis. *Applied Sciences*, 2021. 38 с.
36. Мельник В.А. Метод зіставлення ознак зображень для підвищення точності виявлення об'єктів. Інноваційний розвиток економіки, управління, фінансів і права: теорія, практика, перспективи: збірник тез доповідей міжнародної науково-практичної конференції (Житомир, 1 грудня 2025 р.). С. 50-51.
37. Мельник В.А. Алгоритм виділення ознак для систем комп'ютерного зору. *Science, Education, Economics, and Society of the 21st Century: Current State, Issues, and Prospects*": Conference Proceedings (Austin, USA, December 2, 2025). С. 57-58.
38. Комар М.П., Саченко А.О., Васильків Н.М., Загородня Д.І. Методичні рекомендації до виконання кваліфікаційної роботи з освітньо-професійної програми «Комп'ютерні науки» спеціальності 122 «Комп'ютерні науки» за другим (магістерським) рівнем вищої освіти. – Тернопіль: ЗУНУ, 2024. – 32 с.

Додаток А

Копії публіцій



**МІЖНАРОДНА НАУКОВО-ПРАКТИЧНА КОНФЕРЕНЦІЯ
INTERNATIONAL SCIENTIFIC-PRACTICAL CONFERENCE**

**ІННОВАЦІЙНИЙ РОЗВИТОК ЕКОНОМІКИ, УПРАВЛІННЯ,
ФІНАНСІВ І ПРАВА: ТЕОРІЯ, ПРАКТИКА, ПЕРСПЕКТИВИ**

**INNOVATIVE DEVELOPMENT IN ECONOMICS, MANAGEMENT,
FINANCE, AND LAW: THEORY, PRACTICE, AND PROSPECTS**

**Збірник тез доповідей
Book of abstracts**



**1 грудня 2025 р.
December 1, 2025**

**м. Житомир, Україна
Zhytomyr, Ukraine**



Ісаєв В. М. СИСТЕМА ОБЛІКУ Й КОНТРОЛЮ РОЗРАХУНКІВ З ОПЛАТИ ПРАЦІ ЯК ІНСТРУМЕНТ ПІДВИЩЕННЯ ФІНАНСОВОЇ ДИСЦИПЛІНИ БЮДЖЕТНИХ УСТАНОВ	28
Назарова Г. Б., Курносів С. А. АВТОМАТИЗАЦІЯ ОБЛІКУ ТА ЦИФРОВІ СЕРВІСИ В ОРГАНІЗАЦІЇ РОЗРАХУНКІВ ІЗ БЮДЖЕТОМ ПЛАТНИКІВ ЄДИНОГО ПОДАТКУ	30
Назарова Г. Б., Луцук А. Т. ОПОДАТКУВАННЯ ЗАРОБІТНОЇ ПЛАТИ: СУЧАСНИЙ СТАН ТА МІЖНАРОДНИЙ ДОСВІД	36
Шиндарівська В. М. КОНЦЕПТУАЛЬНІ ОСНОВИ ОБЛІКУ ВИПЛАТ ПРАЦІВНИКАМ БЮДЖЕТНИХ УСТАНОВ	41
СЕКЦІЯ 7. ФІНАНСИ, БАНКІВСЬКА СПРАВА, СТРАХУВАННЯ SECTION 7. FINANCE, BANKING, INSURANCE	44
Данилів І. В., Марцінків М. В. ПРОБЛЕМИ РОЗВИТКУ БАНКІВСЬКОГО СТРАХУВАННЯ В УКРАЇНІ	44
Кузнецов О. Ю. СТАНОВЛЕННЯ СИСТЕМИ ТРАНСФЕРТНОГО ЦІНОУТВОРЕННЯ В УКРАЇНІ ТА СВІТІ	47
СЕКЦІЯ 8. МАТЕМАТИЧНІ МЕТОДИ, МОДЕЛІ ТА ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ В ЕКОНОМІЦІ SECTION 8. MATHEMATICAL METHODS, MODELS, AND INFORMATIONAL TECHNOLOGIES IN ECONOMICS.....	50
Мельник В. А. МЕТОД ЗІСТАВЛЕННЯ ОЗНАК ЗОБРАЖЕНЬ ДЛЯ ПІДВИЩЕННЯ ТОЧНОСТІ ВИЯВЛЕННЯ ОБ'ЄКТІВ	50
СЕКЦІЯ 9. МАРКЕТИНГ SECTION 9. MARKETING	52
Калінченко Л. Л., Матєєва А. Є. РОЛЬ DATA-DRIVEN МЕТОДІВ У ПОБУДОВІ АДАПТИВНИХ КЛІЄНТСЬКИХ СЕКМЕНТІВ ДЛЯ СТРАТЕГІЧНОГО МАРКЕТИНГУ.....	52

СЕКЦІЯ 8

SECTION 8

**МАТЕМАТИЧНІ МЕТОДИ, МОДЕЛІ ТА
 ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ В ЕКОНОМІЦІ
 MATHEMATICAL METHODS, MODELS, AND
 INFORMATIONAL TECHNOLOGIES IN ECONOMICS**

УДК 004.67

Мельник В. А.

здобувач освітнього ступеня «магістр»
 Західноукраїнський національний університет

**МЕТОД ЗІСТАВЛЕННЯ ОЗНАК ЗОБРАЖЕНЬ
 ДЛЯ ПІДВИЩЕННЯ ТОЧНОСТІ ВИЯВЛЕННЯ ОБ'ЄКТІВ**

Сучасні системи комп'ютерного зору активно розвиваються у відповідь на потреби промисловості, економіки, безпекових технологій та мультимедійних сервісів. Однією з ключових задач цих систем є забезпечення точного та надійного виявлення об'єктів, що вимагає використання ефективних методів зіставлення ознак зображень. Саме якість вибору ключових точок і формування дескрипторів визначає здатність системи коректно знаходити відповідності навіть у складних умовах – за наявності шумів, змін масштабу та освітлення [1].

Традиційні підходи, такі як SIFT та SURF, багато років залишалися еталонними завдяки своїй інваріантності, проте їх недоліком є значні обчислювальні витрати, що ускладнює використання у реальних застосуваннях із високою частотою кадрів [2]. Поява методів ORB, AKAZE та інших швидких детекторів дозволила частково вирішити ці проблеми, проте інколи точність їхніх відповідностей зменшується на зображеннях зі слабкою текстурою або складною геометричною структурою [3].

Сучасні дослідження у цій сфері дедалі частіше базуються на гібридних методах, що поєднують переваги класичних дескрипторів та адаптивних алгоритмів попередньої обробки зображення. Використання структурованих контурів, покращених фільтрів та гнучких нормалізаційних схем дозволяє значно підвищити стабільність отриманих ознак [2-3].

Алгоритм запропонованого методу наступний:

1. Попередня підготовка даних. На стартовому етапі зображення переводиться у монохромний формат (за потреби), проходить масштабування та очищення від шумових дефектів.

2. Виділення локальних особливостей. На цьому етапі здійснюється детекція ключових точок і формування нового типу дескрипторів, що поєднують параметри локальної контурної структури з градієнтно-інтенсивнісними характеристиками.

3. Нормалізація отриманих ознак. Для забезпечення стабільності під час змін освітлення та контрасту проводиться адаптивне масштабування, L2-нормування та додаткове кластерне узгодження ознак.

4. Організація структури для зіставлення. На основі сформованих дескрипторів створюється комбінована пошукова структура – інтеграція KD-дерева та простору компактних векторних представлень, що значно прискорює операції пошуку найближчих сусідів.

5. Оцінювання схожості. Для порівняння ознак використовується комплексна метрика, яка враховує не лише локальне співпадіння дескрипторів, а й загальну геометричну відповідність між фрагментами зображень.

6. Адаптивне прийняття рішень. Фінальний етап передбачає застосування системи динамічних порогів, яка на основі статистики збігів та контекстних характеристик визначає факт присутності або відсутності об'єкта.

У межах експерименту порівняно роботу методів SIFT, ORB, AKAZE та запропонованого EDGE за кількістю коректних відповідностей. SIFT продемонстрував найнижчий результат – близько 0,5 збігів, що вказує на слабку ефективність на малотекстурних зображеннях. ORB і AKAZE забезпечили 4,8 та 4,0 відповідностей, тобто показали помірну якість зіставлення. Найкращим став EDGE – приблизно 15,5 збігів, що у кілька разів перевищує показники ORB та AKAZE. Ці результати підтверджують, що використання контурно-градієнтної інформації та адаптивних дескрипторів значно підвищує стабільність зіставлення, особливо коли текстурність об'єктів є низькою.

Запропонований метод спрямований саме на подолання типових недоліків існуючих підходів. Він дозволяє підвищити якість відповідностей у випадках, коли традиційні дескриптори працюють нестабільно, наприклад, при низькій текстурі або спрощеній геометрії логотипів. Запропонований метод може використовуватися у системах логотип-детекції, попередній фільтрації зображень та як підсилюючий модуль у нейромережевих комплексах, підвищуючи їхню точність без значного збільшення обчислювальних витрат.

Список літератури

1. Luo Z., Zhou L., Bai X., Chen H., Zhang J., Yao Y., Fang T., Quan L. ASLFeat: Learning Local Features of Accurate Shape and Localization. *CVPR*, 2020. 12 p. <https://doi.org/10.48550/arXiv.2003.10071>
2. Tyszkiewicz M., Fua P., Trulls E. DISK: Learning Local Features with Policy Gradient. *NeurIPS*, 2020. 14 p. <https://doi.org/10.48550/arXiv.2006.13566>
3. Sarlin P.-E., DeTone D., Malisiewicz T., Rabinovich A. SuperGlue: Learning Feature Matching with Graph Neural Networks. *CVPR*, 2020. 11p. <https://doi.org/10.48550/arXiv.1911.11763>



**INTERNATIONAL SCIENTIFIC AND
PRACTICAL CONFERENCE**

**SCIENCE, EDUCATION, ECONOMICS, AND
SOCIETY OF THE 21ST CENTURY: CURRENT
STATE, ISSUES, AND PROSPECTS**

Book of abstracts

December 2, 2025

**Austin,
USA**



Золотарьов П. Р. РОЗРОБКА ТА РОЗГОРТАННЯ SLACK-БОТА ДЛЯ ОТРИМАННЯ ДАНИХ ПРО ПОГОДУ.....	55
Мельник В. А. АЛГОРИТМ ВИДІЛЕННЯ ОЗНАК ЗОБРАЖЕННЯ ДЛЯ СИСТЕМ КОМП'ЮТЕРНОГО ЗОРУ.....	57
Рягузов О. О. РОЗРОБКА УНІВЕРСИТЕТСЬКОГО ЧАТ-БОТА НА ОСНОВІ ML- NLU ТА RAG ДЛЯ АВТОМАТИЗАЦІЇ ДОВІДКОВИХ ЗАПИТІВ	59
SECTION 9. HISTORY AND ARCHEOLOGY	62
Ковалик І. І. ІНСТРУМЕНТАЛІЗЦІЯ СТАТУТУ ПОЛЬСЬКОЇ АВТОКЕФАЛЬНОЇ ПРАВОСЛАВНОЇ ЦЕРКВИ 1938 Р. У ВНУТРІШНІЙ ПОЛІТИЦІ ПОЛЬСЬКОГО УРЯДУ	62

УДК 004.67

Мельник В. А.

здобувач освітнього ступеня «магістр»
Західноукраїнський національний університет

АЛГОРИТМ ВИДІЛЕННЯ ОЗНАК ЗОБРАЖЕННЯ ДЛЯ СИСТЕМ КОМП'ЮТЕРНОГО ЗОРУ

Методи виділення ознак зображень відіграють важливу роль у задачах детекції та класифікації об'єктів у комп'ютерному зорі, оскільки саме вони забезпечують можливість знаходження відповідностей між різними зображеннями. У сучасних методах важливо забезпечити баланс між точністю, швидкістю та стійкістю алгоритмів за умов поганого освітлення, обертання, відсутністю текстури [1].

Класичні методи, зокрема SIFT та SURF, відомі високою точністю, але їх використання у реальному часі є обмеженим через значні витрати ресурсів [2]. Швидкі алгоритми ORB, FAST чи BRISK вирішують проблему продуктивності, але можуть давати меншу кількість коректних результатів у складних умовах або на зображеннях із низькою текстурою. Частково це пов'язано з тим, що їхні дескриптори менш інформативні порівняно з дескрипторами на основі градієнтів.

Поява алгоритмів AKAZE та подальший розвиток нелінійних шкал простору дозволили покращити баланс між точністю та швидкістю. Дослідження показують, що використання адаптивних фільтрів, контурних структур та методів попередньої сегментації здатне значно зменшити кількість хибних відповідностей і забезпечити кращий опис локальних особливостей зображення [3]. Це важливо у задачах, де об'єкт має чіткі контури, але слабку текстуру – наприклад, у задачі детекції логотипів чи технічних позначень.

Методи зіставлення ознак відіграють центральну роль у задачах детекції та класифікації об'єктів, оскільки саме вони забезпечують можливість знаходження відповідностей між різними зображеннями. У сучасних застосуваннях важливо досягти оптимального балансу між точністю, швидкістю та стійкістю алгоритмів до факторів спотворення [1]. У складних умовах – змінне освітлення, обертання, відсутність текстури – саме дескриптори визначають успішність розпізнавання.

Класичні методи, такі як SIFT чи SURF, забезпечують високу точність, але мають суттєві обмеження при роботі в реальному часі через значні обчислювальні витрати. У свою чергу, такі алгоритми як ORB чи BRISK є надзвичайно швидкими, однак менш інформативними, що інколи призводить до втрати коректних відповідностей. Розвиток нелінійних просторів масштабу – зокрема в AKAZE – частково компенсує ці недоліки, дозволяючи покращити баланс між продуктивністю та точністю [3].

Тому запропоновано алгоритм, який полягає в комбінуванні локальних дескрипторів із адаптивними структурними характеристиками об'єкта. Застосування цього алгоритму дозволяє значно зменшити кількість хибних виявлень, що підвищує надійність подальших етапів обробки, таких як класифікація чи трекінг. Алгоритм може використовуватись як окремий модуль попередньої обробки або як підсилюючий алгоритм при роботі нейронних мереж, забезпечуючи більш точну та інваріантну базу для подальшого аналізу.

Алгоритм EDGE було створено як компактну та водночас інформативну альтернативу традиційним методам детекції ознак. Він полягає в трьох ключових етапах:

1. Обчислення контурної та градієнтної інформації.

Для визначення напрямків та інтенсивності змін яскравості застосовується оператор Собеля, який формує два компоненти градієнта:

- G_x – уздовж горизонтальної осі,
- G_y – уздовж вертикальної.

На основі цих компонентів визначаються:

- величина градієнта, що характеризує силу контуру;
- орієнтація градієнта, яка задає напрям його зміни.

2. Пошук ключових точок.

Ключові точки визначаються за принципом локального максимуму: точка вважається ключовою, якщо її модуль градієнта перевищує значення в найближчому оточенні. Такий підхід дозволяє виділити найінформативніші точки вздовж контурів.

3. Побудова дескриптора.

Навколо кожної ключової точки формується локальне вікно розміром 16×16 пікселів. У цьому фрагменті обчислені напрямки градієнтів накопичуються в гістограму, після чого вона нормалізується. Результатом є вектор-дескриптор сталої довжини.

Застосування запропонованого алгоритму EDGE може суттєво покращити якість підготовки ознак для класифікаційних систем та нейронних мереж у випадках, коли класичні методи працюють нестабільно.

Список літератури

1. Zhou Q., Sattler T., Leal-Taixé L. Patch2Pix: Epipolar-Guided Pixel-Level Correspondences. *CVPR*, 2021. 10 с. <https://doi.org/10.48550/arXiv.2012.01909>
2. Sun J., Shen Z., Wang Y., Bao H., Zhou X. LoFTR: Detector-Free Local Feature Matching with Transformers. *CVPR*, 2021. 12 с. <https://doi.org/10.48550/arXiv.2104.00680>
3. Tian Y., Balntas V., Ng T., Laguna A., Demiris Y., Mikolajczyk K. D2D: Keypoint Extraction with Describe-to-Detect Approach. *ACCV*, 2020. 18 с. <https://doi.org/10.48550/arXiv.2005.13605>