

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Інформаційно-аналітичних технологій та менеджменту
(повна назва)

Кафедра Інформатики
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти другий (магістерський)

ДОСЛІДЖЕННЯ МЕТОДУ КЛАСИФІКАЦІЇ ЗОБРАЖЕНЬ НА
ОСНОВІ ПАРАМЕТРІВ КВАНТУВАННЯ ОПИСУ У ПРОСТОРИ
БАГАТОВИМІРНИХ ДАНИХ
(тема)

Виконав:
здобувач 2 року навчання,
групи ІНФМ-24-1

Мазур Є. В.
(прізвище, ініціали)

Спеціальність 122 Комп'ютерні науки
(код і повна назва спеціальності)

Тип програми освітньо-професійна

Освітня програма Інформатика
(повна назва освітньої програми)

Науковий керівник проф. Гороховатський В. О.
(посада, прізвище, ініціали)

Допускається до захисту

Завідувач кафедри інформатики _____
(підпис)

Кобилін О. А.
(прізвище, ініціали)

2025 р.

Харківський національний університет радіоелектроніки

Факультет Інформаційно-аналітичних технологій та менеджменту

Кафедра Інформатики

Рівень вищої освіти другий (магістерський)

Спеціальність 122 Комп'ютерні науки
(код і повна назва)

Тип програми освітньо-професійна

Освітня програма Інформатика
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

« _____ » _____ 2025 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві Мазуру Єгору Володимировичу
(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження методу класифікації зображень на основі параметрів квантування опису у просторі багатомірних даних

затверджена наказом університету від 14 листопада 2025 року № 1045Ст

2. Термін подання здобувачем роботи до екзаменаційної комісії 24 листопада 2025 р.

3. Вихідні дані до роботи науково-технічна література, матеріали наукових конференцій, середовище розробки Visual Studio 2022, обгортка бібліотеки комп'ютерного зору OpenCV для мови C# OpenCvSharp.

4. Перелік питань, що потрібно опрацювати в роботі _____

1. Аналіз основних методів класифікації зображень. _____

2. Моделі класифікації зображень за множиною дескрипторів. _____

3. Програмна реалізація розроблених методів класифікації зображень. _____

4. Дослідження та тестування розробленої моделі. _____

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри) актуальність проблеми класифікації зображень, постановка задачі, графічні ілюстрації принципів роботи алгоритмів, блок-схеми роботи алгоритмів класифікації зображень, приклади еталонних зображень з обраних баз даних зображень, таблиці значень результатів класифікації зображень, гістограми часу виконання класифікації досліджуваними методами

6. Консультанти розділів роботи (п.6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Строк / терміни виконання етапів роботи	Примітка
1	Отримання завдання на кваліфікаційну роботу	29.09.2025	
2	Аналіз завдання, підбір літератури	30.09.25-07.10.25	
3	Аналіз літератури з досліджуваної проблеми	08.10.25-14.10.25	
4	Дослідження методів класифікації зображень	15.10.25-20.10.25	
5	Дослідження методів кластеризації	21.10.25-25.10.25	
6	Програмна реалізація	26.10.25-07.11.25	
7	Обґрунтування отриманих результатів	08.11.25-09.11.25	
8	Оформлення пояснювальної записки	10.11.25-21.11.25	
9	Перевірка на нормоконтроль	22.11.25-24.12.25	
10	Перевірка на плагіат	25.11.25	
11	Рецензування	25.12.25	
12	Підготовка презентації та доповіді	26.11.25-02.12.25	
13	Занесення роботи в електронний архів	03.12.25	
14	Попередній захист кваліфікаційної роботи	03.12.25	

Дата видачі завдання 29 вересня 2025 р.

Здобувач _____
(підпис)

Керівник роботи _____
(підпис)

проф. Гороховатський В. О.
(посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка до кваліфікаційної роботи: 60 с., 20 табл., 20 рис., 37 джерел.

АПРОКСИМАЦІЯ СТРУКТУРНОГО ОПИСУ, ВАГОВІ КОЕФІЦІЄНТИ, ДЕСКРИПТОРИ КЛЮЧОВИХ ТОЧОК, ДЕТЕКТОР BRISK, КЛАСИФІКАЦІЯ ЗОБРАЖЕНЬ, КЛАСТЕРИЗАЦІЯ, КОМП'ЮТЕРНИЙ ЗІР.

Об'єктом дослідження є методи класифікації зображень за описом у формі множини дескрипторів ключових точок.

Метою дослідження є розроблення швидкісного класифікатора зображень, що використовує вагові коефіцієнти, отримані в результаті кластерного подання структурного опису.

Аналізується ефективність ряду модифікацій пропонованого методу, здійснено порівняльний аналіз з традиційними методами класифікації зображень.

Використано методи k -середніх та k -медоїд для формування кластерного подання описів зображень. Застосовано детектор BRISK для отримання множини дескрипторів зображень.

Наукова новизна роботи полягає у дослідженні ефективності методу класифікації зображень на основі вагових коефіцієнтів у порівнянні з іншими класичними методами.

Взаємозв'язок з іншими роботами полягає у розробленні модифікацій методу класифікації зображень на основі лінійного пошуку.

Експерименти підтвердили суттєві переваги розробленого методу за критерієм швидкодії зі збереженням необхідного рівня точності.

Впровадження розглянутих у роботі методів доцільно у прикладних задачах комп'ютерного зору з високими вимогами до швидкодії класифікації візуальних об'єктів.

ABSTRACT

Explanatory note to the qualification work: 60 pages, 20 tables, 20 figures, 37 sources.

APPROXIMATION OF STRUCTURAL DESCRIPTION, BRISK DETECTOR, CLUSTERIZATION, COMPUTER VISION, IMAGE CLASSIFICATION, KEYPOINT DESCRIPTORS, WEIGHT COEFFICIENTS.

The object of the research is the methods of classifying images by description in the form of a set of key point descriptors.

The purpose of the research is to develop a high-speed image classifier that uses weight coefficients obtained as a result of the cluster representation of the structural description.

The effectiveness of a number of modifications of the proposed method is analyzed. A comparative analysis is carried out with traditional methods of image classification.

The methods of k -means and k -medoid are used to form a cluster representation of image descriptions. The BRISK detector is used to obtain a set of image descriptors.

The scientific novelty of the work lies in the study of the effectiveness of the image classification method based on weight coefficients in comparison with other classical methods.

The relationship with other works lies in the development of modifications of the image classification method based on linear search.

Experiments have confirmed the significant advantages of the developed method in terms of speed while maintaining the required level of accuracy.

The implementation of the methods considered in the work is advisable in applied computer vision problems with high requirements for the speed of classification of visual objects.

ЗМІСТ

Перелік умовних позначень, символів, одиниць, скорочень і термінів	6
Вступ.....	7
1 Аналіз основних методів класифікації зображень.....	8
1.1 Основні поняття у класифікації зображень.....	8
1.2 Аналіз методів кластеризації та їх використання.....	11
1.3 Огляд предметної області комп'ютерного зору	15
1.4 Постановка задачі дослідження.....	17
2 Моделі класифікації зображень за множиною дескрипторів	19
2.1 Класифікація на основі лінійного пошуку	19
2.2 Кластеризація структурного опису	21
2.3 Метод класифікації з використанням вагових коефіцієнтів.....	24
2.4 Модифікації методу класифікації на основі вагових коефіцієнтів..	26
2.5 Способи редукції опису зображень.....	28
2.6 Метрики відстані у спрощеному просторі ознак.....	30
2.7 Опис методів для тестування.....	33
3 Аналіз результатів комп'ютерного моделювання	35
3.1 Обґрунтування вибору програмного середовища	35
3.2 Дослідження та тестування розробленої моделі.....	36
Висновки	55
Перелік джерел посилання	57

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

КЗ – комп'ютерний зір

КТ – ключова точка

ІІІ – штучний інтелект

AKAZE – Accelerated-KAZE (прискорений KAZE)

BRIEF – Binary Robust Independent Elementary Features (бінарні надійні незалежні елементарні ознаки)

BRISK – Binary Robust Invariant Scalable Keypoints (бінарні надійні інваріантні масштабовані ключові точки)

CNN – Convolutional Neural Network (згорткова нейронна мережа)

FAST – Features from Accelerated Segment Test (ознаки з прискореного сегментного тесту)

OpenCV – Open-Source Computer Vision Library (Бібліотека комп'ютерного зору з відкритим кодом)

ORB – Oriented FAST and Rotated BRIEF (орієнтований FAST та обернений BRIEF)

SURF – Speeded-Up Robust Features (прискорені надійні ознаки)

SIFT – Scale-invariant Feature Transform (інваріантні до масштабу трансформації ознак)

ВСТУП

Класифікація зображень є завданням, яке досліджується вже багато років, і попри активний розвиток комп'ютерного зору та нейронних мереж залишається невирішеним. Все більше сучасних застосунків, особливо з прогресом у розвитку штучного інтелекту за останні роки, використовують класифікацію зображень як певну частину свого основного функціоналу. Зміни масштабу, нахилу та позиції зображення, інші геометричні перетворення і погіршеності можуть зменшувати якість класифікації і, як результат, погіршують досвід користування програмним забезпеченням, або призводять до помилок, що можуть мати негативні наслідки для користувача або розробника [1-5].

До того ж все зрозумілішим стає факт важливості швидкого виконання процесу розпізнавання зображень, враховуючи те, що він часто не є фінальним етапом обробки інформації у застосунках або програмах, що стикаються з цією задачею. Швидка обробка допомагає зберегти увагу користувача та запобігти невдоволенню якістю роботи програмного забезпечення.

Саме тому дослідження та розроблення швидкісних та якісних методів класифікації зображень, а також їх налаштувань, є актуальними темами у сфері комп'ютерного зору та інформаційних технологій. Такі методи можуть бути використані у великій кількості застосунків та програм незалежно від напрямку та цілі застосування.

Актуальність роботи полягає у важливості створення удосконалень для алгоритмів класифікації зображень, що націлені на підвищену швидкодію для застосування у комп'ютерних системах з необхідністю у швидкій обробці даних для ефективної роботи.

Завданням роботи є розробка методу класифікації зображень, що спирається на апроксимацію опису зображень через центроїди кластерного подання опису з використанням вагових коефіцієнтів кластерів з подальшим аналізом ефективності методу та трьох модифікацій, включаючи порівняння з іншими методами класифікації зображень.

1 АНАЛІЗ ОСНОВНИХ МЕТОДІВ КЛАСИФІКАЦІЇ ЗОБРАЖЕНЬ

1.1 Основні поняття у класифікації зображень

Комп'ютерний зір – це галузь інформатики та штучного інтелекту, що займається задачами з аналізу, виділення та інтерпретації корисної інформації з фото- та відеоданих завдяки програмам чи комп'ютерним системам [1].

Галузь комп'ютерного зору охоплює та вирішує велику кількість задач, що знаходять застосування у значній кількості сучасних застосунків та систем. Основні задачі комп'ютерного зору це [1]:

- класифікація об'єктів (визначення належності об'єкта до певного відомого класу);
- виявлення об'єктів (визначення на зображенні, чи відео частин зображення, що містять відомі системі об'єкти);
- трекінг (визначення руху об'єкта у відеоряді);
- сегментація (розбиття зображення, або відео на окремі сегменти за вмістом);
- визначення тривимірних об'єктів, розуміння глибини, особливостей сцени та просторового контексту;
- генерація та покращення зображень (розпізнавання категорій зображень, які модель не бачила під час навчання);

Класифікація зображень – це задача комп'ютерного зору, що полягає у визначенні мітки (або міток) класу, до якого належить вхідне зображення. Іншими словами, класифікація зображень полягає у визначенні вмісту зображення [1]. При класифікації мітки класів вже є попередньо відомі, тобто на основі вмісту зображення визначається його належність до заздалегідь відомої та сформованої групи.

Класифікація зображень відбувається у 4 етапи: отримання зображень, попередня обробка, виділення ознак та класифікація [2-4]. На рисунку 1.1 зображено основні етапи класифікації зображень у традиційному варіанті.

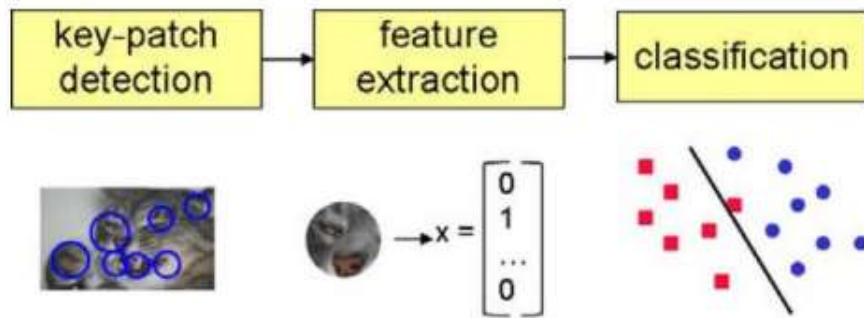


Рисунок 1.1 – Приклад традиційного варіанту класифікації зображень

На етапі отримання зображень збирається база даних та ознак зображень, на основі якої відбуватиметься класифікація, а також маркування зображень на відповідні класи у формі еталонів. Під час попередньої обробки зображення проходять такі видозміни як видалення шуму, геометричні перетворення, та інші операції для стандартизації бази зображень [2]. Ці два етапи є підготовчими та слугують основою успішної роботи класифікатора, оскільки під час виконання цих дій підтверджується якість вхідних даних.

Під час виділення ознак відбувається знаходження та опис ключових точок зображення на основі використання детекторів. Ключові точки – це точки на зображенні, що описують певні його важливі зони. Для більш компактного та зрозумілого для комп'ютера відображення ключових точок, формуються та використовуються вектори, що описують інформацію з ключової точки – дескриптори.

Існує велика кількість детекторів КТ, що вважають різні зони на зображенні за важливі, але зазвичай за ключові точки вважаються кути, ребра та плями [3, 4]. Одним з найбільш надійних і математично обґрунтованих детекторів є SIFT, проте його використання не є бажаним у програмах, що потребують швидкої обробки даних, оскільки цей алгоритм працює повільно.

З надійних сучасних швидкісних способів виділення ознак часто відмічають детектор BRISK, заснований на алгоритмі FAST. FAST вишукує кути на зображенні, вважаючи кожен піксель як кандидат, навколо якого малюється коло з 16 суміжних пікселів (рис. 1.2). Кутом вважається кандидат, у колі якого

є як мінімум n пікселів, інтенсивність (значення від 0 до 255, що відображає яскравість пікселя) яких відрізняється хоча б t на від інтенсивності центра.

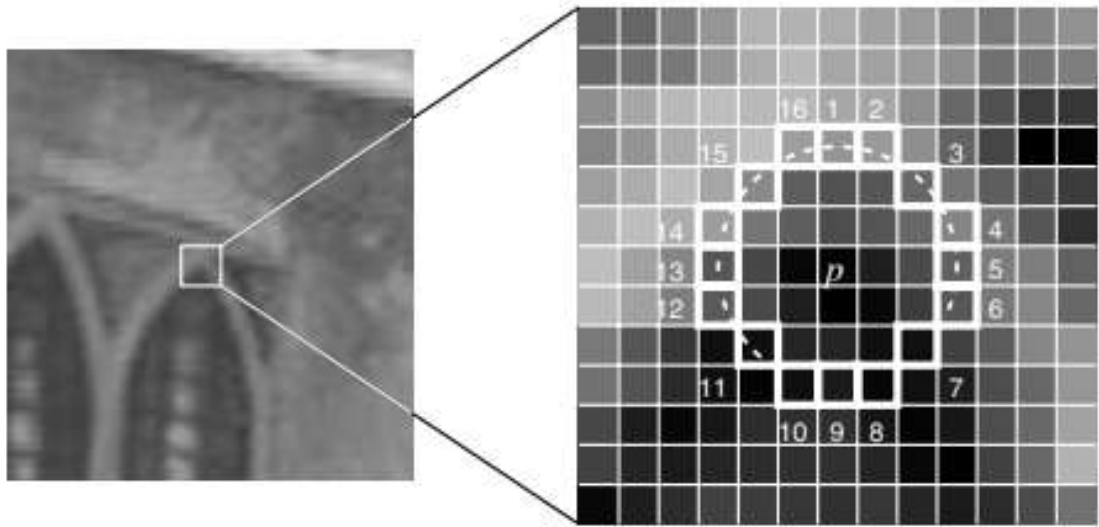


Рисунок 1.2 – Коло точки-кандидата

BRISK застосовує принцип роботи FAST, розраховуючи його для простору масштабів – піраміди з певної кількості октав, на яких знаходяться різні масштаби зображень. Точка-кандидат відкидається не тільки якщо вона не проходить перевірку за FAST, а також якщо ця перевірка не проходить на сусідніх октавах [3].

Залежно від поставленої задачі, або вхідних даних, етап класифікації зображень може відбуватись за різними варіантами.

Існують такі види класифікації зображень [1]:

- бінарна класифікація (визначається чи належить зображення до класу, чи ні);
- багатокласова класифікація (до якого з великої кількості класів належить зображення);
- багатоміткова класифікація (схожа на багатокласову, де відмінність полягає в тому, що зображення може отримати одразу декілька міток класу);
- ієрархічна класифікація (розпізнавання міток класу на різних рівнях ієрархії);

- детальна класифікація (розпізнавання зображень з дуже схожих категорій);
- класифікація з нульовими прикладами (розпізнавання категорій зображень, які модель не бачила під час навчання);
- класифікація з малою кількістю прикладів (розпізнавання зображень на основі мінімальної кількості прикладів для кожного класу);

1.2 Аналіз методів кластеризації та їх використання

Кластеризація є ефективним методом апроксимації та спрощення описів зображень у структурних методах класифікації даних [4]. У більшості випадків опис зображення, отриманий детекторами ключових точок, таких як BRISK, складається з сотень, або тисяч дескрипторів різного розміру, в залежності від обраного детектора. Використання кластерного аналізу та вираження описів зображень через центроїди кластерів, або інші схожі структури на основі подібних дескрипторів, дозволяє значно зменшити розмірність опису зображень та надати сильне прискорення алгоритмам класифікації. Завдяки такому підходу описи зображень можна зменшити з декількох сотень до 2-10 бітових векторів, які приблизно усереднюють дані, що зберігаються в множинах дескрипторів зображень.

Кластерний аналіз (або кластеризація) – це задача розподілу існуючого набору даних (патернів) на групи (кластери) на основі попередньо обраного критерію схожості [5, 6]. Певний об'єкт є більш схожим на інший об'єкт зі свого кластеру, ніж з іншого. Кластеризація дає зрозуміти, виключно на основі вхідних даних, та без врахування жодних попередніх маркувань, як виглядають ці групи та які дані є більш схожими, або відмінними.

В залежності від вибору алгоритму, або критерію схожості, навіть на однакових наборах даних, можна отримати різні варіанти групування (рис. 1.3).

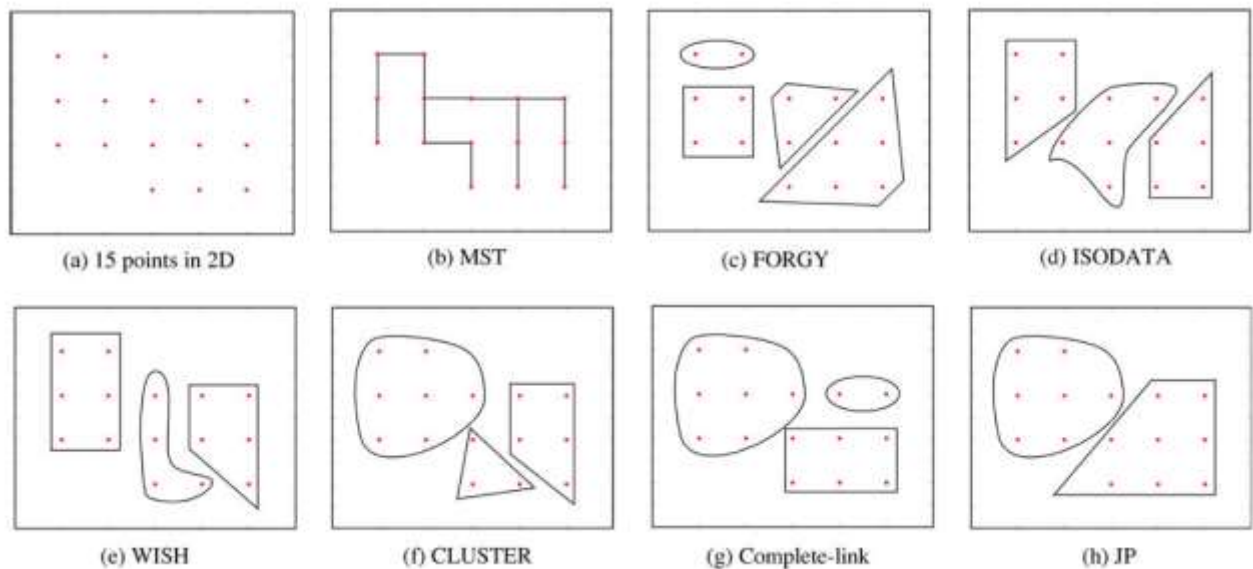


Рисунок 1.3 – Результат кластеризації 15 точок у двовірному просторі різними алгоритмами кластеризації

Цей факт підкреслює важливість визначення найбільш доцільного алгоритму для задачі, що виконується, а також підбору метрики, що визначатиме близькість елементів бази.

Існує різноманіття алгоритмів кластеризації, що виконують групування даних за різними принципами (рис. 1.4). Дві основні великі групи алгоритмів кластеризації – ієрархічні та роздільні (partitional) [5, 6]. Різниця між цими групами полягає у тому, що ієрархічні алгоритми створюють низку упорядкованих розділів (у вигляді дендрограми – дерева кластерів), коли як роздільні створюють лише один розподіл.

Найвідомішими ієрархічними алгоритмами є однозв'язний та повнозв'язний алгоритми кластеризації, що на кожному кроці з'єднують кластери на основі найменшої та відповідно найменшої максимальної відстані між ними. Найбільш популярним роздільним алгоритмом є k -середніх, що відносить елементи бази даних до кластеру, відстань до центру якого є мінімальною, після чого оновлює центроїди на основі середнього арифметичного значень в кластері. Метод k -середніх, окрім роздільних алгоритмів, також можна віднести до групи алгоритмів кластеризації, заснованих на центроїдах [6].

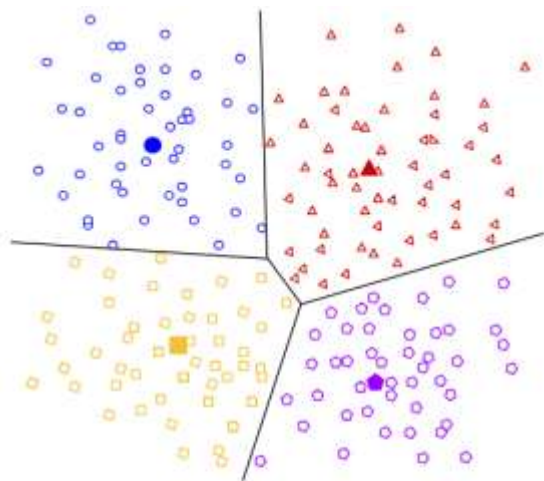


Figure 1: Example of centroid-based clustering.

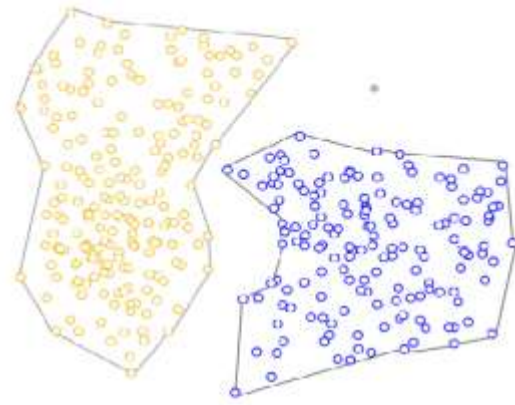


Figure 2: Example of density-based clustering.

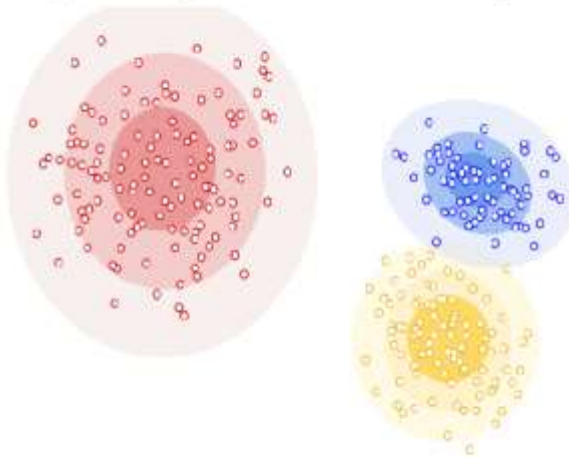


Figure 3: Example of distribution-based clustering.

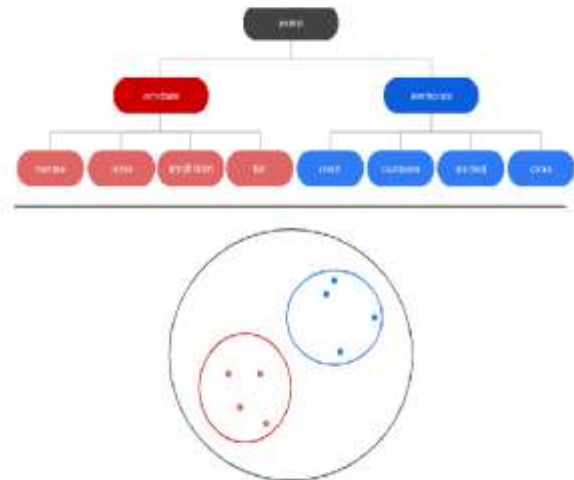


Figure 4: Example of a hierarchical tree clustering algorithm.

Рисунок 1.4 – Приклади видів алгоритмів кластеризації

Роздільні алгоритми містять у собі велику кількість різних підвидів алгоритмів, що працюють за абсолютно різними механізмами. Прикладами таких видів алгоритмів є щільнісні (density-based clustering), або основані на розподілі (distribution-based clustering) [7, 8].

Щільнісні алгоритми вважають кластери щільною сукупністю точок у просторі, розділених простором з низькою щільністю [7, 8]. Такі алгоритми (наприклад DBSCAN або DENCLUE) вміють добре ідентифікувати кластери довільної форми, а також добре відкидати шум. Алгоритми, основані на розподілі, вважають, що дані складаються з імовірнісних розподілів (наприклад Гаусівського), та додають елементи до кластерів на основі ймовірності їх належності до відповідного розподілу.

У великій кількості програм або систем, з метою спрощення та абстракції результатів кластеризації, виникає потреба у вираженні кластера через певні елементи або структури [4, 5]. Існують такі варіанти форматів вираження кластерів (рис. 1.5):

- через точки (центроїд, або три найбільш віддалені точки);
- через дерево рішень;
- через логічні вирази.

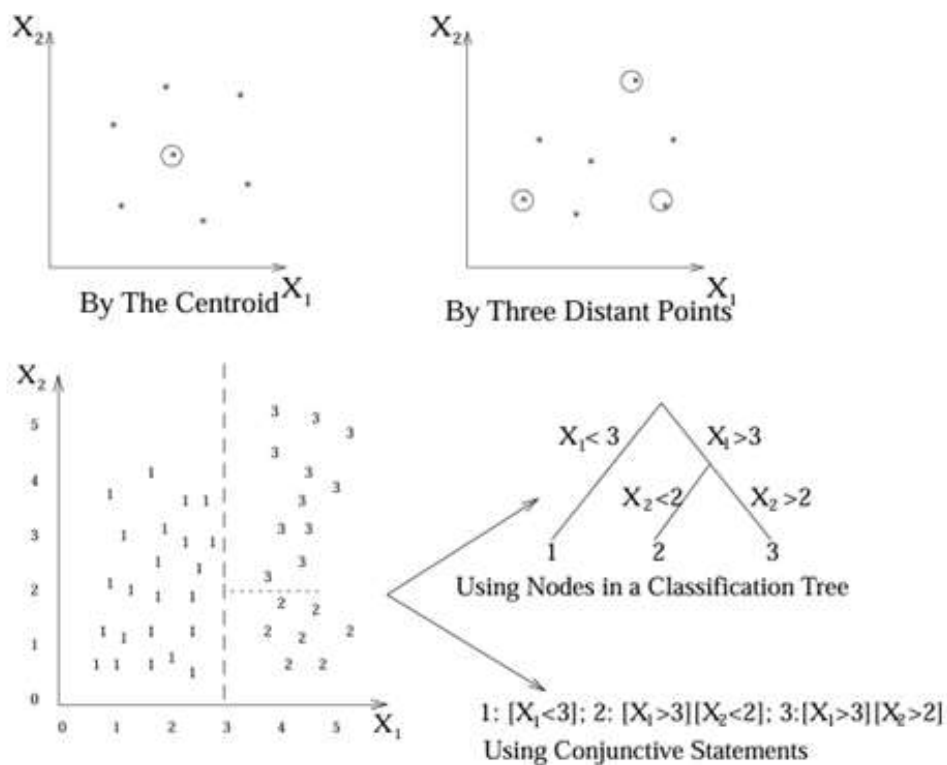


Рисунок 1.5 – Способи вираження кластерів

Вираження кластерів через центроїд є найбільш розповсюдженим та простим – одна точка, що усереднює кластер. Такий підхід працює найкраще всього працює з компактними кластерами. У випадку, коли кластер має нерівну, або довгу форму, опис через три найбільш віддалені точки дає змогу апроксимувати форму кластера та зберегти компактність вираження.

Отже, існує велика кількість алгоритмів кластеризації, що підходять для різних задач, в залежності від вигляду чи формату даних, чи бажаної швидкодії. Окрім визначення груп схожих елементів даних, кластеризація дає змогу

представлення кластерів через спрощене подання у вигляді окремих елементів або структур, згенерованих після кластеризації.

1.3 Огляд предметної області комп'ютерного зору

Розроблення швидкісних алгоритмів, націлених на швидке оброблення даних, з потенційною можливістю застосування в системах реального часу є важливим завданням, що також розглядається у сучасності комп'ютерного зору. Одним з таких алгоритмів, що широко застосовується зараз, є алгоритм виявлення об'єктів YOLO. Замість підходу до детекції, що включає в себе виділення регіонів інтересу на зображенні з подальшою класифікацією цих окремих частин, YOLO уникає таких операцій з метою збереження часу [9]. Натомість, усі кроки виявлення об'єктів – локалізація, класифікація та оцінка достовірності виконуються за один прохід згорткової нейронної мережі (CNN).

На початковому етапі алгоритм зменшує зображення до потрібного розміру та накладає на нього сітку розміром S на S (автори [9] використовували розмір 448×448) з рівномірно розподіленими комірками.

Кожна комірка сітки прогнозує положення об'єктів, виводячи кілька значень: координати x та y центру прямокутника (відносно комірки сітки), ширину та висоту прямокутника, а також значення достовірності, яке відображає ймовірність присутності об'єкта, після чого прогнозовані значення декодуються в абсолютні координати на зображенні. Такі прямокутники називаються обмежувальною рамкою [9].

Після етапу декодування YOLO обчислює оцінки класів для кожної передбаченої обмежувальної рамки, де кожна оцінка представляє ймовірність того, що виявлений об'єкт належить до певного класу. Остаточна оцінка достовірності для кожного виявлення отримується шляхом множення впевненості у наявності об'єкту в рамці на ймовірність класу. Виявлення з достовірністю нижче вибраного порогу відкидаються, щоб видалити рамки, що

є ненадійними кандидатами на детекцію. Оскільки кілька рамок можуть перетинатись та відповідати положенню на зображенні одного й того ж об'єкта, YOLO порівнює їхні значення перетину над об'єднанням (intersection over union) та зберігає лише рамку з найвищою оцінкою для кожного об'єкта, та видаляє решту.

На завершальному етапі роботи, модель масштабує обмежувальні рамки відповідно до початкового розміру зображення та видає на вихід список виявлених об'єктів, кожен з яких має свої координати, мітку класу та оцінку достовірності.

На рисунку 1.6 графічно зображено принцип роботи детектора об'єктів YOLO.

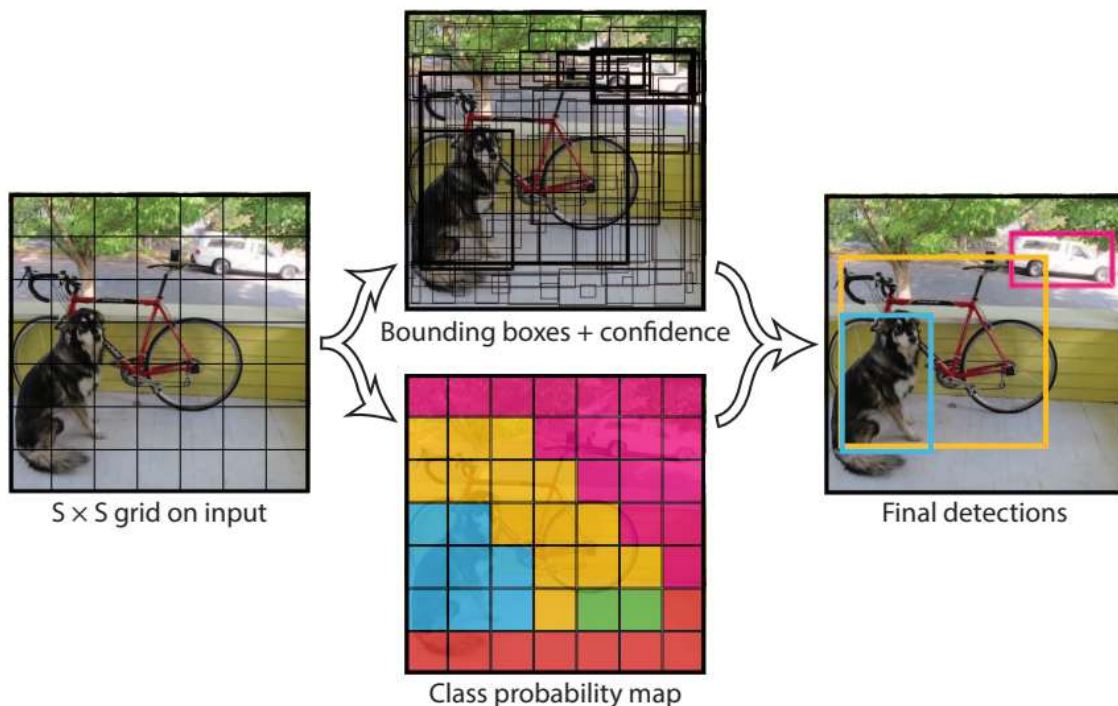


Рисунок 1.6 – Принцип роботи YOLO

Детектор об'єктів YOLO виграє у швидкодії у порівнянні з методиками, що мають двоетапний підхід до задачі винайдення об'єктів, ціною точності [9]. Схожим за принципом підходом до детекції об'єктів є Fast R-CNN. Обидва методи застосовують згорткові нейронні мережі для визначення обмежувальних

рамок та класів об'єктів, проте основна відмінність цього підходу у порівнянні з YOLO полягає у тому, що Fast R-CNN застосовує метод вибіркового пошуку для виявлення зон інтересу на зображенні, які далі аналізує CNN для отримання результатів детекції. Результати порівняння точності детекторів зображені на рисунку 1.7.

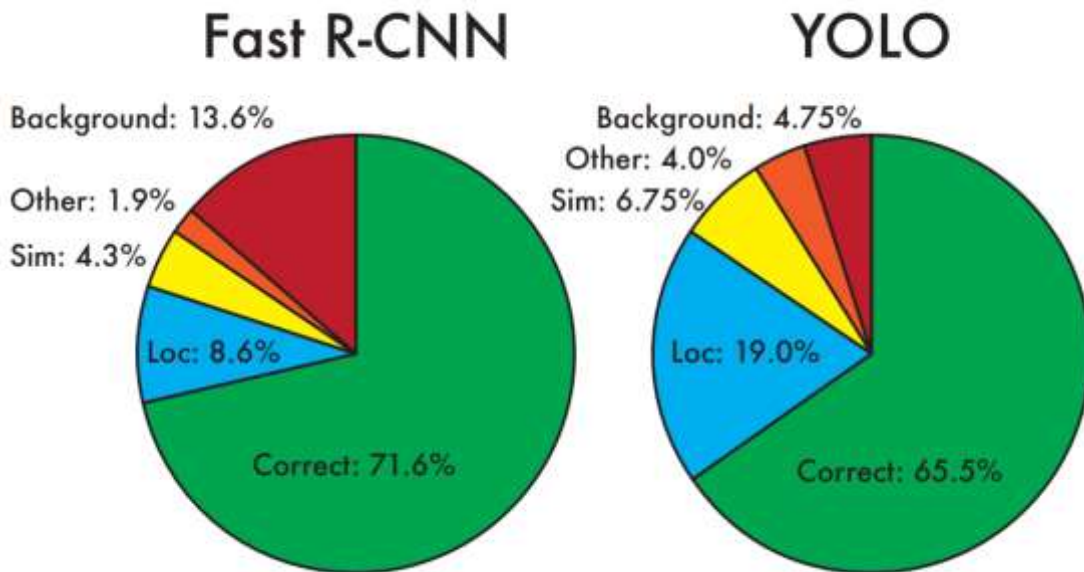


Рисунок 1.7 – Порівняння якості детекції алгоритмами Fast R-CNN та YOLO

З рисунку 1.7 видно, що загалом YOLO трохи програє у точності, також спостерігаємо, що алгоритми по-різному справляються з різними видами помилок виявлення об'єктів. При цьому YOLO виграє у швидкодії у 90 разів [9]. Отже, враховуючи переваги швидкісних алгоритмів комп'ютерного зору, з метою застосування у системах для вирішення реальних задач, створення алгоритмів націлених на підвищення швидкодії є важливою задачею.

1.4 Постановка задачі дослідження

Отже, розробка методів класифікації зображень з орієнтацією на швидкодію є актуальним завданням. Для цього прийняте рішення з розробки

метода класифікації зображень, заснованого на накопиченні вектора вагових коефіцієнтів кластерного подання опису, а також модифікацій методу з метою його пришвидшення, або потенційного підвищення точності.

Об'єктом роботи є методи класифікації зображень за описом у формі множини дескрипторів ключових точок.

Метою дослідження є розроблення швидкісного класифікатора зображень на основі використання вагових коефіцієнтів для кластерів та визначення впливу різних модифікацій та способів попередньої обробки на якість та швидкодію класифікації.

Для досягнення мети необхідно вирішити такі завдання:

- проаналізувати актуальні існуючі методи класифікації зображень;
- реалізувати традиційний метод класифікації, що використовує принцип лінійного пошуку;
- реалізувати алгоритм класифікації на основі накопичення повного вектора вагових коефіцієнтів;
- реалізувати модифікацію алгоритму з використанням параметру однорідності для пришвидшення класифікації;
- протестувати алгоритм з різними методами кластеризації, та модифікаціями з метою визначення доцільності його використання у порівнянні з традиційним методом та методом на основі мінімуму Манхеттенської відстані за критеріями точності та швидкодії.

2 МОДЕЛІ КЛАСИФІКАЦІЇ ЗОБРАЖЕНЬ ЗА МНОЖИНОЮ ДЕСКРИПТОРІВ

2.1 Класифікація на основі лінійного пошуку

Нехай $E_k = \{e_v(k)\}_{v=1}^s$ – множина дескрипторів деякого еталонного зображення k з бази еталонів $E = \{E_1, E_2, \dots, E_N\}$, де $s = \text{card}E_k$ – кількість елементів множини дескрипторів еталону. Нехай $Z = \{z_v\}_{v=1}^s$ – множина дескрипторів вхідного зображення, s – стандартизоване та є еквівалентним для еталонів та вхідних зображень, задля спрощення аналізу [10].

Традиційний класифікатор має двоетапну структуру $R = R_2R_1$: на першому (R_1) – визначення класів елементів кожного з дескрипторів множини Z вхідного зображення, на другому (R_2) – визначення класу зображення на основі голосування, використовуючи мітки класів дескрипторів з R_1 .

Класифікатор R_1 виконує поетапне порівняння кожного дескриптора з множини дескрипторів Z вхідного зображення з усіма дескрипторами E бази даних еталонних зображень

$$R_1: z_v \rightarrow \{1, \dots, N\}. \quad (2.1)$$

Клас вхідного дескриптора z_v при цьому визначається як клас дескриптора еталонного зображення, що знаходиться на мінімальній відстані від порівнюваного дескриптора:

$$R_1: k = \arg \min_{i=1, \dots, N; d=1, \dots, s} \rho(z_v, e_d(i)), \quad (2.2)$$

де $\rho(z_v, e_d(i))$ – відстань між дескриптором вхідного та еталонного зображення. Враховуючи те, що дані, між якими відбувається порівняння, є бінарними векторами, є доцільним використання метрики Геммінга через її ефективність у

просторі бінарних даних та простоті функціонування. Відстань Геммінга розраховується як кількість відмінних бітів на однакових позиціях у векторах бінарних даних [3, 10].

$$\rho(b_k, z_v) = \sum_{a=1}^n 1(b_{k,a}, z_{v,a}), \quad (2.3)$$

де $1(b_{k,a}, z_{v,a})$ – функція порівняння бітів на позиції a для двох векторів, що дорівнює нулю у разі співпадіння бітів, або одиниці в протилежному випадку.

Для виконання класифікатора R_2 введено вектор $\{h_i\}_{i=1}^N$ цілих чисел, що містить кількість голосів дескрипторів вхідного зображення за класи (значення від 0 до s). Лічильники класів інкрементно збільшуються $h_k = h_k + 1$ на основі даних, отриманих з класифікатора R_1 для кожного дескриптора вхідного зображення $z_v \in Z$. Результатом класифікації є мітка класу, що отримав найбільшу кількість голосів з вектора $\{h_i\}_{i=1}^N$:

$$R_2: Z \rightarrow E_k | (k = \arg \max_{i=1, \dots, N} h_i) \ \& \ (h_k \geq \delta_h), \quad (2.4)$$

де δ_h – поріг для мінімально прийнятної кількості голосів. Поріг введено з метою визначення зображень поза базою та недопущення їх до отримання мітки класу. Схему роботи традиційного класифікатора зображено на рисунку 2.1.



Рисунок 2.1 – Принцип роботи традиційного класифікатора

2.2 Кластеризація структурного опису

Як було зазначено у пункті 1.2, кластеризація може застосовуватись у алгоритмах з метою знаходження груп схожих об'єктів множини з подальшою можливістю репрезентації кластерів через окремі елементи чи структури (наприклад центроїд).

Отже, представимо базу еталонних зображень E через множину T непересічних кластерів, до одного з M яких заноситься кожний дескриптор E_i зображень [10].

$$E_i = T(E_i) = \bigcup_{k=1}^M T_k(E_i), k = \overline{1, M}, T_k(E_i) \cap T_j(E_i) = \emptyset. \quad (2.5)$$

Представимо кластери через центри $b_{k,i}$ з метою спрощення подання опису еталонних зображень. Представлення вектора описів через центроїд підвищує швидкодію класифікатора через скорочення кількості операцій порівнянь.

У стандартному вигляді метод k -середніх випадково обирає M точок з множини, які вважає за центри кластерів. Точки далі порівнюються з кожним центром з метою визначення найближчого з них, після чого заносяться до відповідного кластера. Кластеризація завершується, якщо кластери не змінюються між ітераціями, або за досягнення бажаного значення похибки кластеризації [4, 11].

Метод k -середніх має ваду, що полягає у відсутності адаптації методу до різних типів даних. У звичайному вигляді алгоритм працює з числовими даними, для яких можна просто розрахувати середнє арифметичне, тому для використання методу з бінарними векторами, потрібно застосувати модифікацію логіки розрахунку центрів кластерів. Розрахунок центроїдів b_i пропонується проводити за формулою

$$b_i(a) = \begin{cases} 1, & f(T^i, a) \geq s(T^i)/2, \\ 0, & \text{інакше,} \end{cases} \quad (2.6)$$

де $b_i(a)$ – значення біту на позиції a центру b_i ;

$f(T^i, a)$ – функція для підрахунку кількості одиниць на позиції a для кожного з елементів кластеру.

Алгоритм k -медоїд є схожим на алгоритм k -середніх, де ключова різниця полягає у визначенні опорного елемента кластера. Замість розрахунку центроїда на основі середнього арифметичного елементів кластера, k -медоїд обирає елемент кластера, що знаходиться на найменшій сумарній відстані від всіх елементів кластера. Даний метод кластеризації групує дані на основі реального елемента з бази даних, що апроксимує середину кластеру [11].

Принцип роботи та основну різницю обох методів кластеризації графічно описано на рисунку 2.2.

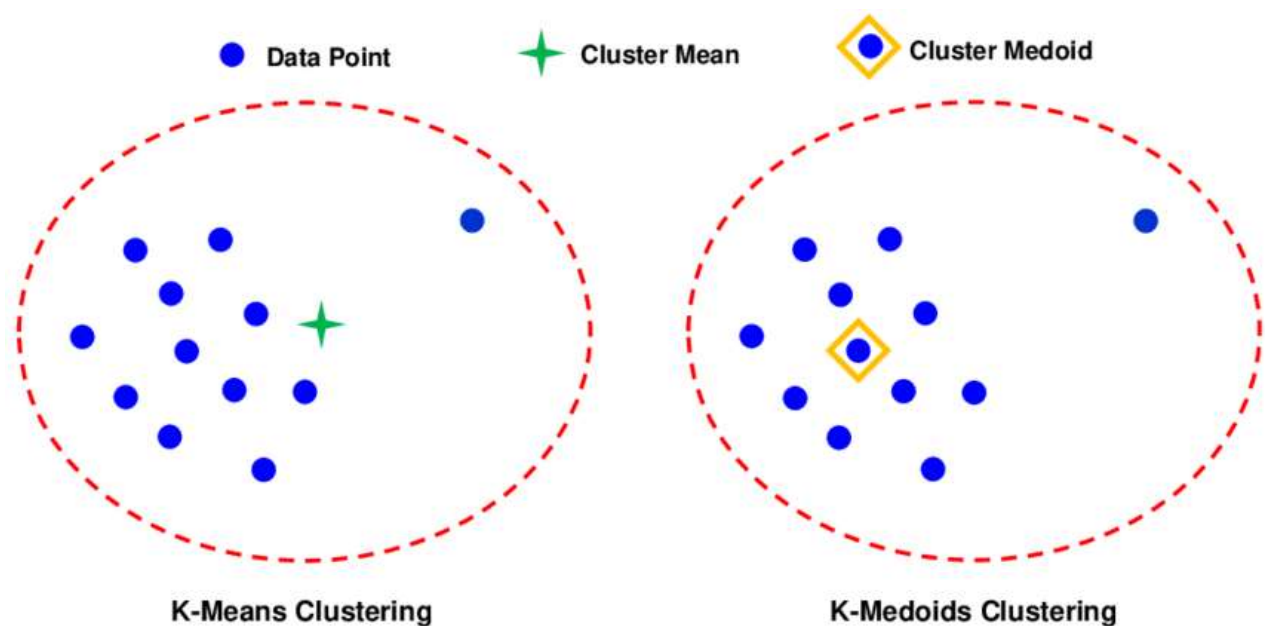


Рисунок 2.2 – Принцип роботи алгоритмів k -середніх та k -медоїд

Алгоритм DBSCAN належить до щільнісних алгоритмів кластеризації. Метод вводить низку понять, якими він оперує для визначення належності точок до кластерів, або визначення як шум.

Поняття ε -околиці точки p ($N_\varepsilon(p)$) визначено як множину точок з набору даних, що знаходяться на відстані, що не перевищує ε . Будь-яка точка q , яка знаходиться у $N_\varepsilon(p)$ за умови якщо множина $N_\varepsilon(p)$ більша за розмірністю за

визначене користувачем ціле число $MinPts$ мінімальне число точок в околиці, називається щільнісно-досяжною з врахуванням ϵ та $MinPts$. Точка, що містить відповідну кількість точок у своїй ϵ -околиці зветься ядром [7]

$$card(N_\epsilon(p)) \geq MinPts. \quad (2.7)$$

Приклад властивості щільнісної досяжності для деяких точок p та q продемонстровано на рисунку 2.3.

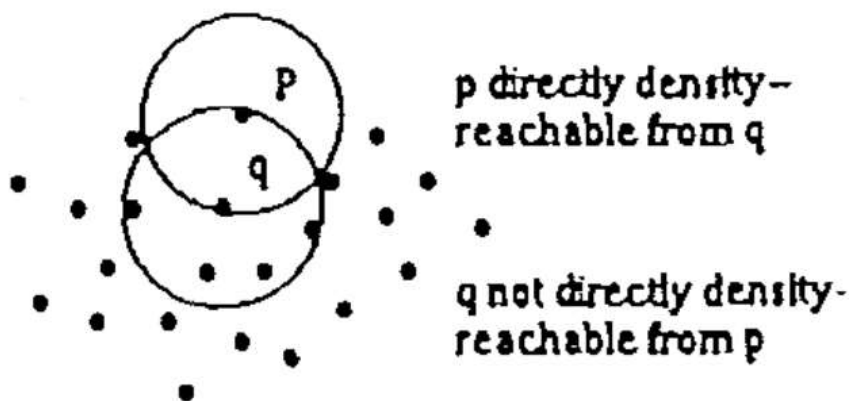


Рисунок 2.3 – Візуалізація поняття щільнісної досяжності

Введемо поняття з'єднаності за щільністю. Точка p вважається з'єднаною за щільністю з точкою q , якщо існує третя точка o , з якої до точок p та q можна побудувати ланцюги з точок, у яких кожна наступна точка є щільнісно-досяжною до наступної [7]. Візуалізація цього принципу зображена на рисунку 2.4.

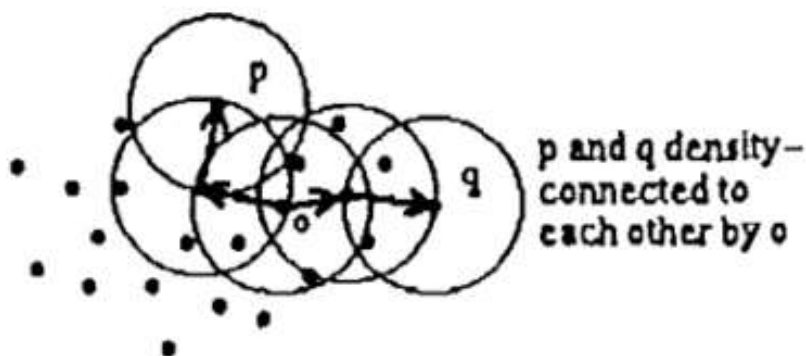


Рисунок 2.4 – Візуалізація поняття з'єднаності за щільністю

Враховуючи це поняття, кластером для DBSCAN вважається ненульова підмножина набору даних, що містить точки, кожна з яких є щільнісно-з'єднаною, при чому у множині наявна кожна точка, досяжна за щільністю з будь-якої іншої наявної у кластері точки [7].

Сам алгоритм працює таким чином: аналізуємо точку з набору даних, що не має мітки кластеру та перевіряємо її на умову ядра (2.7). Якщо точка є ядром, починається операція створення нового кластера з додаванням усіх щільнісно-досяжних точок для даного ядра. До кластера ітеративно додаються усі точки, що задовольняють визначення кластера у DBSCAN. Точки, що не є ядрами на початковому етапі позначаються як шум з подальшою можливістю включення цих точок до деякого кластеру на пізніших ітераціях алгоритму. Повторюємо цю процедуру поки кожна точка з набору даних не отримає мітку класу, чи отримає позначку шуму.

Оскільки метод не має параметру для генерації кількості кластерів, якість кластеризації сильно залежить від обраних значень вхідних параметрів ϵ та $MinPts$. Алгоритм найкраще підходить для кластеризації щільно згрупованих множин довільного розміру з помітною відстанню між ними. Хоч автори алгоритму [7] мають на увазі методи, націлені на визначення вхідних параметрів, доцільність використання методу на бінарних векторах складно оцінити через складність їх візуалізації. За певних небажаних комбінацій значень вхідних параметрів алгоритм має тенденцію до створення одного великого кластеру з деякими точками відміченими як шум, чи створення великої кількості маленьких кластерів, що в подальшому сильно відобразиться на якості класифікації [6, 7].

2.3 Метод класифікації з використанням вагових коефіцієнтів

Розглянемо принцип побудови класифікатора, заснованого на накопиченні вектора вагових коефіцієнтів.

Для роботи, як етап попередньої обробки, класифікатор потребує кластеризації бази дескрипторів еталонів E за довільним методом кластеризації з метою знайдення центроїдів. Дескриптори еталонних зображень далі порівнюються з центроїдами (аналогічно до принципу, застосованого у класифікації традиційним методом) з метою отримання матриці Φ голосів еталонів за центри кластерів, що визначатиме значення ваги, яку отримає певний клас під час класифікації.

Після кластеризації бази дескрипторів еталонів E та отримання матриці голосів еталонів за центри кластерів Φ , елементи бази дескрипторів вхідного зображення Z порівнюються з центроїдами за відстанню Геммінга з метою визначення найближчого з них до відповідного дескриптора. Номер i найближчого центра до дескриптора вхідного зображення є номером стовпчика з матриці Φ , який треба додати до вихідного вектора вагових коефіцієнтів W , як у формулі

$$W_{j+1} = W_j + \Phi_{:,i}. \quad (2.8)$$

Після накопичення повного вектора вагових коефіцієнтів клас вхідного зображення визначається за номером класу ваги, що отримав найбільше значення у векторі W , тобто класифікатор R_2 визначається відповідно до формули

$$R_2: Z \rightarrow E_k; k = \arg \max_{i=1, \dots, N} W_i. \quad (2.9)$$

З метою відкидання зображень поза базою вектор W розраховується для кожного еталонного зображення, де відповідні вагові коефіцієнти за власний клас заносяться до вектора W^E . Вхідне зображення Z отримує мітку класу, якщо найбільше значення його вектора W знаходиться у межах $W_k^E \pm \delta$, тобто виконується принцип формули

$$k = 0, \text{ якщо } W_k^E - \delta \leq W_k \leq W_k^E + \delta. \quad (2.10)$$

Схема принципу роботи методу на основі накопичення повного вектора вагових коефіцієнтів зображена на рисунку 2.5.



Рисунок 2.5 – Схема роботи методу на основі накопичення вектора вагових коефіцієнтів

2.4 Модифікації методу класифікації на основі вагових коефіцієнтів

Якість методу класифікації, оснований на накопиченні повного вектора вагових коефіцієнтів, залежить від якості кластеризації та отриманої матриці Φ . Пропонується використання модифікацій, що змінюють значення всередині матриці Φ з метою потенційного покращення якості та швидкості кластеризації.

Запропоновані такі модифікації: використання логічно обробленого вектора вагових коефіцієнтів стовпця (k найбільших ваги у стовпчику), скорочення числа аналізованих центрів кластерів через попереднє логічне оброблення вектора вагових коефіцієнтів у рядку (k найбільших ваги), а також

скорочення числа аналізованих центрів кластерів через попереднє логічне оброблення даних у стовпці за критерієм однорідності (2.11).

Відкидання найменших значень з рядків та стовпчиків дає змогу залишити лише найважливіші значення матриці Φ . Ці модифікації мають на меті маніпуляцію матриці Φ виключно з метою потенційного покращення точності та не призведуть до покращення швидкодії, оскільки структура матриці не змінюється. Інша модифікація використовує критерій однорідності (2.11) для викидання цілих стовпчиків з матриці.

Критерій однорідності порівнює кожне значення вектора з іншими та визначає наскільки відмінними є ці значення, тобто низьке значення критерію каже про те, що вектор (у нашому випадку це стовпчик матриці Φ) має схожі значення

$$H(x) = \sum_{p=1}^{m-1} \sum_{q=p+1}^m |x_p - x_q|. \quad (2.11)$$

Під час накопичення вагових коефіцієнтів, такі стовпчики даватимуть приблизно однакову вагу кожному класу i , теоретично, не сильно змінюватимуть результат. Отже, видалення таких стовпчиків дозволяє прискорити роботу алгоритму, а також, можливо, підвищити точність класифікації. Схема застосування критерію однорідності зображена на рисунку 2.6.



Рисунок 2.6 – Схема застосування критерію однорідності

2.5 Способи редукції опису зображень

Зменшення розмірності опису бази зображень для класифікації помітно збільшує швидкодію класифікаторів [12]. Редукцію опису можна досягти завдяки використанню критерію значущості з подальшим його застосуванням для відкидання неінформативних елементів опису з множини дескрипторів бази еталонних зображень E на етапі попередньої обробки [12].

Введемо ідентичну за структурою до бази зображень E матрицю $\Lambda = \{ \{ \lambda_{i,v} \}_{v=1}^s \}_{i=1}^N$ чисел, що містить у собі значення параметру значущості для кожного дескриптора еталонних зображень (2.12)

$$\Lambda = \left\{ \begin{array}{c} \lambda_{1,1}, \dots, \lambda_{1,s} \\ \lambda_{2,1}, \dots, \lambda_{2,s} \\ \dots \\ \lambda_{N,1}, \dots, \lambda_{N,s} \end{array} \right\}, \quad (2.12)$$

де $\lambda_{i,v}$ – величина параметра значущості. Даний опис дозволяє отримати уявлення про те, наскільки важливим є будь-який дескриптор для здатності класифікувати відповідне зображення. Слід зазначити, що видалення будь-якого елементу з опису зображень призводить до зміни значень, при чому застосування поетапного відкидання дескрипторів з подальшим перерахунком є більш ефективним за видалення багатьох дескрипторів одразу.

Метою є редукція опису, тому поставимо задачу приведення бази еталонних зображень до вигляду $R: E \rightarrow E^*$, де E^* – підмножина множини описів еталонів E , що має розмір $s^* = \text{card}E^*$, при чому $s^* < s$.

Існує велика кількість методів визначення значущості елементів у просторі описів зображень [12]. Як варіант обчислення значень всередині матриці Λ з можливістю застосування у автоматизованих системах існує метричний критерій інформативності. Для довільного дескриптора $z \in E$ із простору описів еталонних зображень, що належить до деякої з множин векторів дескрипторів

$z \in E_k$ певного еталону E_k з номером k , визначимо критерій інформативності $I(z, E)$ у складі бази E

$$I(z, E) = \rho_m(z, \bar{E}_k) - \rho_m(z, E_k), \quad (2.13)$$

де $\rho_m(z, \bar{E}_k) = \min_{v, i \neq k} \rho(z, e_v(i))$ – мінімальна відстань від дескриптора вхідного зображення z до певного дескриптора з бази еталонних зображень, що не належить класу E_k ;

$\rho_m(z, E_k) = \min_{v, i=k} \rho(z, e_v(i))$ – відстань від z до найближчого елемента із класу E_k (не враховуючи відстані $\rho(z, z) = 0$ до самого себе).

Ідея критерію заснована на тому, що описи, які знаходяться на великій відстані від інших класів матимуть більше значенні інформативності. Знаходження дескриптора на невеликій відстані від власного класу також підвищує його інформативність. Тобто найбільш інформативними дескрипторами в описі будуть такі, що розташовані близько до власного класу та далеко від елементів описів будь-якого іншого еталонного зображення. Виходячи з цього, дескриптори з великими значеннями інформативності у задачі редукції вважатимемо значущими, а малоінформативні дескриптори – кандидатами на виключення з простору описів.

Оскільки критерій працює шляхом порівняння відстаней дескрипторів, введемо поняття матриці відстаней M . Нехай $A = \{a_i\}_{i=1}^s$, $B = \{b_j\}_{j=1}^s$ – дві скінченні множини векторів, що мають розмір n , з однаковою розмірністю множин s , $A \subset R^n$, $B \subset R^n$, $\text{card } A = s$, $\text{card } B = s$, R^n – простір числових векторів розмірності n . Тоді матриця відстаней M має вигляд [13]

$$M[A, B] = \{\{m_{i,j}\}_{i=1}^s\}_{j=1}^s, \quad m_{i,j} = \rho(a_i, b_j), \quad (2.14)$$

де $\rho(a_i, b_j)$ – деяка підходяща метрика для застосування на векторах у просторі R^n , $a_i \in A$, $b_j \in B$. У матриці M , кожен i -й рядок відповідає i -му елементу з

множини опису і містить відстані від відповідного елемента множини A до кожного з векторів B . Розрахунок матриці дозволяє використовувати її для швидкого огляду значень відстані під час розрахунку різних етапів класифікації. Так, на основі матриць $M[E_k, E_k]$ та $M[E_k, E_j]$, $\forall j \neq k$ можна розрахувати значення інформативності для редукції описів еталонів, або застосувати матрицю при розрахунку медоїда у методі кластеризації k -медоїд.

На рисунку 2.7 зображена схема застосування принципу редукції опису на основі критерію інформативності у класифікаторі зображень.



Рисунок 2.7 – Схема класифікатора з застосуванням редукції опису

2.6 Метрики відстані у спрощеному просторі ознак

Розіб'ємо простір еталонних зображень E на множини центроїдів кластерів відповідно до формули (2.2). Ідентично проведемо редукцію простору описів вхідного зображення завдяки розбиттю на M кластерів та отриманню центроїдів бажаним методом кластеризації [14]

$$Z = T(Z) = \{T_k(Z)\}_{k=1}^M, T_k(Z) \cap T_j(Z) = \emptyset. \quad (2.15)$$

Розрахуємо кожну відстань $\rho(T_{k_1}(Z), T_{k_2}(E_i))$, $k_1, k_2 \in [1, 2, \dots, M]$ між кластерним поданням множин описів вхідного та еталонних зображень (сумарно M^2 значень) завдяки обчисленню відстаней $q = \rho(b_{k_1}(Z), b_{k_2}(E_i))$ між множиною центроїдами вхідного зображення та i -го еталону, що позначені у вигляді $\{q_a\}$, $a = 1, 2, \dots, M^2$.

Використовуючи значення множини $\{q_a\}$ можна знайти відстані між множинами спрощених описів еталонів та вхідного зображення. Для цього можна застосувати метрики, розраховані на знаходження відстані між множинами, такі як метрика Хаусдорфа, найближчого сусіда, чи середнього зв'язку. Для розрахунку відстані на основі множини $\{q_a\}$, метрика середнього зв'язку має вигляд [14]

$$\rho_{av}(T_k(Z), T_k(E_i)) = \frac{1}{M^2} \sum_{a=1}^{M^2} q_a. \quad (2.16)$$

Іншим варіантом розрахунку відстані між даними множинами є модифікація метрики (2.16), що полягає у розрахунку суми трьох найменших елементів з відсортованої за зростанням множини $\{q_a\}$, приведеної до вигляду $q^* = q_1 \leq q_2 \leq \dots \leq q_{M^2}$ [14]

$$\rho_{3m}(T_k(Z), T_k(E_i)) = \sum_{a=1}^3 q_a^*. \quad (2.17)$$

Також як варіант знаходження відстані між множинами спрощених описів зображень пропонується знаходження суми відстаней найближчого сусіда між кожним з центроїдів зображень [14]

$$\rho_{NN}(T_k(Z), T_k(E_i)) = \sum_{a=1}^M \min_{j=1, \dots, M} \rho(b_a(Z), b_j(E_i)). \quad (2.18)$$

Проблема метрики, описаної у формулі (2.18), полягає у тому, що існують випадки, у яких певні центри кластерів одного зображення можуть знаходити відповідні собі найближчі кластери іншого декілька разів, залишаючи при цьому певні центроїди незастосованими. Для знаходження оптимального розподілу центрів вхідного зображення до центрів еталонів, де кожному окремому центру вхідного зображення відповідає один центр еталонного, можна використовувати угорський метод.

Позначимо множини центрів кластерів для двох довільних множин Z_1, Z_2 дескрипторів зображень як b_1, b_2 . Під час застосування угорського методу формуються пари бінарних векторів з редукованих описів зображень з мінімальною сумарною вартістю для кожного з елементів двох множин. Значення вартості можна подати як суму відстаней між відповідними парами елементів множин b_1 та b_2 . Тоді зіставлення елементів відповідних множин є задачею оптимізації [14]

$$R(x) = \sum_{i=1}^M \sum_{j=1}^M \rho(b_{1i}, b_{2j}) x_{ij} \rightarrow \min, \quad (2.19)$$

де $b_{1i} \in b_1, b_{2j} \in b_2$;

x_{ij} – бінарний параметр, що дорівнює 1 при виявленні відповідності між i -м та j -м елементом.

При введенні обмеження на однозначність відповідності ознак (2.20), застосування угорського методу дозволяє знайти мінімальну загальну відстань (2.19) між множинами центрів зображень b_1 і b_2 [14]

$$\begin{cases} \sum_{j=1}^M x_{ij} = 1 \forall i = \overline{1, M}, \\ \sum_{i=1}^M x_{ij} = 1 \forall j = \overline{1, M}. \end{cases} \quad (2.20)$$

Застосування метрик (2.15-2.17) допомагає ефективно оцінити відстань між множинами спрощених описів простору дескрипторів у результаті

застосування редукції через методи кластеризації, чи використання методів з застосуванням критерію значимості. Використання угорського методу, при цьому, дозволяє сформувати оптимальний список відповідності між елементами спрощених описів для подальшого використання у класифікації.

2.7 Опис методів для тестування

У розділі 3 буде описано проведене експериментальне дослідження методів класифікації на основі критеріїв точності та швидкодії. З метою спрощення методи були позначені та будуть розглянуті у такому порядку:

– метод 1 – традиційний класифікатор на основі лінійного пошуку, що працює на основі локального по-елементного порівняння дескрипторів вхідного зображення з дескрипторами еталонних зображень з подальшим визначенням глобального класу за системою голосування;

– метод 2 – метод на основі кластеризації бази еталонних зображень, формування агрегованого подання об'єкту за набором кластерів, визначення класу об'єкта за мінімумом Манхетенської відстані між агрегованими поданнями об'єкту та еталонів;

– метод 3 – метод на основі накопичення повного вектора вагових коефіцієнтів класів. У методі кластеризується база еталонних зображень, проводиться попереднє голосування еталонів за центроїди з метою отримання матриці Φ ваг для центроїдів за класи. Після цього проводиться порівняння дескрипторів вхідного зображення з центрами кластерів та накопичуються відповідні ваги за класи з подальшим визначенням класу зображення на основі максимуму ваги;

– метод 4 – модифікація методу 3 на основі критерію однорідності (2.11), що модифікує матрицю голосів еталонів за кластери Φ , розраховуючи однорідність для кожного стовпчика матриці та відкидаючи вказану користувачем кількість стовпчиків на основі мінімальних значень однорідності;

– метод 5 – модифікація методу 3, що полягає у видаленні двох найменших значень ваги з кожного стовпчика матриці Φ ;

– метод 6 – модифікація методу 3, що полягає у видаленні чотирьох найменших значень ваги з кожного рядка матриці Φ .

Для оцінювання точності методів класифікації було введено критерій точності pr , що приймає значення від 0 до 1 (де 1 – правильна класифікація кожного зображення), та визначається як відношення кількості правильно визначених зображень q до загальної кількості проведених експериментів Q

$$pr = \frac{q}{Q}. \quad (2.21)$$

Для визначення зображень поза базою у методі 2 введено поріг на Манхеттенську відстань δ_{μ} . Зображення з мінімальною Манхеттенською відстанню до класу, більшою за попередньо визначене значення δ_{μ} , отримують відмову у визначенні мітки класу.

3 АНАЛІЗ РЕЗУЛЬТАТІВ КОМП'ЮТЕРНОГО МОДЕЛЮВАННЯ

3.1 Обґрунтування вибору програмного середовища

У рамках кваліфікаційної роботи був розроблений метод класифікації зображень на основі накопичення повного вектора вагових коефіцієнтів з використанням кластеризації, а також 3 модифікації даного методу. Також програмно реалізовані методи класифікації на основі лінійного пошуку та мінімуму Манхеттенської відстані. Для програмної реалізації функціоналу була використана бібліотека OpenCV, з використанням обгортки для мови програмування C# – OpenCVSharp [15].

OpenCV – одна з найвідоміших бібліотек комп'ютерного зору, яка містить широкий набір алгоритмів і функцій для задач комп'ютерного зору. OpenCV, розроблена компанією Intel, містить понад 2500 алгоритмів з області комп'ютерного зору, включно з детектором КТ BRISK, описаним у пункті 1.1. Можливості OpenCV охоплюють задачі виявлення та розпізнавання об'єктів, сегментацію зображень, пошук зображень за вмістом, трекінг, 3D-моделювання, а також виконання різноманітних перетворень над зображеннями та відео. Крім того, бібліотека підтримує паралельні обчислення та використання апаратного прискорення через CUDA і OpenCL, що значно покращує швидкодію при роботі з великими обсягами даних [16].

OpenCVSharp є однією з найпоширеніших обгорток OpenCV для C#. Вона максимально схоже відтворює синтаксис та структуру оригінального інтерфейсу, написаного на C++. Основними перевагами даної обгортки є підтримка інтерфейсу IDisposable для ефективного керування непотрібними ресурсами, а також здатність перетворювати матриці Mat, які часто генерує OpenCV своїми методами, у Bitmap і WriteableBitmap, що спрощує їх використання у C# коді.

C# – це сучасна об'єктно-орієнтована мова програмування, розроблена компанією Microsoft у рамках платформи .NET. Мова C# має чітку структуру та

зрозумілий синтаксис, що робить її зручною для розробки великих проєктів [17]. Для C# також доступні різні інструменти для роботи з даними (наприклад LINQ), що дозволяє виконувати прості та зрозумілі запити баз даних. Також ця мова добре інтегрована у середовище розробки Visual Studio (оскільки розробником також є Microsoft), що забезпечує зручне налагодження та тестування програмного коду.

Для реалізації програмної частини кваліфікаційної роботи використано мову C# через такі причини:

- код мови компілюється у проміжний код (Common Intermediate Language), який виконується у середовищі CLR (Common Language Runtime), що забезпечує автоматичне керування пам'яттю, збір сміття та обробку винятків, що спрощує написання коду;
- використання технологій .NET, під які розроблялась мова C# дозволяють легко створювати кросплатформові застосунки для Windows, Linux і macOS;
- синтаксис C# подібний до C++ та Java, що спрощує процес вивчення бібліотеки OpenCV, а також дозволяє легко інтегрувати код з цих мов у існуючу програму на C#;
- наявність зручного менеджера пакетів NuGet, який суттєво спрощує підключення та оновлення сторонніх бібліотек та відкидає необхідність у ручному налаштуванні шляхів і залежностей.

3.2 Дослідження та тестування розробленої моделі

Тестування розроблених методів проводились на двох базах зображень, що складались з восьми картин у кожному випадку. Для першої бази обрані плакати Нікити Тітова (база №1) [18], для другої – картини Івана Марчука (база №2) [19]. База еталонних зображень №1 продемонстрована на рисунку 3.1, еталонні зображення з бази №2 показані на рисунку 3.2.



Рисунок 3.1 – Еталонні зображення з бази №1



Рисунок 3.2 – Еталонні зображення з бази №2

Зображення подані у форматі jpg, розміром у 512×512 пікселів. Зображення бази №2 зменшені до розміру 350×350 та розміщені на сірому фоні з метою збереження більшої кількості ключових точок під час виконання геометричних

перетворень над зображеннями. На рисунку 3.3 продемонстроване одне з еталонних зображень з координатами ключових точок.



Рисунок 3.3 – Еталонне зображення з координатами ключових точок

Тестування виконувалось як із невидозміненими зображеннями, так і з застосуванням таких геометричних перетворень: поворот (± 20 градусів), масштаб ($\pm 10\%$) та зміщення (± 10 пікселів). Тобто в цілому протестовано 72 зображення на базу (8 комбінацій перетворень + 1 звичайне зображення).

Попередньо експериментально визначені пороги на мінімальну кількість голосів для двох баз даних для традиційного методу – 300 для бази №1 та 230 для бази №2.

Традиційний метод класифікації (метод 1) не допустив помилки у класифікації жодного зображення з обох баз. Додатково, метод продемонстрував високу надійність – відмінність у кількості голосів між еталонними зображеннями (навіть з геометричними перетвореннями) та зображеннями поза базою є доволі помітною, що дає методу велику гнучкість у виборі порогу. Результати класифікації еталонного зображення з бази №1 традиційним методом продемонстроване у таблиці 3.1. жирним шрифтом виділені значення зображень, що виграли голосування, червоним кольором позначені зображення, що отримали відповідний голос в результаті класифікації.

Таблиця 3.1 – Результати класифікації еталонного зображення з бази №1 під номером 3 традиційним методом

перетв.\клас	1	2	3	4	5
(0,0,0)	0	0	500	0	0
(20,1.1,10)	23	25	418	18	16
(20,1.1,-10)	15	22	424	20	19
(20,0.9,10)	31	43	386	30	10
(20,0.9,-10)	26	46	379	36	13
(-20,1.1,10)	26	22	416	24	12
(-20,1.1,-10)	25	32	401	21	21
(-20,0.9,10)	23	46	373	33	25
(-20,0.9,-10)	28	38	378	34	22

Як можна побачити з таблиці 3.1, жодне з зображень, навіть з застосуванням геометричних перетворень, не було відкинута. Зображення набрали велику кількість голосів за свій клас, значно перевищуючи поріг на мінімальну кількість голосів. На відміну від цього, зображення поза базою в загальному випадку набирають приблизно в 2 рази менше голосів за клас-переможець (табл. 3.2). Це вказує на ефективність досліджуваних методів.

Таблиця 3.2 – Результати класифікації зображення поза базою з бази №1 під номером 2 традиційним методом

перетв.\клас	1	2	3	4	5
(0,0,0)	72	173	114	71	70
(20,1.1,10)	80	167	113	81	59
(20,1.1,-10)	73	164	114	79	70
(20,0.9,10)	64	177	118	72	69
(20,0.9,-10)	70	175	104	81	70
(-20,1.1,10)	71	176	115	71	67
(-20,1.1,-10)	75	175	110	73	67
(-20,0.9,10)	68	173	118	83	58
(-20,0.9,-10)	63	170	113	89	65

Алгоритми, що використовують кластеризацію, протестовано з двома методами кластеризації – k -середніх та k -медоїд. В якості початкових центрів обрано кожен 175 та 375 дескриптор з вектора дескрипторів кожного з еталонів,

тобто всього – 10 кластерів. Для пришвидшення тестування, обидва методи обмежені п'ятьма ітераціями. Похибка обох методів для кожної з баз зображень зазначена у таблиці 3.3.

Таблиця 3.3 – Похибка кластеризації методів для двох баз зображень

	<i>k</i> -середніх	<i>k</i> -медоїд
База №1	0,327	0,361
База №2	0,329	0,364

Метод класифікації на основі мінімуму Манхетенської відстані (метод 2) між агрегованими поданнями об'єкту та еталонів продемонстрував відносно високий результат точності. Використовуючи метод *k*-середніх для отримання спрощеного опису еталонів, для бази №1 точність *pr* склала 0,96, для бази №2 – 0,75. Класифікатор не мав проблем з визначенням правильного класу, тобто мінімум Манхетенської відстані для еталонних зображень завжди відповідав правильному класу. У таблиці 3.4 продемонстровані результати класифікації еталонного зображення з бази №2.

Таблиця 3.4 – Результати класифікації еталонного зображення з бази №2 під номером 5 методом 2

перетв.\клас	1	2	3	4	5
(0,0,0)	245	267	161	277	0
(20,1.1,10)	292	276	226	322	107
(20,1.1,-10)	290	276	238	330	113
(20,0.9,10)	293	285	223	297	102
(20,0.9,-10)	300	290	238	308	111
(-20,1.1,10)	299	277	233	317	108
(-20,1.1,-10)	303	285	219	297	116
(-20,0.9,10)	286	294	244	332	105
(-20,0.9,-10)	301	299	263	339	112

Проблеми з класифікацією у методі в обох випадках з'явилися під час класифікації зображень поза базою – мінімум Манхетенської відстані виявлявся

надто схожим на значення еталонних зображень, що ускладнює вибір порогу. Приклад класифікації зображення поза базою методом приведено у таблиці 3.5.

Таблиця 3.5 – Результати класифікації зображення поза базою з бази №2 під номером 2 методом 2

перетв.\клас	1	2	3	4	5
(0,0,0)	262	238	194	136	217
(20,1.1,10)	273	249	171	113	220
(20,1.1,-10)	281	257	153	103	224
(20,0.9,10)	276	252	168	102	209
(20,0.9,-10)	281	257	171	97	218
(-20,1.1,10)	320	296	158	94	265
(-20,1.1,-10)	303	279	163	95	258
(-20,0.9,10)	290	266	178	104	217
(-20,0.9,-10)	292	268	168	92	213

З таблиці 3.5 можна побачити, що лише зображення без геометричних перетворень було розпізнане як зображення поза базою. Значення мінімуму сильно схожі і, навіть, у деяких випадках, менші у зображення поза базою. Для обох баз зображень визначення порогу на мінімум Манхеттенської відстані для відкидання зображень поза базою виявилось проблематичним, що помітно знизило якість класифікації.

Метод k -медоїд показав ідентичну проблему – надто близькі значення Манхеттенської відстані для класів-переможців еталонних зображень та зображень поза базою. Також одне з еталонних зображень з геометричними перетвореннями отримало неправильну мітку класу. Результат точності при цьому знизився до 0,88 для першої бази та 0,7 для другої, що підкріплюється тим, що k -медоїд згрупував дані помітно гірше, виходячи зі значення похибки кластеризації (табл. 3.3).

Перед розглядом методу на основі накопичення повного вектора вагових коефіцієнтів (метод 3), розглянемо вигляд згенерованих методом k -середніх матриць Φ голосів еталонів за центроїди (табл. 3.6, 3.7). Жирним шрифтом в таблицях виділено найбільшу вагу у кожному стовпчику.

Таблиця 3.6 – Подання еталонів бази 1 через кластери методом k -середніх

Еталон/го лоси	Клас тер 1	Клас тер 2	Клас тер 3	Клас тер 4	Клас тер 5	Клас тер 6	Клас тер 7	Клас тер 8	Клас тер 9	Клас тер 10	Сума
Еталон 1	56	61	32	82	54	22	62	32	73	26	500
Еталон 2	37	37	82	74	28	25	90	37	45	45	500
Еталон 3	45	43	118	34	30	65	25	59	25	56	500
Еталон 4	135	19	40	37	49	49	40	44	54	33	500
Еталон 5	61	47	30	47	47	15	49	31	134	39	500

Таблиця 3.7 – Подання еталонів бази 2 через кластери методом k -середніх

Еталон/го лоси	Клас тер 1	Клас тер 2	Клас тер 3	Клас тер 4	Клас тер 5	Клас тер 6	Клас тер 7	Клас тер 8	Клас тер 9	Клас тер 10	Сума
Еталон 1	47	28	52	53	48	54	41	12	34	131	500
Еталон 2	77	17	58	61	28	90	46	11	50	62	500
Еталон 3	23	47	43	29	123	35	74	33	25	68	500
Еталон 4	16	91	19	23	128	22	14	102	18	67	500
Еталон 5	42	35	31	41	58	27	98	38	59	71	500

Для візуалізації значень ваги за кластерами, на рисунку 3.4 зображена діаграма кластерного подання одного з еталонів, де чітко видно сильний приріст ваги кластера при віднесенні дескриптора до центроїда третього кластера.



Рисунок 3.4 – Кластерне подання еталону 3 з бази 1

З таблиці 3.6 та 3.7 видно, що майже кожне еталонне зображення має хоча б один кластер, який надає велику вагу у сторону відповідного класу (наприклад, кластер 3 для еталону 3 з бази 1). Еталон 3 з бази 2 має кластери з великою перевагою за свій клас, але не виграє в жодному з них, що не є бажаним, проте і не є критичною проблемою. Деякі варіанти модифікації матриці Φ можуть спричинити складності в визначенні зображень, що належать до цього класу.

Для методу 3 експериментально визначено поріг δ у 10% на максимально допустиме відхилення ваги від ваги W_k^E еталону за свій клас для матриці Φ , отриманої методом k -середніх. Для цього експерименту та подальших експериментів з методом 3, поріг визначався у діапазоні мінімально допустимий поріг для проходження усіх еталонних зображень $\pm 5\%$. Значення ваги W_k^E для класів зазначені у таблиці 3.8.

Таблиця 3.8 – Значення ваг еталонів за власний клас для двох баз

Еталон/база	База 1	База 2
Еталон 1	28878	33868
Еталон 2	29866	30768
Еталон 3	31946	33596
Еталон 4	33898	41708
Еталон 5	34312	29294

З таблиці 3.8 одразу видно, що еталони, які мають кластери з великою вагою, та не мають інших великих значень у відповідній колонці матриці Φ (наприклад еталон 4 з бази 2), набирають значно більші ваги, ніж еталони з відносно рівномірно розподіленими вагами (наприклад еталон 1 з бази 1). Використовуючи матриці з таблиць 3.6 та 3.7, метод на основі накопичення повного вектора вагових коефіцієнтів показав стовідсоткову точність для обох баз зображень. У таблиці 3.9 наведено результат класифікації еталонного зображення з першої бази зображень.

Таблиця 3.9 – Результат класифікації еталонного зображення з бази №1 під номером 4 традиційним методом

перетв.\клас	1	2	3	4	5
(0,0,0)	25445	23456	24319	33898	26952
(20,1.1,10)	25842	23720	24107	32192	26875
(20,1.1,-10)	25920	23479	23410	31557	27930
(20,0.9,10)	25103	23115	24954	30835	26558
(20,0.9,-10)	25174	23043	24716	30603	26807
(-20,1.1,10)	26340	23481	22884	31437	28239
(-20,1.1,-10)	25855	22805	23774	32708	28103
(-20,0.9,10)	25266	22673	24360	31716	27162
(-20,0.9,-10)	24735	22528	24566	32025	27021

Як можна побачити з таблиці 3.9, зображення мають помітно більшу вагу за власний клас, ніж за будь-які інші. Значення ваги переможця є доволі близькими, що свідчить про якість дескрипторів вхідних зображень під впливом геометричних перетворень. Враховуючи значення з таблиці 3.8, допустима вага цього класу знаходиться в діапазоні [30,508; 37,288]. Вага двох зображень наближуються до цього значення, проте все ще знаходяться у необхідному діапазоні. Для протиставлення, розглянемо зображення поза базою, яке теж мало великі ваги за цей клас (табл. 3.10).

Таблиця 3.10 – Результат класифікації зображення поза базою з бази №1 під номером 1 методом 3

перетв.\клас	1	2	3	4	5
(0,0,0)	24706	22720	25759	26467	25527
(20,1.1,10)	24891	22148	24939	28131	25818
(20,1.1,-10)	25135	22131	24144	27321	26248
(20,0.9,10)	25154	21862	24547	27196	26378
(20,0.9,-10)	25244	21782	24649	27417	26902
(-20,1.1,10)	24823	22162	25039	27733	25965
(-20,1.1,-10)	25133	22747	25331	28393	26224
(-20,0.9,10)	25046	21940	24831	27246	26875
(-20,0.9,-10)	25192	22154	24781	27593	26633

Як можна побачити з таблиці 3.10, більшість зображень були віднесені до четвертого класу, проте не отримали мітку класу, оскільки у більшості випадків їхні значення знаходяться відносно далеко від мінімально допустимої ваги у 30508 для четвертого класу, хоча в деяких випадках наближуються до нього. Не зважаючи на це, як мінімум для цих зображень, метод демонструє можливість невеликого коригування порогу класифікації зі збереженням точності. Розглянемо вплив відносно рівномірного розподілу голосів за кластерами на класифікацію на прикладі еталонного зображення 1 з бази 1 (табл. 3.11).

Таблиця 3.11 – Результат класифікації еталонного зображення з бази №1 під номером 1 методом 3

перетв.\клас	1	2	3	4	5
(0,0,0)	28878	26302	21476	25445	28791
(20,1.1,10)	28458	26302	21996	25859	27238
(20,1.1,-10)	28193	25980	22311	26822	26986
(20,0.9,10)	27991	25682	22899	26952	26922
(20,0.9,-10)	28021	25773	22667	26309	26817
(-20,1.1,10)	28106	26525	23131	25697	25701
(-20,1.1,-10)	28429	26614	22583	25204	25967
(-20,0.9,10)	28090	25999	22904	27096	26722
(-20,0.9,-10)	28094	25796	22531	27612	26833

З таблиці 3.6 видно, що зображення за один крок не може набрати значення ваги більше 81, що відображається кількісно на сумарній вазі під час класифікації (табл. 3.11): у порівнянні зі значеннями ваги еталону 4 з таблиці 3.9, ваги першого еталонного зображення значно менші та дуже близькі до ваги, яку зображення отримали за клас 5. При цьому значення ваги зображень з геометричними перетвореннями сильно схожі на значення без перетворень, що підтверджує можливість методу ставити більш строгий поріг на відкидання зображень поза базою.

Використання методу k -медоїд для генерації матриці Φ голосів еталонів за кластери, ідентично до випадку з методом 2, призвело до сильних втрат у точності: $pr = 0,6$ для першої бази та $pr = 0,68$ для другої. Для з'ясування причини

зниження якості класифікації, наведено таблицю 3.12, де продемонстрована матриця Φ для другої бази зображень.

Таблиця 3.12 – Подання еталонів бази 2 через кластери методом k -медоїд

Еталон/го лоси	Клас тер 1	Клас тер 2	Клас тер 3	Клас тер 4	Клас тер 5	Клас тер 6	Клас тер 7	Клас тер 8	Клас тер 9	Клас тер 10	Сума
Еталон 1	29	21	57	48	85	43	79	13	25	100	500
Еталон 2	62	16	41	59	47	65	54	14	54	88	500
Еталон 3	14	31	35	25	152	23	121	18	18	63	500
Еталон 4	13	64	25	26	114	24	81	85	11	57	500
Еталон 5	24	22	35	53	88	25	145	18	36	54	500

З таблиці 3.12 видно, що матриця містить багато однорідних стовпців, а також дуже великі ваги для третього еталону у декількох кластерах. Також ваги еталонів 1 та 2 доволі рівномірно розподілені за кластерами, що знижує їх здатність набирати достатню перевагу для отримання власної мітки в результаті класифікації. Це можна підтвердити тим, що еталон 1 не був правильно ідентифікованим (табл. 3.13).

Таблиця 3.13 – Результат класифікації еталонного зображення з бази №1 під номером 1 методом 3

перетв.\клас	1	2	3	4	5
(0,0,0)	32944	28691	34704	28595	32241
(20,1.1,10)	32642	27881	37405	30333	34548
(20,1.1,-10)	32675	28081	36777	29843	34985
(20,0.9,10)	31025	27286	34034	29199	31991
(20,0.9,-10)	31298	27557	33632	28799	31575
(-20,1.1,10)	32664	28256	35266	29123	33298
(-20,1.1,-10)	32946	28416	35686	29395	33795
(-20,0.9,10)	31964	27720	36921	30199	34886
(-20,0.9,-10)	31299	27311	35502	29557	34210

Як бачимо з таблиці 3.13, метод надав перевагу третьому класу через його великі ваги, а також низьку концентрацію за стовпцями значень ваги для першого класу. Значення третього класу перевищують значення першого

приблизно на 2000-5000. Навіть п'ятий клас має перевагу над першим у цьому випадку, що демонструє повну неможливість класифікатора обрати перший клас, маючи таку вхідну матрицю голосів еталонів за кластери. Слід також зазначити, що при зміні початкових центрів для методу k -медоїд, точність класифікації для першої бази зображень зростає до 0,85. Це демонструє факт сильної залежності алгоритму від якості та обраного методу кластеризації, а для методів k -середніх та k -медоїд – важливості вибору правильних початкових центрів для кластеризації.

Модифікація методу 3 зі скороченням числа аналізованих центрів кластерів через попереднє логічне оброблення даних у стовпці за критерієм однорідності (метод 4) показала високі результати точності pr – 0,99 та 0,97. При цьому значення похибки були збільшені до 12% за результатами попереднього експериментування з методом, визначенні ідентично до методу 3. Значення однорідності, отримані з матриці Φ для першої бази зображень (табл. 3.5), а також модифікована матриця наведені в таблицях 3.14 та 3.15.

Таблиця 3.14 – значення однорідності для стовпців матриці Φ для бази №1

Однорідність /стовпець	Стовпець 1	Стовпець 2	Стовпець 3	Стовпець 4	Стовпець 5	Стовпець 6	Стовпець 7	Стовпець 8	Стовпець 9	Стовпець 10
	424	188	452	266	142	254	304	136	492	144

Таблиця 3.15 – Модифікована матриця Φ для бази №1

Еталон/голови	Клас тер 1	Клас тер 2	Клас тер 3	Клас тер 4	Клас тер 5	Клас тер 6	Клас тер 7	Клас тер 8	Клас тер 9	Клас тер 10	Сума
Еталон 1	56	61	32	82	0	22	62	0	73	0	388
Еталон 2	37	37	82	74	0	25	90	0	45	0	390
Еталон 3	45	43	118	34	0	65	25	0	25	0	355
Еталон 4	135	19	40	37	0	49	40	0	54	0	374
Еталон 5	61	47	30	47	0	15	49	0	134	0	383

На основі значень однорідності з таблиці 3.14 прийнято рішення видалити 3 стовпчики, помічені червоним кольором, оскільки вони мають найменші значення однорідності. Подальше видалення стовпчиків з матриці у таблиці 3.14

може призвести до катастрофічних втрат точності, оскільки стовпчики, що залишилися в таблиці, візуально та за параметром однорідності, є доволі інформативними. До того ж, як бачимо у таблиці 3.15, еталони мають приблизно однакову сумарну кількість голосів, тобто дана модифікація виконала свою ціль – було видалено стовпці, що не надають значної переваги жодному з класів.

Помилки у класифікації з цією модифікацією виникли під час класифікації одного з еталонних зображень з першої бази, де йому було відмовлено у класифікації через надто низьку вагу. Під час класифікації зображень з другої бази, метод помилився у визначенні мітки одного з еталонних зображень, а також не відмовив у класифікації одному з зображень поза базою. У таблиці 3.16 наведені результати класифікації еталонного зображення з бази 1, у якому було допущено помилку.

Таблиця 3.16 – Результат класифікації еталонного зображення з бази №1 під номером 4 методом 4

перетв.\клас	1	2	3	4	5
(0,0,0)	20533	18971	18405	28472	21998
(20,1.1,10)	20902	19252	18219	26750	21913
(20,1.1,-10)	20680	18555	16859	25691	22603
(20,0.9,10)	20129	18533	18963	25360	21494
(20,0.9,-10)	19822	18226	18392	24729	21431
(-20,1.1,10)	21312	19010	16989	25919	23230
(-20,1.1,-10)	21255	18721	18406	27672	23515
(-20,0.9,10)	19856	17774	17921	25765	21716
(-20,0.9,-10)	19175	17312	17671	25831	21350

З таблиці 3.16 видно, що еталонне зображення без впливу геометричних перетворень було найбільш впевненим у своєму класі, проте одне з зображень з геометричними перетвореннями отримало надто низьку вагу – 24729, відхилившись від значення W_k^E власного класу (28472) приблизно на 14%.

Розглядати метод k -медоїд при розгляді модифікації методу 3 на основі критерію однорідності не має сенсу, оскільки ця модифікація має на меті

прибирання цілих неінформативних стовпчиків, що не може вирішити проблему пов'язану з тим, що деякі класи мають стовпчики з надто великими вагами.

Модифікація методу 3 з відкиданням k найменш значущих ваг зі стовпчиків матриці Φ (метод 5) продемонстрував зменшення у точності у порівнянні з немодифікованим методом 3. Значення точності pr при цьому залишаються доволі високими – 0,93 для першої бази зображень та 0,86 для другої. Під час класифікації методом 5, для бази 1 була допущена помилка при визначенні одного еталонного зображення, а також 4 з 25 зображень поза базою. Для другої бази помилки виникли під час визначення мітки 5 з 45 еталонних та 5 з 25 зображень поза базою. У таблиці 3.17 наведено результати класифікації одного з неправильно класифікованих еталонних зображень з другої бази методом 5.

Таблиця 3.17 – Результат класифікації еталонного зображення з бази №2 під номером 2 методом 5

перетв.\клас	1	2	3	4	5
(0,0,0)	24550	25730	17870	6253	20232
(20,1.1,10)	23633	24103	17471	8235	19834
(20,1.1,-10)	24522	22653	18535	9020	20860
(20,0.9,10)	25024	24006	17220	7058	21452
(20,0.9,-10)	24336	24611	16360	6667	21295
(-20,1.1,10)	24619	24050	17466	7420	20807
(-20,1.1,-10)	24728	24149	17599	7259	20441
(-20,0.9,10)	23355	25082	16459	6693	21245
(-20,0.9,-10)	24391	24477	17143	6953	21063

Як бачимо, проблема виникла через те, що класифікатор набрав більшу вагу за перший клас, ніж на потрібний другий. Всі інші помилки були пов'язані з порогом (відмова у наданні мітки класу еталонному зображенню, або надання мітки зображенню поза базою). Для визначення проблеми, що виникла, у таблиці 3.18 наведено розподіл ваг за кластерами для цієї бази після застосування модифікації матриці методом 5.

Таблиця 3.18 – Подання еталонів бази 2 через кластери після модифікації матриці Φ методом 5

Еталон/го лоси	Клас тер 1	Клас тер 2	Клас тер 3	Клас тер 4	Клас тер 5	Клас тер 6	Клас тер 7	Клас тер 8	Клас тер 9	Клас тер 10	Сума
Еталон 1	47	0	52	53	0	54	0	0	34	131	371
Еталон 2	77	0	58	61	0	90	46	0	50	0	382
Еталон 3	0	47	43	0	123	35	74	33	0	68	423
Еталон 4	0	91	0	0	128	0	0	102	0	0	321
Еталон 5	42	35	0	41	58	0	98	38	59	71	442

Як можна побачити з таблиці 3.18, кластер 10 надає велику вагу в сторону першого класу, не даючи жодної ваги другому. До видалення значень, еталон 2 мав доволі високу вагу у цьому кластері – 62. Вірогідно, що під впливом геометричних перетворень деякі зображення з цього класу набрали більше голосів за десятий кластер, чим сильно знизили якість класифікації. Також стає помітною вада методу: оскільки деякі стовпці можуть мати декілька великих ваг через те, що на етапі створення матриці Φ голоси еталонів розподіляються по стовпцях (500 дескрипторів з кожного ряду розбиваються за стовпцями), то найменші ваги у стовпчику не є неважливими у глобальному для алгоритму сенсі.

Модифікація методу 3 з логічним обробленням вектора вагових коефіцієнтів у рядку (метод 6) значно доцільніше підходить до відкидання незначущих ваг з матриці Φ ніж метод 5, що відобразилось на результатах точності. Класифікатор продемонстрував показник точності $pr = 1$ для першої та $pr = 0,96$ для другої бази зображень. Помилки під час класифікації зображень другої бази з'явилися під час визначення 3 з 45 еталонних зображень. 3 зображення з геометричними перетвореннями, що належали до 5 класу, не були допущені до отримання мітки класу через ваги, що перевищили допустимий поріг відмінності.

У таблиці 3.19 зображений вигляд матриці Φ після застосування модифікації з методу 6.

Таблиця 3.19 – Подання еталонів бази 2 через кластери після модифікації матриці Φ методом 6

Еталон/го лоси	Клас тер 1	Клас тер 2	Клас тер 3	Клас тер 4	Клас тер 5	Клас тер 6	Клас тер 7	Клас тер 8	Клас тер 9	Клас тер 10	Сума
Еталон 1	47	0	52	53	48	54	0	0	0	131	385
Еталон 2	77	0	58	61	0	90	0	0	50	62	398
Еталон 3	0	47	43	0	123	35	74	0	0	68	390
Еталон 4	0	91	0	23	128	22	0	102	0	67	433
Еталон 5	42	0	0	41	58	0	98	0	59	71	369

З таблиці 3.19 видно, що ваги більш рівномірно розподілені у порівнянні з методом 5, проте у цьому випадку клас 5 має помітно меншу суму ваги, через що і виникли проблеми під час класифікації еталонних зображень, що належали до цього класу.

Значення точності класифікації досліджуваних методів зображені на рисунку 3.5.

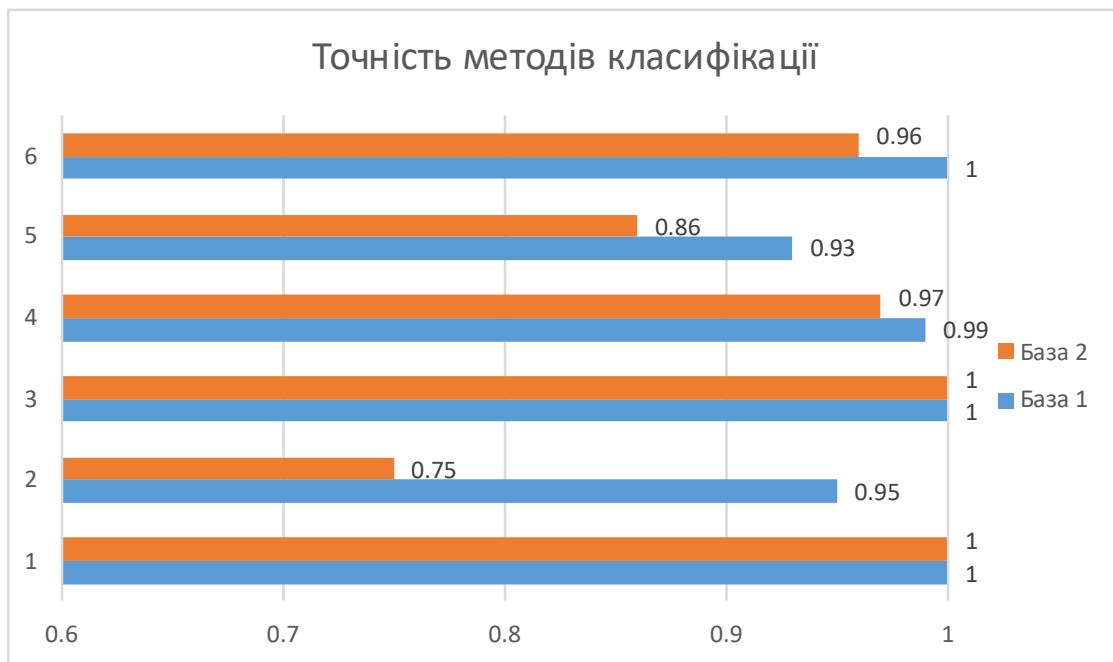


Рисунок 3.5 – Експериментальні показники точності класифікації

У результаті дослідження, традиційний метод та метод на основі накопичення повного вектора вагових коефіцієнтів виявилися найточнішими з

точністю класифікації рівною 1. При цьому останній виграв у швидкодії приблизно у 223 рази – 19,4 мс у порівнянні з 4366,9 мс. При цьому перевага традиційного методу полягає виключно у більшій надійності та простоті визначення порогу класифікації. Метод на основі мінімуму Манхеттенської відстані виявився ненадійним через складність підбору порогу класифікації, що виникає через схожість результатів відстані еталонних зображень та зображень поза базою. Це відзначилося на результатах, де на першій базі зображень метод майже повністю правильно ідентифікував зображення, проте на другій показав сильне падіння точності до $pr = 0,75$. Через це метод на основі мінімальної Манхеттенської відстані сильно програє методу на основі накопичення повного вектора вагових коефіцієнтів. При цьому метод зберігає високу швидкодію – 19,7 мс. Модифікація методу 3, що зберігає лише найбільші ваги за стовпчиками виявилась неефективною на обох тестових базах даних – помічене зниження точності методу (до $pr = 0,93$ та $pr = 0,86$) без жодних вигравів у швидкодії. При цьому модифікація з виділенням 4 незначущих ваг з рядків є значно більш доцільною, незважаючи на незначні втрати у точності класифікації ($pr = 1$ та $pr = 0,96$). Модифікація методу 3 зі скороченням числа аналізованих центрів кластерів через попереднє логічне оброблення даних у стовпці за критерієм однорідності продемонструвала незначне зниження у точності ($pr = 0,99$ та $pr = 0,97$ для першої та другої бази відповідно), але з підвищенням швидкості виконання класифікації в залежності від кількості відкинутих стовпчиків матриці Φ (20-30% від значення методу 3 на використаних базах зображень). При цьому метод має можливість підвищувати точність класифікації на деяких базах зображень. До того ж, оскільки кількість видалених стовпців матриці Φ можна редагувати, модифікацію можна використовувати для збереження ще більшої кількості часу, в залежності від бажаного значення точності класифікації.

Під час класифікації був заміряний середній час виконання класифікації зображень кожним методом. Класифікація виконувалась на комп'ютері з процесором Intel Core I5-12600K (3.7GHz) та 32 Гб оперативної пам'яті.

Середній час виконання традиційного методу склав 4366,9 мс. Метод на основі мінімуму Манхетенської відстані та метод на основі накопичення повного вектора вагових коефіцієнтів з двома його модифікаціями з видаленням найменш значущих ваг зі стовпців та рядків продемонстрували виконали класифікацію за майже ідентичний час – 19,7, 19,4, 19,5 та 19,7 мс. Модифікація методу з ваговими коефіцієнтами на основі критерію однорідності показала швидкість у 14,5 мс та 15,9 мс при видаленні 3 та 2 стовпчиків відповідно. Кожен з методів, що використовував кластеризацію (методи 2-7) досягли сильно помітного збільшення швидкодії у порівнянні з методом 1, що класифікує зображення за принципом лінійного пошуку.

Результати точності та швидкодії класифікації кожним з методів для обох баз зображень занесені до таблиці 3.20.

Таблиця 3.20 – Точність та швидкодія методів класифікації (метод кластеризації – k -середніх)

Метод		1	2	3	4	5	6
Точність	База 1	1	0,95	1	0,99	0,93	1
	База 2	1	0,75	1	0,97	0,86	0,96
Час, мс	База 1	4366,9	19,7	19,4	14,5	19,5	19,7
	База 2				15,9		

При цьому метод кластеризації k -середніх виявився більш точним у порівнянні з методом k -медоїд для застосування у досліджуваних методах класифікації. У кожному з виконаних експериментів, використання матриці Φ , згенерованої k -медоїд, призводило до зниження якості класифікації. Оскільки медоїди є найближчим реальним елементом з кластера до його елементів (а не реальним центром), вони знаходяться на більшій середній відстані ніж центри k -середніх, тому їх використання з метою репрезентації кластерів є менш ефективним для даної задачі. Хоч у кожному з методів, що використовують кластеризацію, вона йде як етап попередньої обробки, слід також зазначити, що метод k -медоїд має значно гіршу швидкодію у порівнянні з k -середніх через

значно складніший процес визначення опорного елементу кластера. Швидкодії методів кластеризації зазначені на рисунку 3.6.



Рисунок 3.6 – Час виконання методів кластеризації

ВИСНОВКИ

У кваліфікаційній роботі досліджено методи класифікації зображень на основі вектора вагових коефіцієнтів для кластерного подання даних.

Розроблено метод класифікації зображень на основі накопичення повного вектора вагових коефіцієнтів, що виконано за принципом визначення ваги класів через попередню кластеризацію з подальшим розподілом голосів еталонів за центроїди.

Розроблено три удосконалення пропонуваного методу на основі накопичення повного вектора вагових коефіцієнтів з використанням критерію однорідності, а також видалення незначущих значень зі стовпчиків та рядків з матриці Φ вагових коефіцієнтів.

Метод протестовано з різними варіантами кластеризації та модифікаціями з метою визначення доцільності його використання у порівнянні з традиційним методом, а також методом на основі мінімуму Манхеттенської відстані на основі точності та швидкості виконання.

В результаті дослідження виявлено, що метод на основі накопичення повного вектора вагових коефіцієнтів дає сильне прискорення у порівнянні з традиційним методом, та має майже ідентичну швидкодію до методу на основі мінімуму Манхеттенської відстані, при цьому демонструючи високі показники точності. Модифікації методу на основі видалення значень з рядків та стовпців показали зниження точності, при чому видалення ваг з рядків є значно більш ефективним за видалення зі стовпчиків. Модифікація на основі критерію однорідності показала незначне падіння точності класифікації, підвищивши швидкодію ще більше ніж немодифікований метод.

Новизною дослідження є розроблення методу класифікації зображень, націленого на швидкодію з використанням вектора вагових коефіцієнтів, отриманих спрощеним поданням еталонів через кластеризацію, а також створення трьох модифікацій розробленого методу для пришвидшення чи

підвищення точності методу з подальшим порівнянням з методом на основі лінійного пошуку та методом на основі мінімуму Манхеттенської відстані.

Перспектива подальшого розвитку методів може бути пов'язана з розглядом впливу інших методів кластеризації на розроблені методи, дослідження інших способів модифікації матриці Φ для збільшення завадостійкості класифікатора, а також оцінювання результативності методів на більшій базі зображень.

Результати роботи апробовано у вигляді статті [10] та тез доповідей під час Міжнародного молодіжного форуму «Радіоелектроніка і молодь у XXI столітті» [22] та VIII Міжнародної науково-практичної конференції «Academic research by scientists in the field of modern technologies» [23].

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Szeliski, R. (2022). *Computer vision: algorithms and applications*. Springer Nature.
2. Ponnusamy, R., Sathyamoorthy, S., & Manikandan, K. (2017). A review of image classification approaches and techniques. *International Journal of Recent Trends in Engineering & Research*, 3(3), 1-5.
3. Leutenegger, S., Chli, M., & Siegwart, R. Y. (2011, November). BRISK: Binary robust invariant scalable keypoints. In *2011 International conference on computer vision* (pp. 2548-2555). Ieee.
4. Гороховатський, В. О., & Творошенко, І. С. (2022). Аналіз багатовимірних даних за описом у формі множини компонент.
5. Jain, A. K., Murty, M. N., & Flynn, P. J. (1999). Data clustering: a review. *ACM computing surveys (CSUR)*, 31(3), 264-323.
6. Jain, A. K. (2010). Data clustering: 50 years beyond K-means. *Pattern recognition letters*, 31(8), 651-666.
7. Ester, M., Kriegel, H. P., Sander, J., & Xu, X. (1996, August). A densitybased algorithm for discovering clusters in large spatial databases with noise. In *kdd* (Vol. 96, No. 34, pp. 226-231).
8. Clustering algorithms. URL: <https://developers.google.com/machine-learning/clustering/clustering-algorithms> (дата звернення 18.10.2025).
9. Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 779-788).
10. Gorokhovatskyi, V., Stiahlyk, N., Mazur, Y., & Vechirska, A. (2024). Способи метричної грануляції для опису зображень у задачі класифікації. *Системи управління, навігації та зв'язку. Збірник наукових праць*, 3(77), 106-112.

11. Lund, B., & Ma, J. (2021). A review of cluster analysis techniques and their uses in library and information science research: k-means and k-medoids clustering. *Performance Measurement and Metrics*, 22(3), 161-173.

12. Gorokhovatskyi, V., Vlasenko, N. (2021). Редукція опису зображення у складі множини дескрипторів на основі метричного критерію інформативності. *Advanced Information Systems*, 5(4), 10-16.

13. Gorokhovatskyi V., Tvoroshenko I., Yakovleva O. (2024) Transforming image descriptions as a set of descriptors to construct classification features, *Indonesian Journal of Electrical Engineering and Computer Science*, 33 (1), 113-125.

14. M. A. Ahmad, V. Gorokhovatskyi, I. Tvoroshenko, N. Vlasenko, S. K. Mustafa (2021) The Research of Image Classification Methods Based on the Introducing Cluster Representation Parameters for the Structural Description, *International Journal of Engineering Trends and Technology*, 69(10), pp. 186-192, DOI: 10.14445/22315381/IJETT-V69I10P223.

15. OpenCV wrapper for .NET. URL: <https://github.com/shimat/opencvsharp> (дата звернення 30.10.2025).

16. Документація OpenCV. URL: <https://docs.opencv.org/> (дата звернення 30.10.2025).

17. C# Language Documentation. URL: <https://learn.microsoft.com/en-us/dotnet/csharp/> (дата звернення 30.10.2025).

18. Нікіта Тітов. URL: <https://www.facebook.com/nikitavltitov/> (дата звернення 01.11.2025).

19. Марчук Іван Степанович. URL: <https://www.wikiart.org/uk/marchuk-ivan-stepanovich> (дата звернення 01.11.2025).

20. Bohdan N., Tvoroshenko I., Gorokhovatskyi V., and Kobylin O. (2025) Development of a hybrid method to enhance context memory for a chatbot application based on large language models, *International Journal of Academic Information Systems Research*, 9(10), pp. 7-18.

21. Suprun A., Tvoroshenko I., Gorokhovatskyi V., and Yakovleva O. (2025) Development and research of a method for the combined use of large language models

for text generation, *International Journal of Academic and Applied Research*, 9(10), pp. 249-263.

22. Мазур, Є. В. (2025). Класифікація зображень на основі квантування ознак, *Радіоелектроніка та молодь у XXI столітті: тези доповідей 29-го Міжнародного молодіжного форуму*. Харків: ХНУРЕ, 2025, Т. 7, pp. 86-88.

23. Мазур, Є. В. (2025). Швидкісні методи класифікації зображень з використанням квантування ознак, *The 8th International scientific and practical conference "Academic research by scientists in the field of modern technologies" (October 21-24, 2025) Milan, Italy. International Science Group. 2025.* pp. 38-40.

24. Gadetska S., Gorokhovatskyi V., Stiahlyk N., Vlasenko N. (2022) Aggregate Parametric Representation of Image Structural Description in Statistical Classification Methods. In CEUR Workshop Proceedings: Computer Modeling and Intelligent Systems (CMIS-2022), 3137, pp. 68-77.

25. Гороховатський В.О., Гадецька С.В., Стяглик Н.І. (2019) Вивчення статистичних властивостей моделі блочного подання для множини дескрипторів ключових точок зображень. *Радіоелектроніка, інформатика, управління*, №2, 100–107.

26. Gorokhovatsky V.A. Efficient Estimation of Visual Object Relevance during Recognition through their Vector Descriptions. *Telecommunications and Radio Engineering*. – 2016, Vol. 75, No 14. – P. 1271–1283.

27. Gorokhovatskyi, V. A. (2003). Recognition of images in conditions of incomplete information. *KNURE*, Kharkov.

28. Gorokhovatskyi, V., Chmutov, Y., Tvoroshenko, I., & Kobylin, O. (2025). Reducing computational costs by compressing the structural description in image classification methods. *Advanced Information Systems*, 9(1), 5–12. <https://doi.org/10.20998/2522-9052.2025.1.01>

29. Gorokhovatskyi V., Tvoroshenko I., Yakovleva O., and Hudáková M. (2025) Image description compression in classification structural methods, *IEEE Access*, vol. 13, pp. 43631-43641, doi: 10.1109/ACCESS.2025.3548910.

30. Gadetska, S.V., Gorokhovatskyi, V. O., Stiahlyk, N. I., Vlasenko, N.V. Statistical data analysis tools in image classification methods based on the description as a set of binary descriptors of key points. *Radio Electronics, Computer Science, Control*, 2021, №4, 58-68.

31. Gorokhovatskyi, O., Peredrii, O., Gorokhovatskyi, V., Vlasenko, N. (2023) Explanation of CNN Image Classifiers with Hiding Parts. In: J. Benois-Pineau, R. Bourqui, D. Petkovic, G. Quenot (eds), *Explainable Deep Learning Artificial Intelligence*, pp. 125-146, Academic Press, 346 p.

32. Gorokhovatskyi, V., Gadetska, S., & Stiahlyk, N. (2023). Accelerating Image Classification based on a Model for Estimating Descriptor-to-Class Distance. *International Journal of Computing*, 22(4), 485-492.

33. Daradkeh Y.I., Gorokhovatskyi V., Tvoroshenko I., Gadetska S., and Al-Dhaifallah M. (2023) Statistical data analysis models for determining the relevance of structural image descriptions, *IEEE Access*, 11, 126938-126949.

34. Gorokhovatskyi V., Tvoroshenko I., Yakovleva O., Hudáková M., and Gorokhovatskyi O. (2024) Application a committee of Kohonen neural networks to training of image classifier based on description of descriptors set, *IEEE Access*, vol. 12, 73376-73385.

35. Gorokhovatskyi V., Gadetska S., Stiahlyk N. (2020) Image structural classification technologies based on statistical analysis of descriptions in the form of bit descriptor set. In *CEUR Workshop Proceedings: Computer Modeling and Intelligent Systems (CMIS-2020)*, 2608, 1027-1039.

36. Daradkeh Y.I., Gorokhovatskyi V., Tvoroshenko I., and Zeghid M. (2024) Improving the effectiveness of image classification structural methods by compressing the description according to the information content criterion, *Computers, Materials & Continua*, vol. 80, no. 2, 3085-3106.

37. Tvoroshenko I., Pomazan V., Gorokhovatskyi V., and Kobylin O. (2023) Application of video data classification models using convolutional neural networks, *International Journal of Academic and Applied Research*, 7(11), pp. 134-145.