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 перший (бакалаврський)

РОЗРОБЛЕННЯ МЕТОДУ КЛАСИФІКАЦІЇ ЗОБРАЖЕНЬ ІЗ ЗАСТОСУВАННЯМ МАТРИЦІ ВІДСТАНЕЙ ДЛЯ КОМПОНЕНТІВ ОПИСУ
(тема)

Виконав:
студент 4 курсу, групи ІТІНФ-19-2
Марков Т. Є.
(прізвище, ініціали)

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

Тип програми освітньо-професійна

Освітня програма Інформатика
(повна назва освітньої програми)

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

Допускається до захисту

Зав. кафедри _____ Кобилін О.А.
(підпис) (прізвище, ініціали)

2023 р.

Харківський національний університет радіоелектроніки

Факультет Інформаційно-аналітичних технологій та менеджменту
(повна назва)

Кафедра Інформатики
(повна назва)

Рівень вищої освіти перший (бакалаврський)

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

Тип програми освітньо-професійна

Освітня програма Інформатика
(повна назва освітньої програми)

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)
«____»_____ 2023 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові Маркову Тарасу Євгеновичу
(прізвище, ім'я, по батькові)

1. Тема роботи Розроблення методу класифікації зображень із застосуванням матриці відстаней для компонентів опису

затверджена наказом університету від 15 травня 2023 року № 474 Ст

2. Термін подання студентом роботи до екзаменаційної комісії 27 травня 2023 р.

3. Вихідні дані до роботи науково-методична та науково-технічна література з аналізу та розпізнавання зображень, матеріали конференцій, дані інтернет-мережі, бібліотека комп'ютерного зору з відкритим кодом OpenCV, мова програмування Java, середовище розробки IntellijIdea.

4. Перелік питань, що потрібно опрацювати в роботі
1. Огляд різних метричних просторів.
2. Математичний опис кваліфікаційних ознак.
3. Опис моделі матриці відстаней.
4. Застосування та порівняння класифікаційних ознак на тестових зображеннях.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри) Актуальність задачі, моделі ключових точок і дескрипторів, постановка задачі, тестові зображення, схема зіставлення розподілів і інтеграційних ознак, результати програмного моделювання розробленої модифікації.

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
Консультант з дотримання діючих стандартів та норм	Доцент Творошенко І.С.		

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Отримання завдання на кваліфікаційну роботу	10.04.2023	
2	Аналіз завдання, підбір літератури	11.04.23-15.04.23	
3	Аналіз літератури з досліджуваної проблеми	16.04.23-19.04.23	
4	Аналіз технічних засобів	20.04.23-30.04.23	
5	Проведення дослідження	01.05.23-12.05.23	
6	Програмна реалізація	13.05.23-18.05.23	
7	Оформлення пояснювальної записки	19.05.23-22.05.23	
8	Перевірка на плагіат	30.05.23	
9	Рецензування	31.05.23	
10	Підготовка презентації та доповіді	25.05.23-31.05.23	
11	Занесення роботи в електронний архів	05.06.23	
12	Попередній захист кваліфікаційної роботи	05.06.23	

Дата видачі завдання 10 квітня 2023 р.

Студент _____
(підпис)

Керівник роботи _____ проф. Гороховатський В.О.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ/ABSTRACT

Пояснювальна записка до кваліфікаційної роботи: 63 с., 29 рис., 51 джерело.

КОМП'ЮТЕРНИЙ ЗІР, КЛАСИФІКАЦІЯ ЗОБРАЖЕНЬ, ДЕСКРИПТОР КЛЮЧОВИХ ТОЧОК, МАТРИЦЯ ВІДСТАНЕЙ, КЛАСИФІКАЦІЙНА ОЗНАКА.

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

Метою роботи є формування, застосування і дослідження систем класифікаційних ознак, створених на основі матриці відстаней для опису зображення у вигляді множини дескрипторів ключових точок.

Здійснено побудову і програмне моделювання класифікаційних ознак зображень, заснованих на значенні матриці відстаней між дескрипторами із використанням метрики Геммінга.

На підставі впровадження апарату матриці відстаней вдалося сформувати інтегровані ознаки у вигляді одновимірних розподілів даних і скоротити обчислювальні витрати при забезпеченні результативності класифікації на навчальній вибірці даних.

COMPUTER VISION, IMAGE CLASSIFICATION, KEYPOINT DESCRIPTOR, DISTANCE MATRIX, CLASSIFICATION MARK.

The object of the work is the method of classifying images by description in the form of multiple descriptors of key points.

The method of work is shaping, designing and maintaining systems of classifiers, created on the basis of a matrix of views to describe the image at the sight of a plurality of descriptors of key points.

It was designed to develop and programmatically model the classification features of images, based on the value matrix between the descriptors from the variation of the Gemming metric.

On the basis of the implementation of the matrix apparatus, it was possible to form integrated signs in the visualization of similarly similar subdivisions of data and the shortness of the calculation of counting at the safe effectiveness of the classification on the initial selection of data.

ЗМІСТ

Перелік умовних позначень, символів, одиниць, скорочень і термінів.……6

Вступ...……..7

1 Методи розпізнавання зображень за множиною ключових точок ……..9

1.1 Визначення та методи формування ключових точок……...9

1.2 Способи і приклади використання ключових точок……..15

1.3 Постановка задачі……..18

2 Математичні моделі аналізу матриці відстаней……...19

2.1 Застосування апарату метрик……...19

2.2 Аналіз систем класифікаційних ознак……..……...29

3 Результати моделювання і експериментального дослідження……..……35

3.1 Особливості програмної реалізації……..35

3.2 Результати експериментів…….42

Висновки…….……..57

Перелік джерел посилання…..……..……..58

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

PSI – Population Stability Index (індекс стабільності популяції)

AKAZE – Accelerated-Kaze (прискорений-Kaze)

SIFT – Scale-Invariant Feature Transform (масштабно-інваріантне перетворення ознак)

ORB – Oriented FAST And Rotated BRIEF (орієнтований FAST та обертаємий BRIEF)

LDB – Local Difference Binary (бінарна локальна різниця)

M-LDB – Modified-LDB (модифікований- LDB)

КТ – ключова точка

ВСТУП

Людина отримує більшу частину інформації про навколишнє середовище через зір, його можна назвати головним почуттям людини, тому можливість навчити комп'ютер бачити, є досить привабливим напрямком у розвитку комп'ютерних технологій [1].

Упродовж всього розвитку люди у різних сферах аналізували великі обсяги візуальних даних, але в наш час більшу частину роботи намагаються автоматизувати, тому можливість розрізняти різні образи є вкрай необхідною [2]. Комп'ютерний зір використовують у різних науках і галузях техніки, у медицині, виробництві та інших сферах. Особливою сферою застосування таких технологій є стрімкий розвиток безпілотних автомобілів та літальних апаратів, де від точності розпізнавання різних об'єктів залежить не тільки правильність побудови траєкторії або орієнтація на основі дорожніх знаків, але й життя пасажирів та людей навколо [1-8].

Розпізнавання складних візуальних фізичних об'єктів, завжди було складною задачею. Подібні задачі є надзвичайно важливими для розвитку технологій, тому що такі об'єкти є навкруги нас, і правильна взаємодія автоматизованих систем з ними – запорука їх розвитку та безпеки [2, 9].

Зараз навіть проводяться спеціальні конкурси для визначення найкращого алгоритму, що може правильно класифікувати візуальні об'єкти. Наприклад на конкурсі ImageNet 2014 року вже була представлена модель, яка може розпізнавати зображення майже на одному рівні з людиною. Такий результат є надзвичайним, враховуючи дата сет, що включає в себе 1000 різних категорій, серед яких навіть є різні породи собак [3]. Всі найкращі методи, в особливості ті, що були розроблені в рамках цього конкурсу, представлені на даний момент і основані на нейронних мережах.

Окремим інструментом, який хоч самостійно не є ефективним в такому розпізнаванні, але може доповнити нейронні мережі для зменшення затрат на навчання та класифікацію, є методи розпізнавання на підставі визначення

множини ключових точок зображення. Їх використовують для ідентифікації головних особливостей об’єкту та опису їх у стислому вигляді задля розпізнавання [4].

Алгоритми формування ключових точок розвиваються та удосконалюються з кожним роком, поступово оптимізуються та адаптуються для використання у різних умовах та прикладних системах.

Апарат дескрипторів ключових точок результативно застосовується у сфері комп’ютерного зору, за його допомогою можна набагато простіше порівнювати об’єкти для встановлення ступеню їх подібності. Це в сукупності з їх меншим потребами щодо обчислювальних ресурсів, відкриває широкий спектр для сфери їх використання у виробництві або обробці зображень.

Таким чином, на даний момент використання та дослідження апарату дескрипторів ключових точок та ознак, основаних на них, є перспективним напрямком розвитку, так як використовується в багатьох сферах [2]. Бакалаврська робота має на меті створити та дослідити у засобах класифікації зображень певний клас таких ознак, а саме ознаки, сформовані на основі застосування моделі матриці відстаней.

1 МЕТОДИ РОЗПІЗНАВАННЯ ЗОБРАЖЕНЬ ЗА МНОЖИНОЮ КЛЮЧОВИХ ТОЧОК

1.1 Визначення та методи формування ключових точок

Комп’ютерний зір є перспективним напрямком у сфері комп’ютерних наук. Розпізнавання зображень використовується у багатьох сферах виробництва, науки або аналізі різноманітної інформацій. Розпізнавання зображень – задача досить легка для людини, але комп’ютери потребують застосування певних складних алгоритмів для цього. Деякі алгоритми можуть розпізнавати зображення за їх піксельним описом, як це роблять деякі нейронні мережі, наприклад для розпізнання окремих рукописних символів [2, 5]. Але для багатьох сучасних прикладних алгоритмів основою розпізнавання є ключові точки.

Ключові точки – це частина зображення, яка особливо помітна та/або підкреслює унікальну особливість [6].

Для формування ключових точок були розроблені спеціальні алгоритми, що називаються детекторами ключових точок. Детектори ключових точок повинні виділяти ключові точки, сукупність яких є унікальними для зображення, та бути інформативними [7].

Ключові точки використовуються для визначення ключових областей об'єкта, які є основою для подальшого зіставлення та ідентифікації на новому зображенні [6].

Виявлення ключових точок досить популярно у сфері комп'ютерного зору для визначення місцезнаходження особливих частин об'єкта на зображенні. Цей метод використовують для визначення просторового розташування об’єкту або точок, які виділяються на зображенні [4].

Іноді пошуком ключових точок називається визначення певних об’єктів на зображенні, за якими можна дослідити положення основного об’єкту.

Найпоширеніший випадок – це системи розпізнавання рухів (рис. 1.1) або облич. Тут у якості ключових точок виступають частини нашого тіла, наприклад лікті або губи [8].

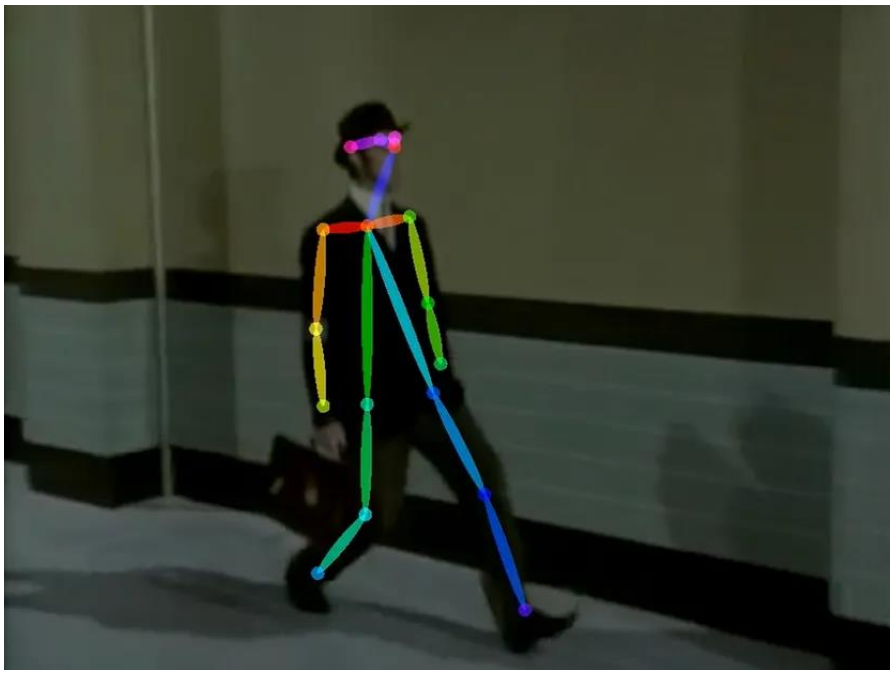

Рисунок 1.1 – Розпізнавання рухів

Інші ключові точки – це особливості самого зображення, не завжди співпадають з особливостями самого об’єкту. Зазвичай такою точкою виступає піксель, який різко відрізняється від інших пікселів в певній області. Зазвичай детектори працюють так, щоб точки знаходились на кутах об’єктів або на якійсь невеликій деталі. Це робиться через те, що майже завжди саме такі зони є максимально стійкими до різноманітних спотворень зображень, освітлення, шумів або до змін положення об’єкту в просторі [4].

Отримані ключові точки описуються дескрипторами. Дескриптори ключових точок являють собою набір бітів, що описує ключову точку.

Для зображення ключових точок (рис. 1.2) використовуються не дескриптори, а інші параметри, це координати точки на зображені і іноді – радіус зони, яку аналізували під час формування точки.

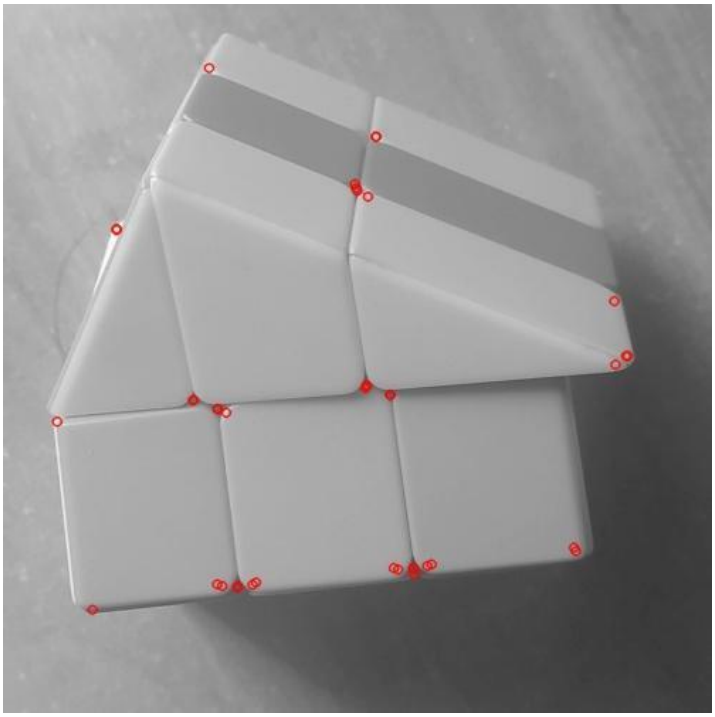

Рисунок 1.2 – Приклад візуалізаціїї ключових точок

Їх можна умовно поділити на дві групи. До першого напрямку віднести ті, що шукають точки за допомогою нейронних мереж. До таких відносяться: DetNet, TILDE, LIFT. До другого можна віднести детектори з вручну запрограмованим алгоритмом.

До таких детекторів відносяться SIFT, SURF, BRIKS, ORB, KAZE, AKAZE, FAST та інші. Детектори представляють описи точок у вигляді дескрипторів. Зазвичай вони мають свій дескриптор, який за назвою відповідає детектору, але є винятки: FAST не має свого дескриптора, тому використовується з іншими, і навпаки, BRIEF и FREAK є тільки дескрипторами [6, 9, 10]. Не обов'язково використовувати детектор тільки в парі з своїм дескриптором, наприклад, детектор AKAZE може працювати, як і з дескриптором KAZE так і з M-LDB [9].

Ці методи можна умовно поділити на декілька типів, найпомітніші з них – робота на основі початкового зображення, та на основі зображення після застосування деякого фільтру.

Використання початкового зображення базується на порівнянні

інтенсивності пікселя з інтенсивністю пікселів в зоні навколо цього пікселя. До таких детекторів відносяться наприклад Harris і FAST, що є покращеною версією попереднього [6].

Для прикладу (рис. 1.3), детектор FAST розглядає пікселі з номерами 1, 5, 9, 13, у випадку, коли кожен з цих пікселів є одночасно темнішим або яскравішим за кандидата I_p, а різниця значень перевищує певний поріг, то точка помічається як ключова. В оригінальному алгоритмі радіус кола 3 але допускається його зміна задля покрашення результату [6].

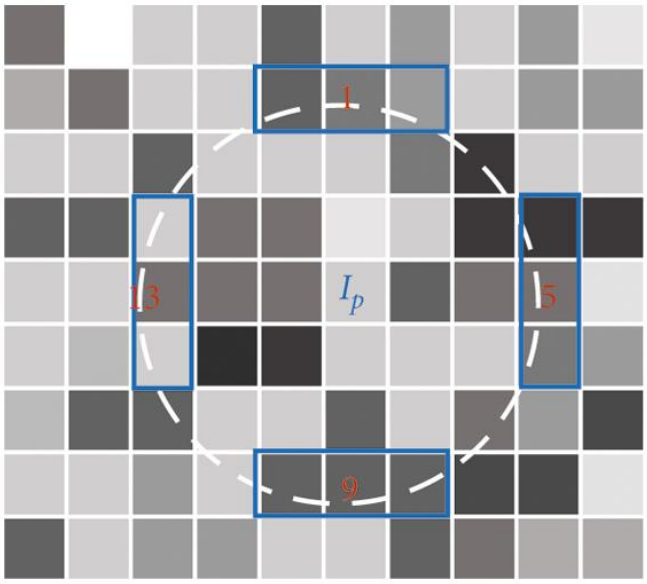

Рисунок 1.3 – Приклад роботи алгоритму FAST

Через такий підхід детектор FAST та інші детектори, що працюють на схожих алгоритмах, чутливі до шуму, розмиття та зміни масштабу. Також певні неприємності дає освітлення, так як при низькій інтенсивності різниця для будь-яких двох пікселів може бути нижче порогової [6].

Не дивлячись на такі недоліки, вони мають певні переваги над наступним типом дескрипторів. Головна перевага полягає в їх швидкодії та використанні невеликої кількості ресурсів, що дозволяє легше їх інтегрувати в різноманітні системи.

Боротьба з цими недоліками виконувалася у інших дескрипторах,

наступний з яких є покращеною версією попереднього: SIFT та SURF, KAZE та AKAZE [9, 6]. Для запровадження інваріантності до перетворень оригінальний детектор SIFT використовує фільтр Гауса для розмиття зображення, створюючи так звані октави. Вже на різниці різних розмитих зображень з октави обраховуються ключові точки, такі як локальний максимуми яскравості, що задовольняє певним умовам. Так як кожна ключова точка обраховується на основі початкового зображення з різним ступенем розмиття, це робить їх майже нечутливими до перелічених умов.

Іншим перетворенням, що заважає зіставленню точок зображень, є перетворення повороту. Для боротьби з цим обчислюється основний напрямок градієнту ключової точки та фактично створюються багато однакових точок з різними напрямками градієнту та масштабу [11].

Таким чином, отримуються ключові точки, що описані положенням, масштабом та напрямком градієнту. Хоча дескриптори називаються дескрипторам ключових точок, насправді дескриптори описують не саму точку, а певну зону навколо неї, у SIFT це зона 16 на 16. Ця зона в свою чергу розбивається на 16 блоків, в кожному блоці обраховується градієнт у 8 сторін, в результаті отримується вектор ознак з 128 бінів. Можна сказати, що дескриптор є описом градієнтів для 16 точок, що розміщені навколо ключової [11].

Так як для окремих частин зони описуються напрямки градієнту, а при повороті зображення вони змінюються, утворюється залежність дескриптора від повороту. Для досягнення інваріантності і для них віднімають напрямок ключової точки. Так як різниця між напрямком градієнта точки та напрямком градієнту зон незалежна від повороту зображення, то отримується незалежна від нього ознака.

Інша складність пов’язана не з самим зображенням, а з умовами його створення. Різне освітлення потребує різного порогового значення для інтенсивності, через що дескриптори можуть змінюватися. Для боротьби з цим в оригінальному алгоритмі розглядають всі 128 бінів. Ті, що більше 0,2,

заміняються на 0,2, і після цього всі вони знов нормалізуються. Це дозволяє зменшити залежність від освітлення [11].

Всі похідні від SIFT детектори працюють на схожих принципах, що зарекомендувало себе як гарний спосіб зменшення впливу перетворень. Тому використання таких детекторів є більш ефективним.

Завжди потрібно враховувати, що не дивлячись на складні алгоритми, отримані таким чином ключові точки не є абсолютно інваріантними до цих умов, завжди є ймовірність невеликих змін [12]. Як правило, такі зміни майже не впливають на близькість таких точок, тому алгоритми їх порівняння досить прості.

Сучасні методи аналізу візуальної інформації базуються на оцінках подібності між еталоном та об’єктом, що описується набором ключових точок [13, 14].

В алгоритмі SIFT передбачалося, що ключові точки між двома зображеннями будуть порівнюватися шляхом визначення їх найближчих сусідів. Такий підхід є найпростішим, але він створює багато зайвих зв’язків між точками, які насправді не є відповідними один одному. Досить часто буває так, що в деяких випадках відстань до найближчої та другої найближчої майже співпадають. Досить поширеною причиною такої поведінки є шум або наявність повторюваних елементів на зображенні. У цьому випадку розглядають відношення найближчої відстані до другої найближчої відстані. Невеликі коливання відстаней між відповідними ключовими точками можуть бути наслідком зміни освітлення або повороту об’єкту. В такому випадку встановлюють, чи могло виникнути таке змішення випадково, так як поєднати неправильні ключові точки гірше, ніж не поєднувати зовсім. У SIFT знаходиться відношення між двома найменшими відстанями, якщо таке відношення більше 0,8, вони відбраковуються. Згідно з документацією, такий підхід повинен усувати близько 90% помилкових збігів та помічає лише 5% правильних збігів [11, 15-17].

Такий спосіб працює для всіх типів дескрипторів, підхід може

змінюватися тільки за типом відстані та значенням порогу еквівалентності для збігу відстаней, при якому ключова точка відкидається. Приклад цього процесу можна побачити на рисунку 1.4. На ньому наведено ключові точки, сформовані детектором AKAZE, де відкидалися точки, якщо відношення відстаней до двох найближчих була більша за 0,8.

Також потрібно зауважити що створювати зв’язки між ключовими точками з дуже великою різницею теж недоцільно, так як велика ймовірність, що вони ніяк з один одним не пов’язані, тому в прикладі відсікалися зв’язки, в яких відстань перевищувала 100.

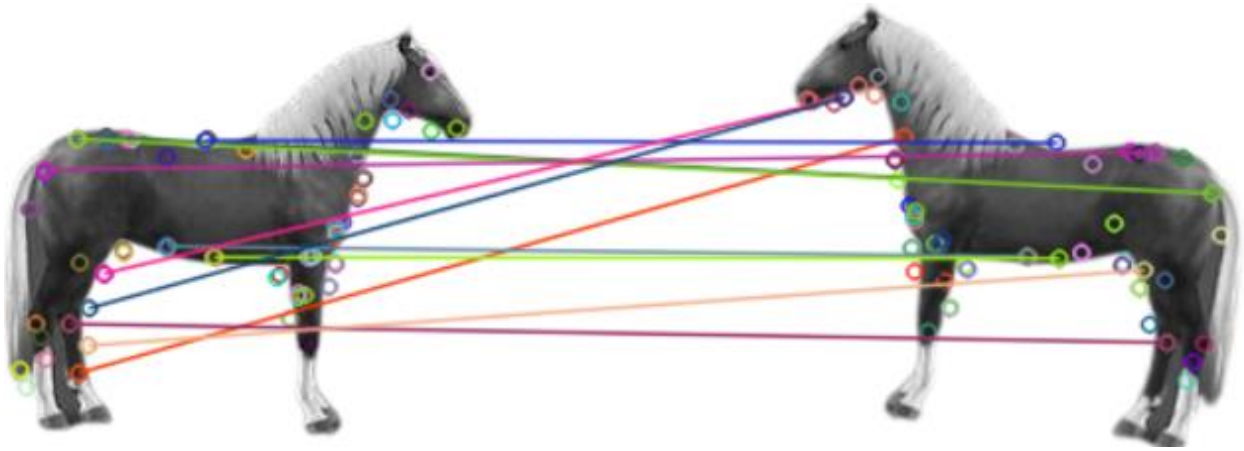

Рисунок 1.4 – Відповідні точки двох зображень

2.2 Способи і приклади використання ключових точок

Так як ключові точки зображення виділяють його особливості, найпростішим напрямком їх використання є розпізнавання предметів на зображенні. На їх основі може проводитись як пошук конкретно предмету на зображенні, так і предмету, що відноситься до певної групи чи класу. Наприклад, ключові точки для обличчя людини також можуть знаходитись на основі дескрипторів ключових точок, хоча відповідні частини лиця для різних людей можуть мати дуже не схожі набори точок.

Розпізнавання конкретного предмету на зображенні проходить з

використанням уже розглянутих алгоритмів. Знаходимо область, в якій багато дескрипторів відповідають дескрипторам із зображення відповідного предмета, і якщо кількість співпадінь перевищує певний поріг, ця область позначається відповідною. Також при певних налаштуваннях фільтрації дескрипторів можна навіть розпізнавати декілька однакових предметів на зображенні [4].

На рисунку 1.5 великими рамками виділені зони, де на думку моделі знаходяться предмети, а маленькими – зони з ключовими точками, за яким вони були розпізнані.

Рисунок 1.5 – Приклад пошуку предметів на зображенні

Іншою сферою використання ключових точок є порівняння зображень та визначення того, чи є вони перетвореннями один одного. Такий приклад було наведено на рисунку 1.4. За такими ключовими точками після зіставлення можна навіть встановити, які перетворення були проведені з зображенням.

Іншою задачею, яку вирішують за допомогою ключових точок, є зшивання декількох зображень в одне. При цьому зображення можуть бути різного розміру, з різним освітленням або навіть з трішки зміненим ракурсом. В такому випадку на двох зображеннях знаходяться відповідні ключові точки, та одне з зображень змінюється таким чином, щоб ключові точки максимально близько співпадали з ключовими точками іншого. Потім

проходить з’єднання зображень таким чином, щоб координати цих точок співпадали, в результаті буде отримано нове зображення. До цього зображення вже можна приєднати ще одне і так далі. Такий підхід дозволяє зшити зображення без втрати інформації, в результаті отримують зображення, що включає в себе інформацію з усіх зображень (рис. 1.6) [4].

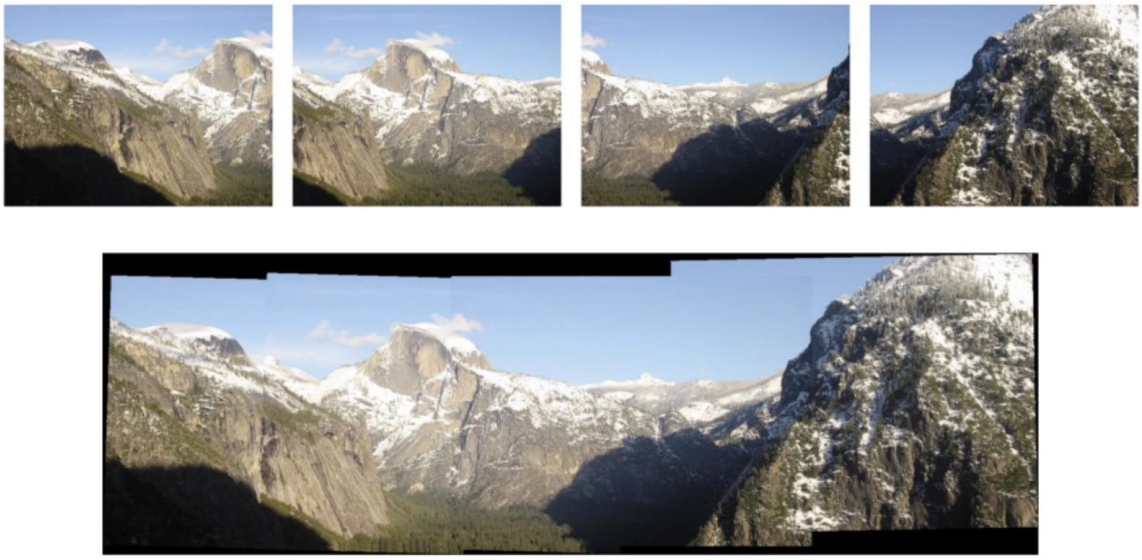

Рисунок 1.6 – Приклад зшивання зображень

Згадане вище розпізнавання предметів з певної групи стикається з проблемою, що нема універсального набору дескрипторів для різних об’єктів. Така проблема часто вирішується за допомогою використання нейронних мереж. В даному випадку принцип навчання нейронної мережі не відрізняється від усіх інших застосувань. Єдиною потребою для нього є розмічений набір даних, що включає в себе великий діапазон різноманітних ситуацій [18]. На даний момент нейронні мережі мають значний недолік який пов’язаний з тим, що неможливо точно встановити точну причину отримання певного результату. Це впливає на такі чутливі напрямки, як ідентифікація людини, медицина та інші [17-20].

Таким чином, ключові точки і їх дескриптори є невід’ємною частиною комп’ютерного зору, вони дають можливість швидко та легко знаходити відповідності між різними зображеннями. Це дозволяє використання апарату

ключових точок у різноманітних сферах прикладного застосування.

1.3 Постановка задачі

Дослідження ознак, основаних на множині ключових точок, є актуальною для обробки і розпізнавання зображень. Тому ставиться завдання дослідження таких ознак, де використовується апарат матриці відстаней між множинами дескрипторів.

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

Метою роботи є формування, застосування і дослідження систем класифікаційних ознак, створених на основі матриці відстаней для опису зображення у вигляді множини дескрипторів ключових точок.

Для реалізації цієї мети необхідно виконати наступні завдання:

− вивчити різні типи класифікаційних ознак на основі множини дескрипторів;

− створити модель матриці відстаней всередині опису та між описами;

− побудувати класифікаційні ознаки на основі розподілів значень матриці відстаней;

− виконати програмне моделювання класифікатора зображень з використанням побудованих ознак;

− оцінити результативність класифікації на реальних зображеннях.

Як матеріал для дослідження обрано кольорові зображення трьох коней з кількістю дескрипторів 40.

2 МАТЕМАТИЧНІ МОДЕЛІ АНАЛІЗУ МАТРИЦІ ВІДСТАНЕЙ

2.1 Застосування апарату метрик

Матриця відстаней містить найбільш повну інформацію для аналізу, включаючи в себе інформацію про метричні зв'язки між компонентами описів. На підставі повноцінної інформації класифікація відбувається з максимальною результативністю. Існують більш швидкісні методи, засновані на подані еталонів у вигляді їх «центрів» [20-22], при тому їх результативність залежить не тільки від способу вибору центрів та й від самих даних, тому близькі описи не є достатньою умовою для результативності.

Розповсюджені застосування матриці відстаней у сфері візуалізації багатовимірних об'єктів, основуючись на картах Кохонена [23], що використовується в системах пошуку інформації, при оцінюванні результативності класифікації [24]. Матриця відстаней також впроваджується на множині класів, на основі чого може бути реалізований критерій класифікації – матриця неточностей класифікатора [25, 26].

Матриця відстаней між описами класів застосовується для експериментального оцінювання якості розроблених класифікаторів, але формалізм матриці відстаней між компонентами описів класів досліджено ще недостатньо глибоко.

Матриця відстаней зазвичай використовується як проміжна модель структури даних при обчисленні ступеня релевантності структурних описів. Детально вивчені тільки розподіл компонентів, способи побудови класифікаторів на їх основі та вплив складових опису на загальну результативність класифікатора.

Нехай $A = \{a_i\}_{i=1}^{s}$, $B = \{b_j\}_{j=1}^{s}$ – дві скінченні множини n-мірних векторів з потужністю s, $A \subset R^n$, $B \subset R^n$, R^n – простір числових векторів

розмірності n. Використання однакої потужності для аналізованих множин дещо спрощує їх аналіз і може бути досягнутий спеціальним корегуванням потужності множин. Конкретні множини *A,B* дескрипторів КТ для зображень на практиці відносять до класу мультимножин, тому що вони досить часто включають досить близькі, а подекуди і повністю еквівалентні елементи [27], чого не було отримано під час практичного розгляду отриманих КТ.

Розглянемо формалізм матриці відстаней M розміром $s \times s$ для елементів двох множин *A,B*

$$M[A,B] = \{\{m_{i,j}\}_{i=1}^{s}\}_{j=1}^{s},\ m_{i,j} = \rho(\mathrm{a}_i, \mathrm{b}_j), \tag{2.1}$$

де $\rho(a_i, b_j)$ – деяка фіксована відстань у векторному просторі R^n, $a_i \in A$, $b_j \in B$.

Метрика ρ – це функція, яка визначає відстань у метричному просторі.

Метрики повинні задовольняти таким умовам:

– $\forall x, y \in X : \rho(x, y) \geq 0$, причому $\rho(x, y) = 0 \Leftrightarrow x = y$ (невід'ємність);

– $\forall x, y \in X : \rho(x, y) = \rho(y, x)$ (симетричність);

– $\forall x, y, z \in X : \rho(x, y) \leq \rho(x, z) + \rho(z, y)$ (нерівність трикутника).

Ці властивості відстаней покладені в основу означення метричного простору. Метричним простором називається множина, між елементами якої визначена метрика (відстань). Наявність метрики дозволить визначити означення границі послідовності елементів цієї множини. А вона, в свою чергу, лежить в основі відомих понять: границі функції в точці, неперервності, похідної, інтеграла Рімана, суми ряду. Основні поняття теорії метричних просторів сформував на початку двадцятого століття відомий французький математик Моріс Фреше [28].

Метрики також підпорядковуються певним правилам, що дозволяють порівнювати їх між собою.

Метричні простори (X_1, ρ_1) і (X_2, ρ_2), рівні тоді і тільки тоді, коли

$X1 = X_2$ і $\rho_1 = \rho_2$. Потрібно враховувати, що навіть коли (X, ρ) – метричний простір, а $Y \subset X$. Тоді (Y, ρ) теж метричний простір, але він буде рівним тільки в тому випадку коли $Y = X$.

Базуючись на аксіомах, виводяться найпростіші властивості метрик.

Нерівність многокутника – для довільних $x_1, x_2, \dots, x_n \in X$, з метричного простору (X, ρ) виконується нерівність (2.2)

$$\rho_{(x_1,x_n)} \leq \rho_{(x_1,x_2)} + \rho_{(x_2,x_3)} + \cdots + \rho_{(x_{n-1},x_n)} . \tag{2.2}$$

Це випливає із послідовного застосування нерівності трикутника (2.3)

$$\begin{aligned} \rho_{(x_1,x_n)} \leq \rho_{(x_1,x_2)} + \rho_{(x_2,x_n)} \leq\ & \rho_{(x_1,x_2)} + \rho_{(x_2,x_3)} + \rho_{(x_3,x_n)} \leq \cdots \\ & \leq \rho_{(x_1,x_2)} + \rho_{(x_2,x_3)} + \cdots + \rho_{(x_{n-1},x_n)} . \end{aligned} \tag{2.3}$$

Нерівність чотирикутника – для довільних $x, y, v, u \in X$, з метричного простору (X, ρ), виконується нерівність (2.4)

$$|\rho_{(x,u)} - \rho_{(y,v)}| \leq \rho_{(x,y)} + \rho_{(u,v)}. \tag{2.4}$$

Ця нерівність випливає з нерівності многокутників при $n=4$ та аксіоми симетричності

$$\begin{aligned} \rho_{(x,u)} &\leq \rho_{(x,y)} + \rho_{(y,v)} + \rho_{(v,u)} , \\ \rho_{(y,v)} &\leq \rho_{(y,x)} + \rho_{(x,u)} + \rho_{(u,v)} . \end{aligned} \tag{2.5}$$

Друга нерівність трикутника – для довільних $x, y, u \in X$, з метричного простору (X, ρ) виконується нерівність (2.6)

$$|\rho_{(x,u)} - \rho_{(y,v)}| \leq \rho_{(x,y)} . \tag{2.6}$$

Це отримується з нерівності чотирикутника (2.4) при $v = u$.

Для використання метричного простору потрібна множина та функція відстані. Множиною в такому випадку виступають множина дескрипторів, всі можливі множини дескрипторів зображення відповідно будуть її підмножинами. А відстань буде вибрана з урахуванням особливостей елементів розглянутої множини.

Евклідова відстань – це відстань між вдома точками евклідового простору, що базується на використанні теореми Піфагора, є найбільш поширеною для просторів різної вимірності. Вона фактично представляє собою найкоротший шлях, який потрібно пройти, щоб потрапити з одного елемента в інший, або найкоротший шлях від точки однієї множини до точки іншої множини

$$\rho_E(a,b) = \sqrt{\sum_{i=1}^{n}(a_i - b_i)^2}, \tag{2.7}$$

де a, b – вектори;

n – їх довжина вектора.

Мангетенська відстань – Mg, відстань, також відома як «відстань міських кварталів», за цією метрикою відстань між точками у багатовимірному просторі дорівнює сумі модулів різниць відповідних координат цих точок

$$\rho_{Mg}(a,b) = \sum_{i=1}^{n}|a_i - b_i|. \tag{2.8}$$

Мангетенська відстань зазвичай більша ніж Евклідова, так як відстань в більшості випадків відрізняється від прямої.

Відстань Мінковського – метрика, яка використовується в нормованому векторному просторі (n-вимірному реальному просторі), що означає, що її

можна використовувати в просторі, де відстані можуть бути представлені у вигляді вектора, що має довжину.

$$\rho_{Mi}(\mathrm{a},\mathrm{b}) = (\sum_{i=1}^{n}|a_i - b_i|^p)^{1/p}, \tag{2.9}$$

де p – порядок.

Відстань Геммінга – це число позицій, у яких у двох двійкових слів різне значення, при умові що ці слова однакові за довжиною. Хоча вона також може використовуватись і для строк цифр, чисел та навіть букв, що не доступно попередніми відстаням без перетворення їх в чисельні еквіваленти

$$\rho_G(a,b) = \sum_{i=1}^{n} g(a_i, b_i), \tag{2.10}$$

де функція g порівнює відповідні елементи, і у випадку їх співпадіння повертає одиницю, у противному випадку повертається нуль.

Для бінарних даних відстань Геммінга співпадає з Мангетенською.

Розглянемо метрику косинуса – це косинус кута між векторами

$$d_{cos}(a,b) = cos(\theta) = \frac{a \cdot b}{\|a\|\|b\|} \quad . \tag{2.11}$$

При цій метриці два вектори з однаковою орієнтацією мають подібність 1, а вектори з діаметрально протилежною орієнтацією мають –1. Необхідно враховувати, що ця відстань не залежить від розміру самих векторів, тобто не завжди така метрика відповідає дійсної відстані між об’єктами.

Відстань Чебишева, або відстань шахової дошки – дорівнює найбільшій різниці між двома векторами у будь-якому координатному вимірі (2.12). Іншими словами, це просто максимальна відстань вздовж однієї осі

$$d_{CH}(a,b) = \max_i(|a_i - b_i|). \tag{2.12}$$

Відстань Канберра схожа на Мангетенську, вона теж розглядає пари координат, але бере не абсолютну, а відносну відстань

$$d_K = \sum_{i=1}^{n} \frac{|a_i - b_i|}{|a_i| + |b_i|} \quad . \tag{2.13}$$

Іншою, але вже нормалізованою відстанню є відстань Соренсена. Відстань Соренсена або відстань Брея Кертіса – схожа на відстань Канбера, але замість початкового ділення та суми спочатку знаходиться сума, а вже потім – ділення

$$d_s = \frac{\sum_{1=i}^{n} |a_i - b_i|}{\sum_{i=1} (a_i + b_i)}. \tag{2.14}$$

Відстань Махаланобіса ефективно працює з багатовимірними даними, оскільки використовує коваріаційну матрицю змінних для знаходження відстані між точками даних і центром (2.15). Це означає, що ця відстань виявляє виключення на основі схеми розподілу точок даних набагато краще, ніж Евклідова відстань. Ця відстань частіше за все використовується для того, щоб знайти границі множини

$$d_M^2(a, b) = (a - b)^T C^{-1} (a - b), \tag{2.15}$$

де C – коваріаційна матриця.

Іншою задачею є знаходження відстаней на сфері. Основною відмінністю є те, що найкоротшою відстанню не може витупати пряма, і те, що координати задаються не векторами, а кутами широти та довготи. Для таких задач зазвичай використовують відстані Гаверуса та Вінсенті.

Відстань Гаверсуса – це відстань між двома точками, що знаходяться на сфері. Таку відстань можна порівняти з Евклідовою, так як ці відстані

найменші, враховуючи те що прямі лінії на сфері неможливі

$$d_G = 2 \cdot r \cdot arcsin\left(\sqrt{sin^2\left(\frac{\varphi_2-\varphi_1}{2}\right) + cos(\varphi_1) \cdot cos(\varphi_2)\, sin^2\left(\frac{\lambda_2-\lambda_1}{2}\right)}\right), \quad (2.16)$$

де, φ_i – широта;

λ_i – довгота відповідних точок.

Головним недоліком цієї відстані, є те, що за замовчуванням вважається, що точки повинні лежати на поверхні сфери, чого на практиці майже не відбувається, тому при обрахуванні відстаней на поверхні землі краще використовувати відстань Вінсенті.

Відстань Вінсенті – це відстань між двома точками, що находяться на поверхні еліпсоїда. Хоч ця відстань і подібна до відстані Геверуса, але потребує більше параметрів: довжину великої півосі, довжину малої півосі, згладжування, координати широти, довготи. Цей метод є ітераційним, поступово мінімізуючи різницю між попереднім та новим результатом. І відповідно, вона обраховується набагато важче, ніж відстань Геверуса.

Для бінарних векторів у якості відстані ρ була застосована найефективніша з розглянутих у обчислювальному аспекті відстань Хемінга [29, 30].

Формалізм (2.1) відповідає встановленню метричного відношення Θ для елементів пари множин: $\Theta\ [A, B] \to M$, де кожній двійці елементів $a_i \in A$, $b_j \in B$ ставиться у відповідність значення відстані ρ між ними. Відстань між об’єктами безпосередньо пов’язана із ймовірністю їх близькості (еквівалентності) в рамках метрично-статистичної теорії класифікації образів [25].

Кожний i-й рядок матриці M відповідає i-й компоненті опису і зберігає всі відстані від i-го елемента множини A до всіх елементів множини B. Проте у задачі класифікації одна із множин зазвичай вважається еталоном [31] (наприклад, A), тому аналіз доцільно здійснювати за рядками

чи стовпцями M .

Особливим випадком для M є випадок, коли множини A та B співпадають. В цьому разі матриця буда описувати спектр значень відстаней в середині множини, це також є значимою класифікаційною ознакою.

Матриця відстаней несе в собі інформацію про такі властивості множин, як перетин, об'єднання, різниця, симетрична різниця [29] з використанням різних граничних значень для метрики подібності двох векторів. Самі граничні значення визначаються, як відстань, на якій або меншій за неї елементи вважаються рівними. Якщо відстань між двома елементами менша за δ (задане значення), вони вважаються рівними

$$a \sim b | \rho(a,b) < \delta. \tag{2.17}$$

Таким чином, перетин множин A та B визначається як множина елементів, для яких в рядку матриці виконується хоча б одна з умов (2.6) [27, 28]. Знаючи потужність множин та їх перетин, легко обраховується коефіцієнт Танімото, що вже використовується в сфері розпізнавання (рис. 2.1) образів [32].

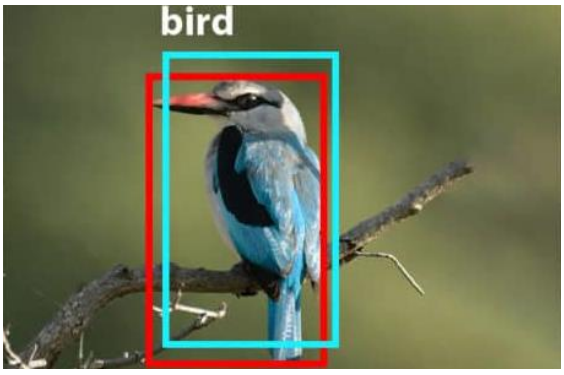

Рисунок 2.1 – Приклад розпізнавання зображення

Використання коефіцієнту Танімото вже зарекомендувала себе як швидкий засіб порівняння [33]. На цьому рисунку зображено приклад розпізнавання, де показані червона та прогнозована (голуба) множини точок,

охоплених прямокутником.

Коефіцієнт Танімото T демонструє відмінність однієї множини від іншої, формально визначається відношенням потужностей пересічення та об'єднання множин і чисельно враховує кількісні показники для подібних і несхожих елементів. Його можна засовувати для визначення рівня подібності двох множин, або для виділення об'єктів з однаковими властивостями. Ці два застосування підходять для розпізнавання зображення

$$T(A,B,C) = \frac{N_C}{N_A + N_B - N_C}, \tag{2.18}$$

де N_A, N_B, N_C – довжини відповідних множин;

C – перетин A та B.

Чим ближче коефіцієнт до одиниці, тим ближче знаходяться множини. Цей коефіцієнт фактично описує відношення кількості елементів, які не потрібно змінювати при переході від однієї множини до іншої.

Метрика Танімото/Жаккара демонструє відстань від однієї множини до іншої, вона визначається відношенням потужностей симетричної різниці та об'єднання множин і чисельно враховує кількісні показники для подібних і несхожих елементів [12]. Він може бути отриманим шляхом віднімання відстані Танімото від 1 або обчисленням

$$TJ(A,B,C) = \frac{N_A + N_B - 2N_C}{N_A + N_B - N_C}. \tag{2.19}$$

Така відстань більш вдало описує відношення близькості між множинами, так як при співпадаючих множинах відстань буде дорівнювати 0, а при зовсім різних – 1.

Відстань Соренсена-Дайса – за своїм означенням схожа на дві попередні, тому що вона теж відображає ступінь подібності двох множин. При цьому вона вважається більш інтуїтивно зрозумілою, оскільки зображає

відсоток пересічення двох різних наборів, теж вварюючись від одного до нуля

$$T(A,B,C) = \frac{N_c}{N_A + N_B}. \quad (2.20)$$

У всіх цих трьох відстаней є один важливий недолік, що полягає в великій залежності відстані від розмірів множин. Так збільшення потужності множин може значно збільшити об'єднання, у випадку відстані Соренса-Дайса суму потужності, не гарантуючи збільшення перетину, та навпаки. В такому випадку це може викликати різкі зміни відстаней при збільшені або зменшені кількості розглянутих дескрипторів, особливо якщо початково розглянути набори були невеликими.

Відстань Хаусдорфа – *H*, це відстань, яка відображає те, наскільки далеко один від одного знаходяться дві підмножини у метричному просторі. Вона знаходиться як значення максимальної з мінімальних відстаней від елементів однієї множини до елементів іншої [34]

$$d_H(A,B) = max\left\{\min_{a\in A}\{\rho(a,B)\}, \min_{b\in B}\{\rho(b,A)\}\right\}, \quad (2.21)$$

де $\rho(a,B) = \min_{b\in B} \rho(a,b)$;

ρ – метрика для пари дескрипторів.

Використання аналізованих відстаней дозволяє оцінити фактичну відмінність між множинами, коли вони не перетинаються.

Відстань Громова-Хаусдорфа дозволяє встановити, наскільки далеко один від одного знаходяться два компактних метричних простори. Якщо (X_1, ρ_1), (X_2, ρ_2) – компактні метричні простори, тоді відстань Громова-Хаусдорфа визначається за формулою (2.20)

$$d_{GH} = inf\{d_H(J_1(X_1), J_2(X_2))\}. \quad (2.22)$$

Тут $J_1: X_1 \to Z$, $та$ $J_2: X_2 \to Z$ – деякі ізометричні вкладення у метричний простір (Z, ρ).

Для підрахування відстаней між кривими теж існують метрики. Найпоширенішими є метрики Ферше та Громова-Ферше.

Метрика Ферше, для двох кривих $f: [a,b] \to X$, i $g: [a`, b`] \to X$ відстань Фреше між ними обраховується за формулою

$$d_F = \inf_{\alpha,\beta} \max_{t\in[0,1]} d\left(f(\alpha(t)), g(\beta(t))\right). \quad (2.23)$$

Тут $d(x, y)$ – евклідова відстань між точками x та y, $\alpha(t)$ та $\beta(t)$ – довільні неперервні неспадні функції з проміжку [0,1] на проміжки $[a,b]$ та $[a`, b`]$ відповідно. Значення функції $\alpha(0) = 0$, $\alpha(1) = 1$, і відповідно для $\beta(t)$.

2.2 Аналіз систем класифікаційних ознак

Створення ефективної системи інформативних ознак для забезпечення результативної класифікації зображень у комп'ютерному зорі є одним з головних завдань теорії розпізнавання даних [17, 35-41].

Найпростішою класифікаційною ознакою виступає множина всіх дескрипторів зображення. Те, наскільки різні множини дескрипторів близькі одна до одної, вказує на те, чи однакові зображення, так як, незважаючи на високу розрізненість та широке практичне застосування дескрипторів, інваріантність до перетворень не абсолютна [11, 12].

Матриця відстаней може містити найбільш повну інформацію для аналізу, але саму матрицю складно використовувати для класифікації, так як при зміні порядку дескрипторів матриця теж змінюється.

Враховуючи те, що при геометричних перетвореннях порядок дескрипторів у описі може змінюватися, ступінь подібності двох множин в

основах матриць можна оцінювати на ознаках, незмінних для порядку дескрипторів. Такою ознакою є розподіл або гістограма значень відстаней. Розглянемо розподіл *h(X)* – це функція від значень матриці відстаней, де по осі *X* відкладається значення відстаней $\rho = 0 \ldots 487$, а по осі *Y* – кількість таких відстаней у матриці.

За гістограмою (рис. 2.2) можна оцінити і закономірності розподілу відстаней, те що більшість відстаней сконцентровано в лівій частині гістограми, та подібність до нормального розподілу. Також вона дозволяє оцінити інші параметри матриці, що можуть використовуватися у якості класифікаційних ознак. Такими ознаками є математичне очікування, мода, дисперсія та інші.

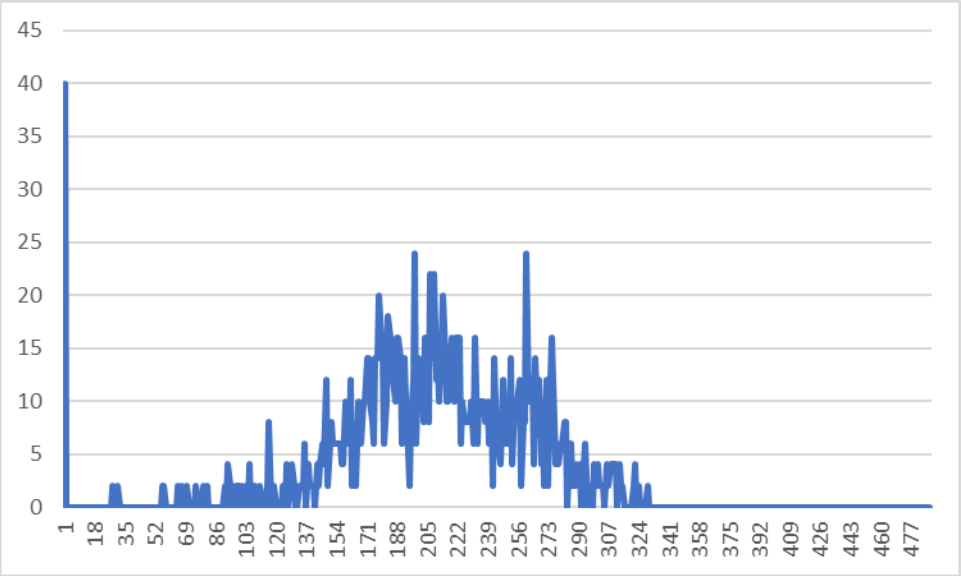

Рисунок 2.2 – Приклад візуалізації гістограми даних

На прикладі видно, що в даному випадку найбільшим значенням є стовпець нулів, що утворилися через порівняння дескрипторів з самим собою. Така тенденція зберігається у всіх еталонах, тому такий стовпець краще не враховувати при розпізнаванні.

Математичне очікування та мода описує центри масиву відстаней, а дисперсія показує те, наскільки в середньому відхиляються значення елементів від центру множини.

Для матриці відстаней розміром $n \times n$ середнє для всіх елементів обчислюється за формулою (2.24), а дисперсія – за формулою (2.25).

$$Mu = \frac{1}{n \cdot n} \sum_{i=1}^{n} \sum_{i=1}^{n} m_{ij}, \tag{2.24}$$

$$\sigma_i^2 = \frac{1}{n \cdot (n-1)} \sum_{i=1}^{n} \sum_{i=1}^{n} (m_{ij} - Mu)^2. \tag{2.25}$$

Для одновимірних масивів даних:

$$Mu = \frac{1}{n} \sum_{i=1}^{n} m_i \,, \tag{2.26}$$

$$\sigma_i^2 = \frac{1}{n-1} \sum_{i=1}^{n} (m_i - Mu)^2 \,. \tag{2.27}$$

Проте потрібно враховувати те, що при використанні дисперсії для опису характеристики розсіювання величин існує проблема, що полягає в тому, що коли величини вимірюються в деяких одиницях, то дисперсія завжди вимірюється в квадраті цієї величини. В більшості випадків замість дисперсії доцільніше мати величину розсіяння, що вимірюється в тих самих одиницях, що і вимірювані значення. Для цього використовується середнє квадратичне відхилення. Середня квадратичне відхилення σ – це корінь квадратний з дисперсії цієї величини.

Крім середнього квадратичного відхилення та дисперсії, мірою розсіяння також виступає середнє лінійне відхилення. Середнє лінійне відхилення – це сума всіх відхилень від центру до елементів масиву, розділене на їх кількість. Але через те, що з властивості середнього випливає, що сума всіх позитивних та негативних відхилень дорівнюють нулю, потрібно використовувати їх модулі для отримання результату.

Для розподілів на основі ймовірності існує спеціальна відстань – метрика розбіжності Кульбака-Лейблера або відносна ентропія. Вона не є метрикою в повному сенсі, так як не виконується властивість симетрії, але її називають асиметричною метрикою. Вона вимірює відносну ентропію або різницю інформації, що подана двома розподілами.

Існує безперервна (2.27) та дискретна (2.28) форми

$$d_k\big(p(x)||q(x)\big) = \int_{-\infty}^{\infty} p(x)\, ln\frac{p(x)}{q(x)}\, dx, \tag{2.28}$$

$$d_k\big(p(x)||q(x)\big) = \sum_{x\in P} p(x)\, ln\frac{p(x)}{q(x)}, \tag{2.29}$$

Відстань Кульбака-Лейблера асиметрична, тобто якщо поміняти *p(x)* та *q(x)*, то отримується різний результат, що може затрудняти порівняння розподілів, тому замість неї можуть використовувати індекс PSI [42].

PSI – це метрика, яка вимірює відносну ентропію чи різницю інформації, поданої двома розподілами, і на відміну від метрики Кульбака-Лейблера, вона симетрична. Її також можна розглядати, як інструмент для виміру відстані між двома розподілами даних, що показує, наскільки ці розподіли відрізняються один від одного.

$$PSI(p(x), q(x)) = d_k\big(p(x)||q(x)\big) += d_k\big(q(x)||p(x)\big), \tag{2.30}$$

$$PSI\big(p(x), q(x)\big) = \int_{-\infty}^{\infty} (p(x) - q(x)) ln\frac{p(x)}{q(x)}\, dx\ , \tag{2.31}$$

$$PSI\big(p(x), q(x)\big) = \sum_{x\in P} (p(x) - q(x)) ln\frac{p(x)}{q(x)}\ . \tag{2.32}$$

Відстань Бхаттачар’ї використовується для оцінки відмінностей між

двома розподілами ймовірностей. Вона описує, наскільки перетинаються два розподіли, і може допомогти визначити, наскільки вони схожі чи різні. Ця відстань базується на коефіцієнті Бхаттачар'ї (2.32), який вимірює ступінь перекриття між двома статистичними вибірками чи сукупностями

$$BC(p(x),q(x)) = \sum_{i=1}^{n} \sqrt{\rho(x_i) * q(x_i)}\ . \qquad (2.33)$$

Сама відстань обчислюється як десятинний логарифм (2.34) від коефіцієнту

$$dB(p(x), q(x)) = -\ ln\big(BC\left(p(x), q(x)\right)\ . \qquad (2.34)$$

Основною переваго описаних відстаней над аналогами є те, що вони враховують весь розподіл, а не лише середнє значення та дисперсію. Це робить їх більш стійкими, ніж інші аналоги, до виключень та шуму даних. Це робить їх придатними для наборів даних з високою мінливістю.

Критерій *xi*-квадрат Пірсона – це статистичний критерій, що застосовується до наборів категоріальних даних для оцінки ймовірності того, що будь-яка різниця між наборами є випадковою

$$x^2\big(d(x), q(x)\big) = \sum_{i=1}^{n} \frac{\left(d(x_i) - q(x_i)\right)^2}{q(x_i)}\ . \qquad (2.35)$$

Крім описаних ознак, інваріантною до порядку дескрипторів виступає сума значень стовпців або рядків матриці, що однаково через її симетричність.

За розподілом сум відстаней (рис. 2.3) можна оцінити середню відстань, на якій знаходяться дескриптори від всіх інших, або чи є дескриптори, які значно вирізняються серед усіх інших, у прикладі це 31, так як сума відстаней значно відрізняється від усіх інших.

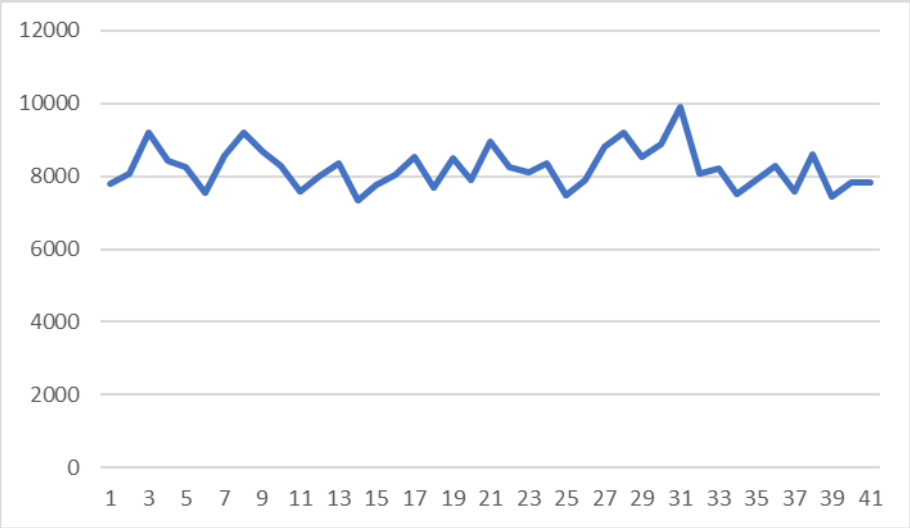

Рисунок 2.3 – Приклад розподілу сум відстаней

За розподілом сум відстаней можна оцінити середню відстань, на якій знаходяться дескриптори від всіх інших, або чи є дескриптори, які значно вирізняються серед усіх інших, у прикладі це 31, так як сума відстаней значно відрізняється від усіх інших.

Важливу класифікаційну інформацію містить параметр медоїду [24, 29]. Медоїд z – це елемент множини з найменшою сумарною відстанню до решти елементів множини, у прикладі – це 14 дескриптор, так як його сума відстаней найменша. Медоїд у задачі класифікації застосовується як деякий центр опису, що може використовуватися у класифікації [43]

$$M_m = arg \min_{y \in M} \sum_{i=1}^{n} d(y, m_i) . \quad (2.36)$$

3 РЕЗУЛЬТАТИ МОДЕЛЮВАННЯ І ЕКСПЕРИМЕНТАЛЬНОГО ДОСЛІДЖЕННЯ

3.1 Особливості програмної реалізації

Для розробки програми була використана мова програмування Java, а саме Java 8 та середовище розробки Intellij Idea. В якості бібліотеки для обробки зображень використовувалася бібліотека комп'ютерного зору OpenCV.

OpenCV – це бібліотека з відкритим кодом для обробки зображень, що включає в себе багато алгоритмів, що забезпечує виконання різноманітних функцій. Однією з основних задач цієї бібліотеки є реалізація алгоритмів комп'ютерного зору [9].

Для обробки ображення у роботі використовувався детектор ключових точок AKAZE, в парі з дескриптором M-LDB. Детектор AKAZE працює на основі октав по аналогії з SIFT, але використання нелінійного коефіцієнту масштабування забезпечує більш швидку обробку зображення. Також досягнута ним висока інваріантність до повороту та масштабування дає перевагу перед іншими більш швидкими детекторами, такими як BRISK або ORB, які працюють на інших алгоритмах [12, 11]. Ці фактори є головними для дескрипторів, тому саме його було обрано для виконання роботи.

Дескриптори LDB та M-LDB, як і багато інших, будуються на основі градієнтів та інтенсивності в певній області навколо ключової точки, використовуючи для опису бінарні тести між середніми значеннями розглянутих областей, а не між окремими пікселями як попередники. Це не сильно уповільнює обчислення у випадку коли детектор вертикальний, тобто не є інваріантних до поворотів. При використанні інваріантних детекторів це вже може значно уповільнити обчислення. До цього додається також порівняння середніх значень вертикальних та горизонтальних похідних в цих областях, що вже дає три біти замість одного. Також іншою перевагою перед

попереднім аналогами є можливість налаштовувати розмір сітки на розглянутій області починаючи від 2 на 2 і більше. Також при використанні з детектором AKAZE вони можуть використовувати значення похідних, що були обчислені ще на етапі формування ключових точок, що вже прискорює обчислення.

Дескриптор M-LDB є бінарним займаючи до 486 бітів. Він, на відміну від свого попередника LDB, майже не залежить від повороту та масштабу. Це досягається за допомогою того, що при формуванні зони, на якій обчислюється дескриптор, враховується орієнтація точки та октава, на якій точка була знайдена. Інваріантність до повороту досягається тим, що зона, на якій проходить обчислення дескриптора, завжди повертається на той самий кут, що і ключова тачка. А масштаб враховується тим, шо її розмір залежить від розміру октави, на якій була знайдена відповідна точка. Таким чином, при повороті зображення або об’єкту кути ключових точок змінюються і настільки ж обертається зона, тому вона буде виділяти ту саму частину об’єкту або зображення, як і до повороту [44].

Головним недоліком через який дескриптор, в даному випадку M-LDB втрачає свою точність – це значна зміна інтенсивності освітлення об’єкту. Особливо така проблема проявляється при використанні декількох джерел освітлення. При зміні одного з джерел освітлення може змінитися напрямок градієнту, що вже призведе до неточності. На практиці ця проблема спостерігається при аналізі денних та нічних зображень місцевості, де іноді призводить до повної неможливості розпізнати ті або інші об’єкти. В таких ситуаціях напрямки градієнту на границях неба та землі змінюються на протилежні. Схожа ситуація відбувається з будь яким об’єктом що випромінює світло, та іншими об’єктами, які цим світлом освітлюються [45]. Не враховуючи досить специфічних умов, такий дескриптор є досить стійким до змін освітлення.

Також його важливою особливістю є можливість зміни розміру. Це є наслідком того, що M-LDB обчислює апроксимацію середнього значення

одних і тих самих областей зображення їх інтенсивності та градієнта, логічні значення, отримані в результаті таких порівнянь, не є незалежними один від одного. Це призводить до ситуації коли не існує інформації, яку несе тільки один біт, на відміну від інших дескрипторів, тому виключення окремих бітів незначно зменшує інформативність дескриптору. Можливе зменшення розміру дескриптора шляхом вибору випадкової підмножини бітів або за допомогою більш складних методів, що можуть якщо не поліпшити результати, то принаймні зменшать обчислювальне навантаження при подальшій обробці без зниження продуктивності [35-38,43-51]. Бітове представлення має певні переваги над представленням у вигляді десяткових дробів, що використовується у деяких інших дескрипторах. Це полягає у простоті обробки та інваріантності для системи та мов програмування, так як його подають у вигляді масиву байтів.

Враховуючи всі переваги та особливості описаного поєднання детектору та дескриптору, його було обрано для виконання кваліфікаційної роботи.

Так як дескриптор M-LDB може змінювати розмір до 486 бітів [9], у роботі буде використовуватися максимально можлива довжина, ці ключові точки унікальні майже для кожного зображення, тому за ними можна їх розрізняти.

Якщо розглянути приклад (рис. 3.1), можна побачити, що навіть за розташуванням ключові точки знаходяться певними групами. Точки, що знаходяться в схожих умовах, наприклад грива коня, можуть мати як схожі, так і відмінні між собою дескриптори. Зрозуміло, що можна зробити системи класифікації, яким не потрібно зберігати всі дескриптори зображення. Схожість описів таких точок очікується через однаковість напрямків градієнту на певних частинах зображення. Наприклад, волоски мають однаковий або достатньо схожий напрям градієнту, так як розташовані паралельно один одному. А при формуванні дескрипторів та ключових точок, особлива увага приділяється саме йому.

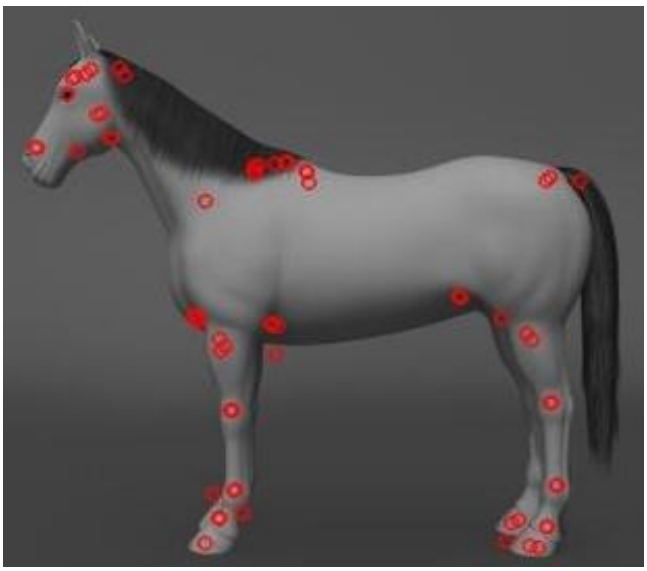

Рисунок 3.1 – Приклад візуалізації ключових точок

Розглянемо дані прикладу: розмір зображення – 308 × 274, число ключових точок – 65.

На прикладі цього зображення можна побачити, що досить велика кількість точок може знаходитись на відносно невеликій частині зображення та згрупована в певні групи.

Таким чином у роботі буде використовуватись алгоритм знаходження ключових точок AKAZE, та алгоритм формування дескрипторів M-LDB. Всі використані алгоритми вже були реалізовані в бібліотеці OpenCV [9].

Розроблений застосунок включає в себе чотири власноруч створених компоненти (дві бібліотеки, дві програми – одна з графічним інтерфейсом для їх використання).

У першій бібліотеці реалізовані операції з обробки зображень, розбиті по таким класам:

– знаходження дескрипторів на його основі;

– знаходження матриці відстаней, як і всередині одного зображення, так і між двома різними, в залежності від ситуації;

– обрахування відстані Танімото/Жакарра на основі як матриць, так і множини дескрипторів зображень;

– створення розподілу на основі матриці, базуючись на довжині одного

інтервалу або їх кількості;

– запис обчисленого розподілу в файл, використовуючи текстовий формат для подальшого ручного аналізу, хоча зазвичай в таких ситуаціях краще використовувати бінарний формат, так як потрібно зберігати числові значення;

– окремо створено інтерфейс для реалізації та демонстрації відстаней, реалізовано відстань Геммінга на його основі;

Інша бібліотека виконує одну основну функцію щодо порівняння отриманого розподілу з еталоном, отримуючи перший з попередньо сформованого файлу, вона може зіставляти дані розподілу як з одним, так і з декількома еталонами одночасно.

Всі дії по обчисленню розподілу та створення розподілу (еталону) виконувалося в окремій програмі, яка зчитувала зображення, вибирала кількість інтервалів та записувала результат в файл. Приклад роботи програми показано на рисунку 3.2.

Рисунок 3.2 – Приклад роботи програми під час виконання роботи (розбиття на квартилі)

Отримано файл, в якому описувалися довжина інтервалів, номер першого та останнього збереженого інтервалу, значення кількостей в

інтервалах. Це дозволяло порівнювати виділені характеристики зображень.

Розгляд та порівняння інших використаних метрик виконувалися через консольну програму.

Основою всіх проведених обчислень є відстань Геммінга, також аналізувалися такі відстані: Евклідова, Мангетенська та Мінковського.

Так як для бінарних слів ці відстані дають однаковий результат, була порівняна їх швидкодія та простота реалізації, вибрана найкраща з них і наведена у лістингу 3.1.

Лістинг 3.1 Реалізація відстані Геммінга:

```
int solve(byte A, byte B) {
    int count = 0;
    for (int i = 0; i < 8; i++) {
        if (((A >>> i) & 1) != ((B >>> i) & 1)) {
            count++;
        }
    }
    return count;
}

public int distance(byte[] A, byte[] B) {
    int count = 0;
    for(int i = 0; i< A.length;i++){
        count += solve(A[i],  B[i]);
    }
    return count;
}
```

Через використання побітових операції в купі з тим, що на один елемент вектору виділяється один біт інформації, досягається велика

швидкодія. Така реалізація має певний недолік, що полягає в тому, що вона може працювати лише з двійковими словами, і не може бути використана для порівняння слів, що складаються з букв, цифр та іншого.

Ця відстань може використовуватися для виконання двох головних задач – обчислення матриці відстаней та відстані Танімото.

Для оцінювання того, щоб взяти для обрахування відстань Танімото, була оцінена складність алгоритмів.

Основою оцінки ефективності алгоритмів на великих масивах даних є їх складність. Складність – це кількісна характеристика, що відображає споживання алгоритмом ресурсів під час свого виконання.

Зазвичай розглядають часову складність – це наскільки швидко зростає кількість обчислень при зростанні кількості даних.

В основі відстані Танімото та матриці відстаней лежить порівняння одного дескриптора з усіма іншими. Для того, щоб порівняти кожен елемент з усіма іншими, зазвичай потрібно $n*n$ операцій. У відстані Танімото це буде найгіршим випадком, тому що цикл переривається, коли знаходиться дескриптор, що достатньо близький, щоб вважатися еквівалентним, найкращим випадком буде значення n. Матриця відстаней, обчислена для двох різних зображень, вже потребує стійку та більшу кількість обчислень, їх оцінка завжди $n*n$, потрібні також додаткові дії вже на обчислення самої відстані. Для обчислення матриці відстаней через її симетричність потрібно буде у два рази менше обчислювальних дій. Також потрібно врахувати затрати пам’яті на зберігання самої матриці. Через це теоретично стає неефективним обрахування тільки відстані Танімото на основі матриці, тому були реалізовані обидва розглянуті варіанти.

3.2 Результати експериментів

Зображення та їх ключові точки наведені на рисунках 3.3-3.5.

Рисунок 3.3 – Перше зображення та його ключові точки

Рисунок 3.4 – Зображення номер два та його ключові точки

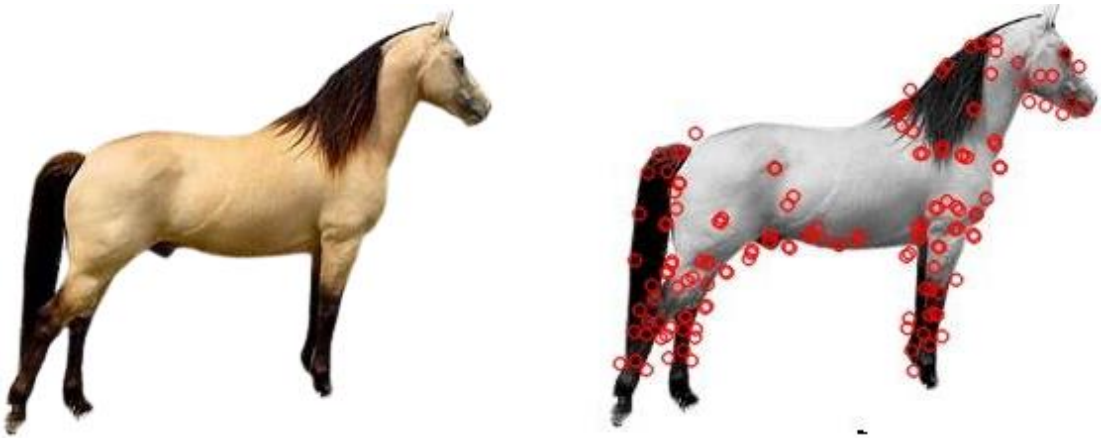

Рисунок 3.5 – Зображення номер три та його ключові точки

Для створення матриці відстаней були використані три різних зображення коней на білому фоні, з яких було узято 40 перших дескрипторів, отримано 1600 відстаней, розподіл яких був показаний на рисунку 3.6, що взагалі є параметром для оцінювання обговорюваної методики.

Хоча на людську думку ці зображення мають деякі спільні риси, програма з використанням OpenCV знаходить між ними досить мало спільних ключових точок.

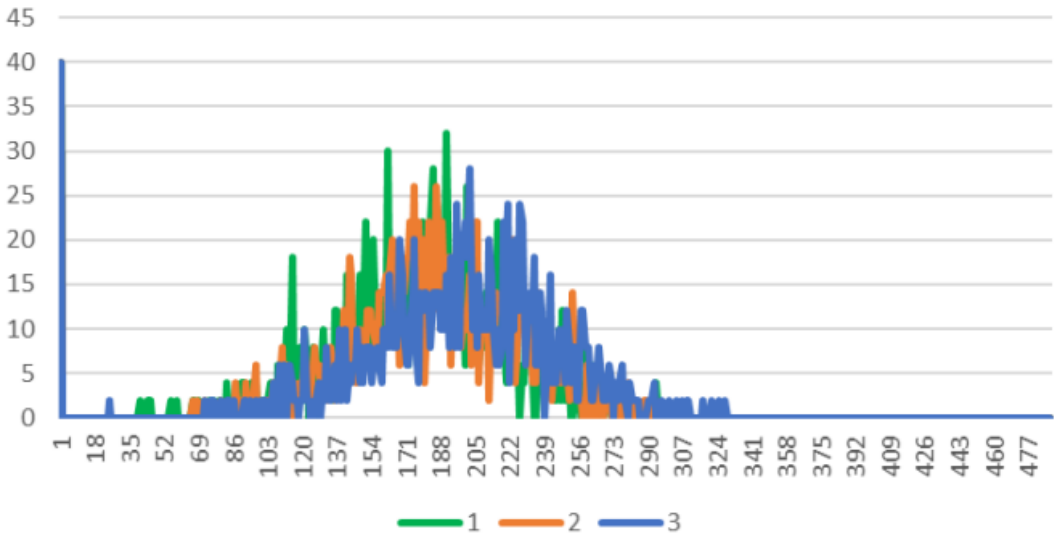

Рисунок 3.6 – Розподіл відстаней в трьох різних зображеннях

Хоч зображення в оригіналі кольорове, дескриптор AKAZE все рівно перетворював його у чорно-біле так як працює тільки з ними.

На основі матриці відстаней побудуємо спочатку класифікаційні ознаки, описані в розділі 2, зокрема, розподіл відстаней.

Розподіли відстаней різних зображень схожі один на одного, та в більшості їх значення згруповані в зоні від 100 до 300. Стовпець нулів не враховується через те, що в кожній розглянутій множині дескрипторів їх рівно 40, так як вони утворюються через порівняння дескриптора з самим собою. Також неможливо не помітити те, що розподіл схожий з нормальним. Це пов’язано з тим, шо розподіл відстаней між випадковими наборами з 488 бітів теж нормальний, що і частково тут зберігається.

Також було проведене укрупнення (рис. 3.7) розглянутих проміжків на предмет збереження особливостей вихідного еталону. Розглянуті діапазони відстаней не можуть бути рівними один з одним, через те, що кількість унікальних значень відстаней 487, а це просте число, яке не можливо розбити на інтервали, відмінні від 1 або 487. Тому один з крайніх інтервалів доведеться робити меншим за всі інші, таким інтервалом у кожному разі краще робити останній, так як великих відстаней в результаті практичного використання не було знайдено, на відміну від маленьких.

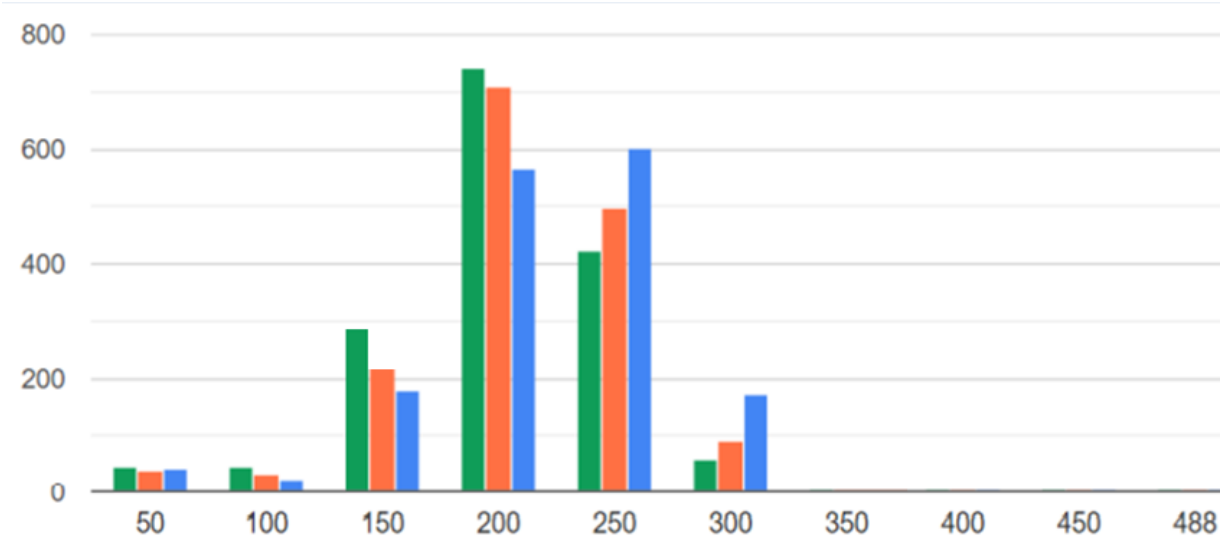

Рисунок 3.7 – Розподіл відстаней в діапазонах

Керування розміром інтервалів може бути корисним не тільки при збереженні інформації, а й при розпізнаванні, так як воно більш стійке до невеликих спотворень в дескрипторах, так як чим більший інтервал, тим менша ймовірність переходу дескриптору з одного в інший. Але й використання в якості еталону теж погіршується, так як збільшується ймовірність випадкового збігу.

Як і при розбитті на окремі дескриптори, тут теж зберігається тенденція візуальної різниці третьої множини від перших двох. Розподіл схожий на нормальний, також тут різко виділяється мода кожного зображення. Можна побачити, що вона більша для третього зображення, і в

цілому значення відстаней у ньому більші.

Як можна побачити з графіку, чотири останні інтервали не мають в собі жодної відстані, а перші два дуже схожі один на одного. Тому вони майже не несуть класифікаційну інформацію, тому вони можуть бути вилучені з розгляду без втрати більшої частини корисної інформації.

Було розглянуто 4 інтервали по 50 елементів з 100 до 300, де були розташовані більшість елементів, а саме для першого – 1508, другого – 1528, а у третього 1518. Також для порівняння було проведено подальше укрупнення, до 122 елементів в інтервалі. Розглянуто також варіант на чотири інтервали (квартилі) довжиною 122, що включали в себе всі можливі відстані. Отримані розподіли відображені на рисунку 3.8.

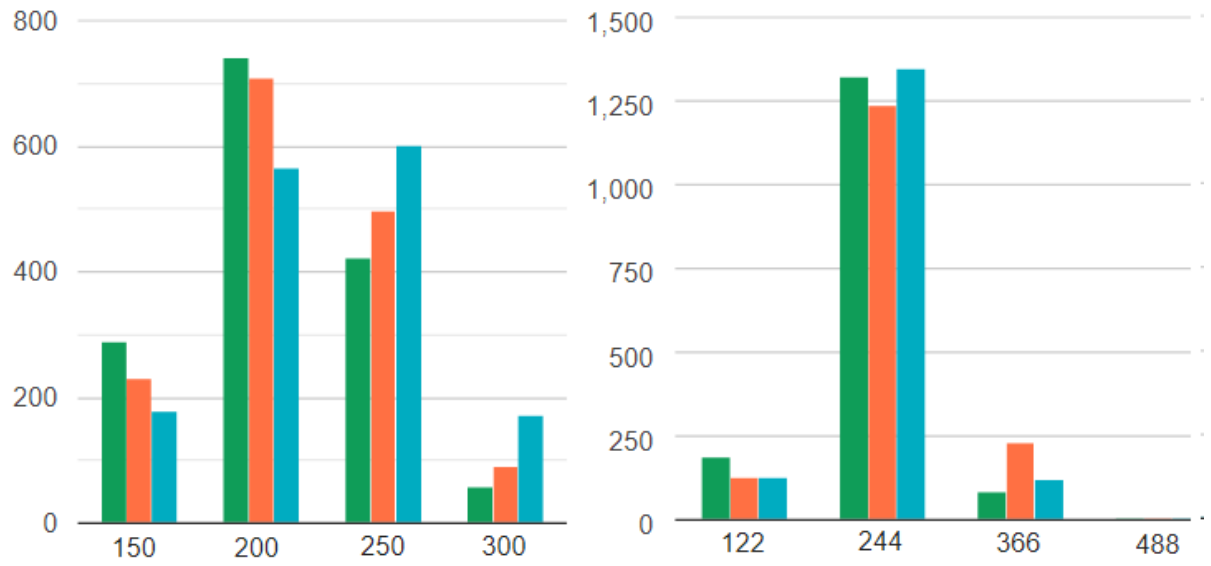

Рисунок 3.8 – Чотири інтервали з розбиття на 10 та на квартилі

Візуально можна побачити, що при розбитті на чотири інтервали менш точно описує картину в цілому, так як перший інтервал майже не несе інформації, у другого і третього він взагалі співпадає, а останній не несе жодної інформації про розподіл.

При цьому відстані між зображеннями у першому випадку збереглися на досить високому рівні: між першим та другим 196, між першим та третім

578, між другим та третім 382. Тут зберігається тенденція високої відстані від другого та першого до третього, і схожості розподілів перших двох.

У другому випадку відстань між зображеннями майже вирівнялась до якогось середнього значення: між першим та другим 292, між першим та третім 128, між другим та третім 220. Тенденція збереження високої відстані від другого та першого до третього, тут вже не зберігається, як і майже не спостерігаються сліди нормального розподілу.

Так як відстань між зображеннями зменшується зі збільшенням довжини інтервалів, потрібно брати якийсь найбільш доцільний для ситуації інтервал, графік зміни відстаней між розподілами зображення номер два та три наведено на рисунку 3.9.

Рисунок 3.9 – Схожість розподілів в залежності від довжини інтервалів

В цілому графік схожий на логарифм, так же швидко зростає та потім повільно збільшується. Його помітною особливістю являється певна циклічність зростань та падінь з подальшим зменшенням частоти та збільшення амплітуди. Це виникає через те, що весь розподіл не завжди націло ділиться на довжину розглянутих інтервалів, тобто іноді існує інтервал, який коротший за всі інші. Через це при однаковій кількості

інтервалів вони можуть бути різної довжини. Коли кількість інтервалів зменшується на один, тоді йде повільне збільшення. Через деякий час знов відстань починає зменшуватися, а інтервали між локальними максимумами збільшується, так як самі інтервали збільшуються, і треба більше ітерацій.

Сума стовпців матриці відстаней теж несе певну інформацію про розподіл дескрипторів, графік сум відстаней представлено на рисунку 3.10. Потрібно пам'ятати, сама послідовність сум не може бути класифікаційними ознаками, так як вони змінюються при зміни порядку дескрипторів.

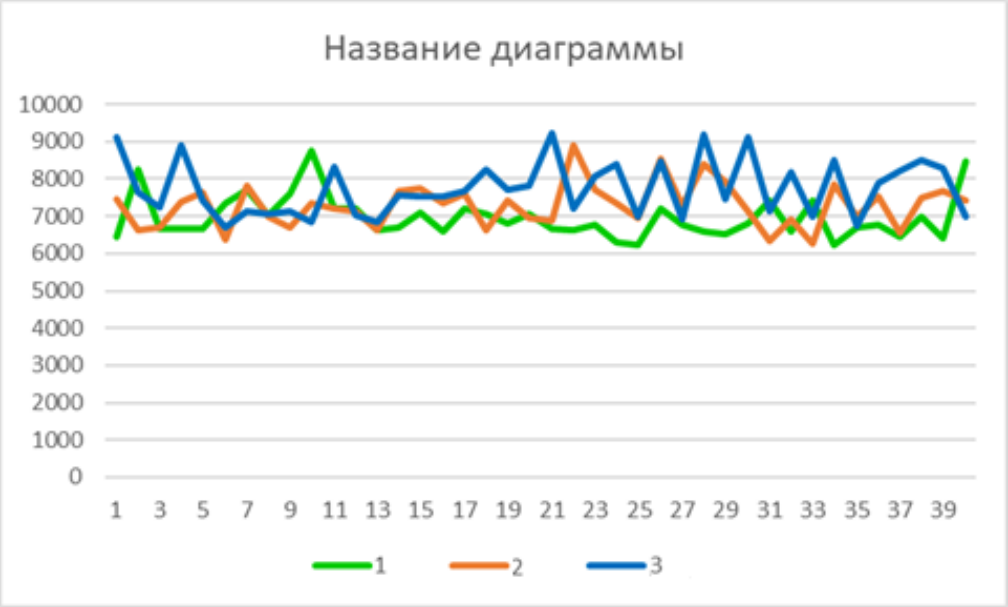

Рисунок 3.10 – Сума відстаней від дескрипторів до інших

На основі графіку (рис. 3.7) легко оцінюються центри класів, що основані на медоїді. У даному випадку центральним дескриптором буде той, де на графіку найглибша впадина, а саме для першого – дескриптор 33 з загальною відстанню 6224, для другого – 25 з відстанню 6251, а для третього – 6 з відстанню 6812. Також можна побачити дескриптори, які вирізняються серед усієї кількості, найбільш відмінні у даному випадку. Для першого – це сумарна відстань 8750 для дескриптору під номером 10, для другого 8917 дескриптор 22, для третього 9220 – дескриптор 22.

Також за цими значеннями можна оцінити те, наскільки далеко

дескриптори зображення в цілому знаходяться один від одного. Можна побачити, що у третьому вони розкидані дальше один від одного, так як сумарна сума візуально більша, а у першому та другому цей розподіл схожий один на одного. Це зберігає тенденцію схожості першого та другого і відповідає графіку з рисунку 3.3, де теж можна побачити, що у третьому більше великих відстаней ніж у другому або першому.

Якщо той самий графік побудувати на основі матриці відстаней між двома множинами, можна оцінити загальний рівень схожості цих двох множин, приклад на ведено на рисунку 3.11.

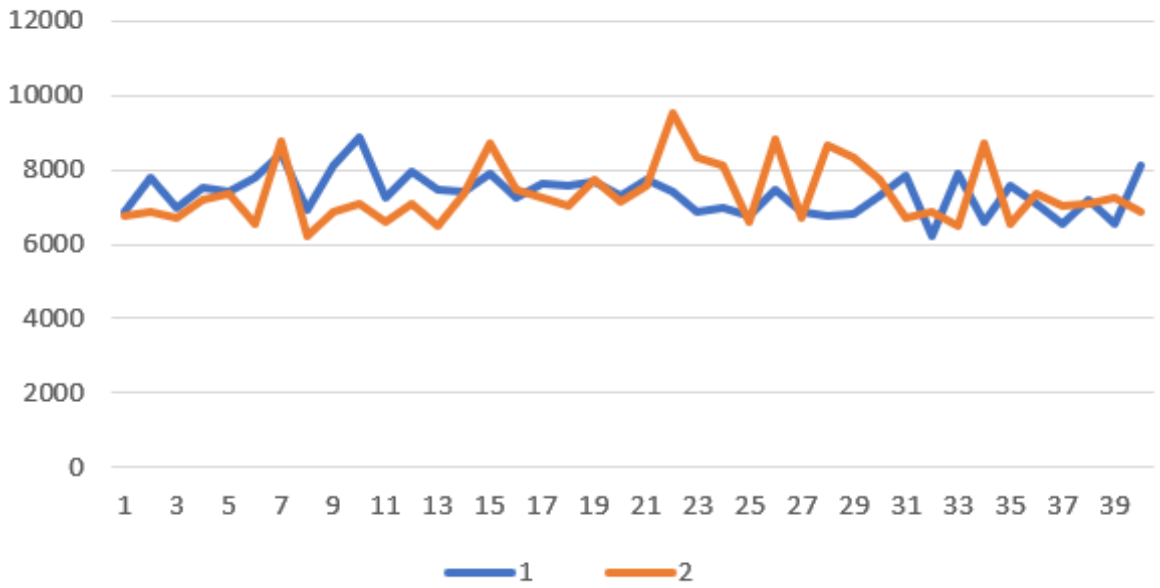

Рисунок 3.11 – Суми стовпців 1 та рядків 2 матриці відстаней

В такому випадку сума стовпців – це буде сума відстаней від дескрипторів першого зображення до дескрипторів другого, а сума рядків – відстань від дескрипторів другого зображення до дескрипторів першого. Тут можна побачити, що зображення два має дескриптори, що знаходяться, як далеко так і дуже близько від дескрипторів першого зображення, а перше зображення у цьому сенсі більш збалансовано.

Як було сказано в попередньому розділі, відстань Танімото/Жакарта більш вдало описує відстань між множинами, тому в роботі вона буде

використовуватися для порівняння множини дескрипторів з еталонами зображень.

Для використання відстані Танімото потрібні множини дескрипторів. Множини за своїм значенням не можуть включати в себе однакові компоненти, в одному зображенні може бути однакові дескриптори, але у цьому випадку так не сталося, тому отримані дескриптори вже були потрібною множиною. Потім потрібно вибрати деяку відстань, на якій різні дескриптори з еталону та порівнюваного масиву будуть вважатися однаковими, та вибрати відстань між множинами, на якій множини будуть вважатися однаковими. Чим менші ці значення, тим більша точність буде у випадках, коли множини дескрипторів не спотворені, але вже не буде розпізнаватись у випадку, коли через перетворення дескриптори змінені.

Відстань Танімото обчислювалася як на основі матриці відстаней, так і на основі множин дескрипторів.

Для обчислення відстані Танімото на основі матриці використовувалися інші матриці відстаней. Ці матриці відстаней будувалась на основі двох різних множин дескрипторів, і на відміну від попередніх, що несуть інформацію про відстані дескрипторів в середині одного зображення, ці матриці несуть інформацію про відстані саме між дескрипторами виключно різних множин. Вони несиметричні та не мають нулів по головній діагоналі, тому можна сказати, що в них більше значень несуть корисну для класифікації інформацію, а головна різниця полягає в тому, що ці матриці вже самі по собі несуть всю інформацію про подібність множин. А для того, щоб оцінити подібність множин у попередньому випадку, потрібно було порівнювати дві матриці. Проте мінусом таких матриць перед попередніми є те, що вони не можуть вказати на певні особливості одного зображень. Так як і матиці інші, то відповідно, розподіли в них теж відрізняються, їх можна побачити на рисунку 3.12.

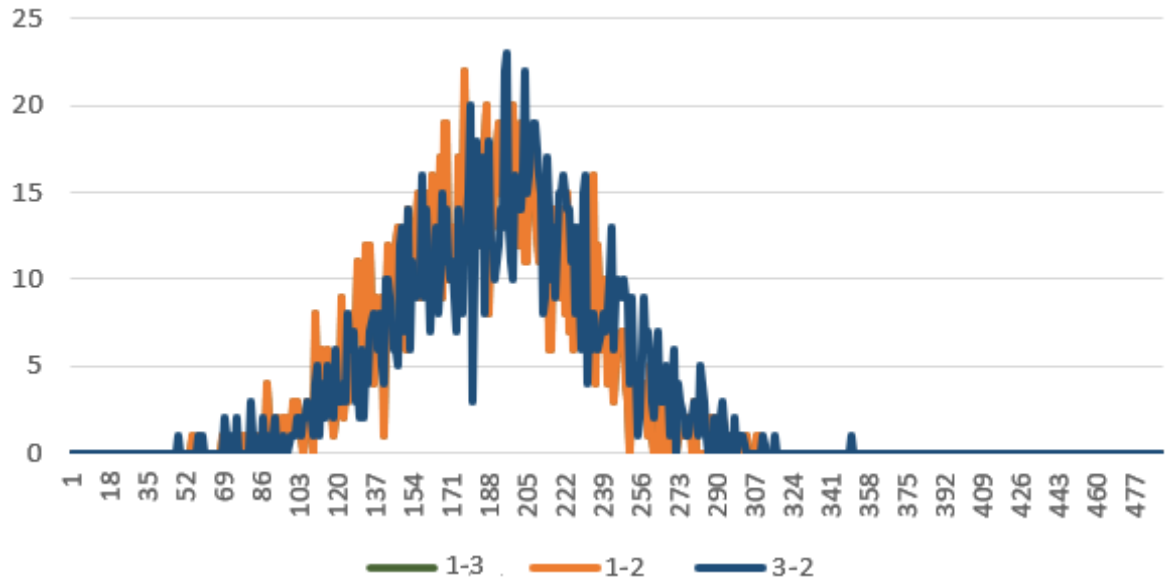

Рисунок 3.12 – Розподіл відстаней в матрицях відстаней між двома різними зображеннями

Одразу можна побачити основну відмінність з попередніми розподілами, це відсутність стовпця з нулями, але основною відмінністю все ж є велика подібність всіх трьох розподілів. Це частково пояснюється тим, що в основі кожної пари розподілів лежить хоча б одне спільне зображення. Також можна підкреслити менший діапазон відстаней та відповідно більшу кількість кожного значення.

Для кожного з цих методів була спочатку обрана відстань у 122, для якої дескриптори вважають рівноцінними. Вона була вибрана на основі того, що випадкове збігання між дескрипторами в середньому 50%, тому було взято половина від цієї кількості. Через нормальність розподілу при випадковому формуванні множин дескрипторів настільки малих відстаней майже не було.

Відстань Танімото було обраховано двома способами: на основі множин дескрипторів, так і на основі матриць відстаней між множинами дескрипторів. Відстань Танімото між розглянутими зображеннями склала відповідно: між першим та другим 0,26, першим та третім – 0,43, другим та третім – 0,49. Відстані Танімото розраховані на основі матриці відстаней та

безпосередньо на множині дескрипторів співпадають. Пари співпадаючих дескрипторів з відстанню менше 122, на основі яких були знайдені відстані Танімото, зображені на рисунках 3.13-3.15

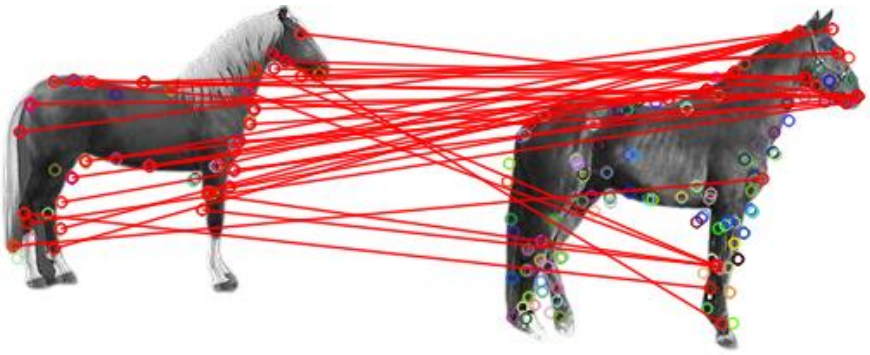

Рисунок 3.13 – Пари близьких точок зображень 1 та 2

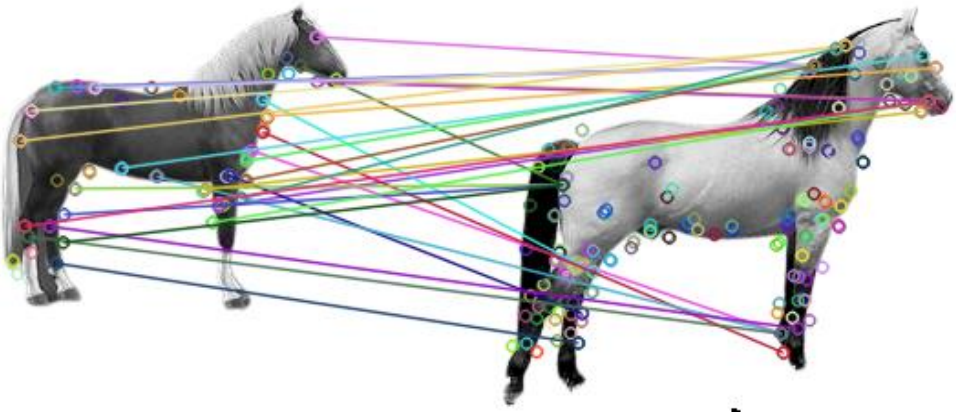

Рисунок 3.14 – Пари близьких точок зображень 1 та 3

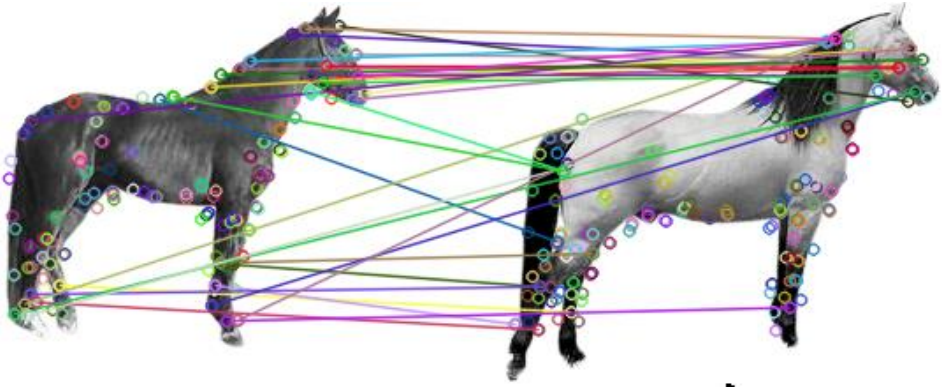

Рисунок 3.15 – Пари близьких точок зображень 2 та 3

Видно, що відстань вказує на близькість дескрипторів першого та другого зображень та їх більшу відстань від третього. Відстань 0,26, на мій погляд, дуже маленька для різних зображень, тому було проведена зменшення граничної відстані у два рази до 61, отримано відстані 0,97, 0,98, 0,97.

Подальше зменшення цієї відстані призвело до неможливості хоч якось порівняти зображення, так як вона завжди дорівнювала одиниці. Таке відбувається, коли нема жодної пари дескрипторів, які б знаходилися на потрібній відстані один від одного.

Так як вони достатньо відрізняються, дуже великі або досить малі, з цього робиться висновок, що існує певний інтервал, при якому відстань Танімото можна використовувати для порівняння ступеню схожості множин дескрипторів для різних зображень. Якщо брати до уваги частково змінені зображення, то відстань там буде меншою. Для практичної демонстрації такого інтервалу був побудована крива зміни відстаней між розподілами при зміні граничної відстані, що наведено на рисунку 3.16.

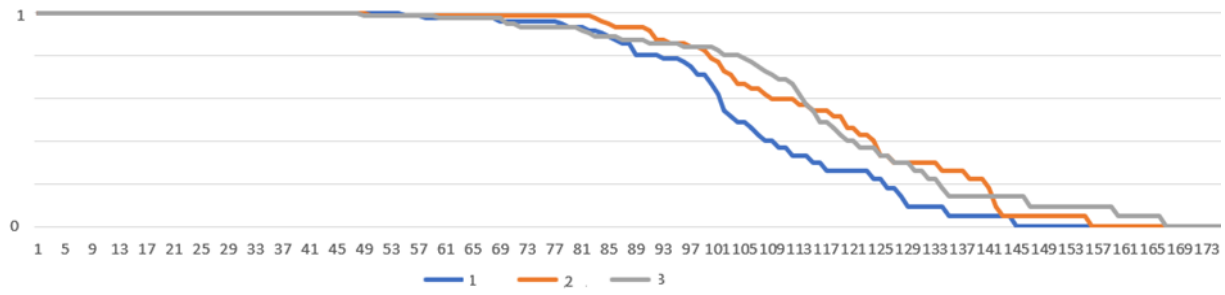

Рисунок 3.16 – Графік зміни відстані Танімото зі зміною граничної відстані

Можна побачити, що для всіх наборів дескрипторів діапазон, у якому відстань Танімото для всіх порівнянь відмінна від нуля або одиниці, приблизно від 50 до 160. Потенційно порівняння ступені близькості наборів дескрипторів ефективніше робити на цьому діапазоні значень. Якщо брати відстані з іншого діапазону, випадку співпадання не буде, так як з високою ймовірністю всі відстані будуть одиницями або нулями.

Але потрібно враховувати, що для побудови графіків були використані

різні зображення. У випадку коли одне зображення отримане шляхом перетворення іншого, приклад такого зображення наведено на рисунку 3.17, значення такого інтервалу можуть відрізнятися. Такі розподіли наведено на рисунку 3.18.

Рисунок 3.17 – Зображення один після перетворення

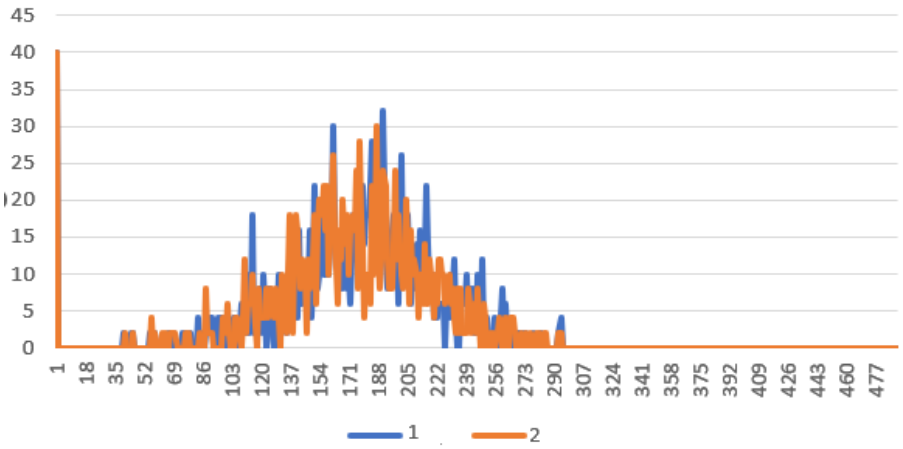

Рисунок 3.18 – Розподіл відстаней у оригінального та перетвореного зображень

Навіть візуально можна побачити високий рівень схожості цих множин. Відстань між розподілами склала 860. Якщо розбити на 10 інтервалів, то відстань між ними буде рівною 112. Тобто ця відстань менша, ніж була при різних зображеннях навіть при чотирьох інтервалах.

Відстань Танімото між цими двома зображеннями відповідно теж менша, так як є більше близьких дескрипторів (рис. 3.19). Це було перевірено

при використанні у якості допустимого максиму відстані значення 122, відстань Танімото дорівнює 0, а при 61 – 0,36. Такий результат був очікуваним, так як, на основі характеристик детектору, дескриптори повинні були спотворитися незначно. Деяка особливість не відповідала очікуванням, вона полягала у відсутності повністю співпадаючих дескрипторів. Найменша відстань між дескрипторами в цьому випадку склала 2.

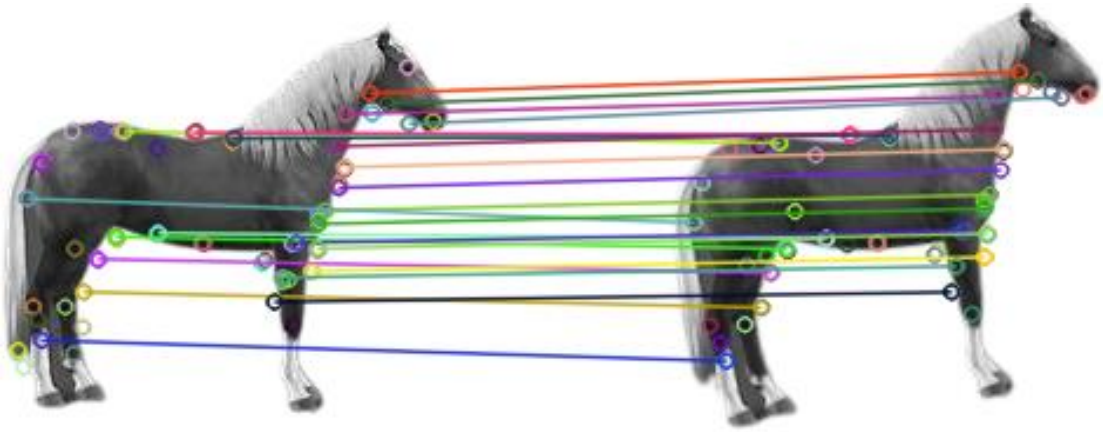

Рисунок 3.19 – Пари точок з оригінального та перетвореного зображення, які знаходяться на відстані 14 або менше один від одного

В даному випадку результат знаходження пар точок навіть при значенні максимальної відстані 14. Візуально ці точки відповідають одна іншій. З цього робиться висновок, що краще при оцінці близькості наборів дескрипторів брати невеликі значення. Середнє значення мінімальних відстаней кожного рядка склало 19,5.

Все ж таки ці відстані набагато менші, ніж між різними зображеннями. Відстань Танімото в цьому випадку швидше починає зменшуватись на початку, але довгий проміжок не змінюється. Те, що діапазони при різних зображеннях та цьому випадку не співпадають та майже не пересікаються, можна зробити висновки, що при порівнянні множин при використанні більших значень граничної відстані зображення, утворені шляхом перетворень оригіналу, зазвичай будуть мати набагато меншу відстань та сильно вирізнятися серед усіх інших.

Графік зміни відстаней між розподілами при зміні граничної відстані наведено на рисунку 3.20.

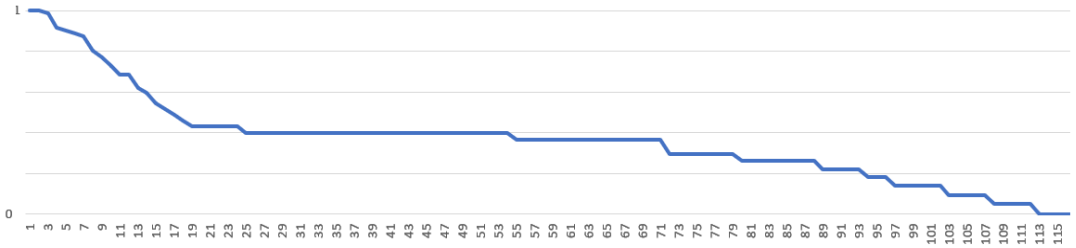

Рисунок 3.20 – Зміна відстані Танімото між зображенням та оригіналом

Це включає і випадки, коли множини утворені на різних зображеннях, і ті, коли одне зображення є спотвореним іншим.

Окремо можна виділити параметри розподілів відстаней, такі як медіана, мода, середні значення, дисперсія та відхилення. Базуючись на отриманих розподілах, значущими для порівняння множин є тільки середнє значення, дисперсія та відхилення. Це пов'язане з тим, що при розподілі не спостерігалось постійного монотонного зростання або спадання. Велика кількість локальних максимумів або мінімумів створює ситуацію, коли мода може досить суттєво змінюватися навіть при невеликих змінах в розподілі. При цьому потрібно враховувати, що дисперсія та відхилення демонструють не приблизну точку знаходження відстаней, а тільки довжину інтервалу, в якому більшість відстаней знаходяться, що не може гарантовано вказувати на близькість розподілів.

Всі ці параметри були практично обчислені для кожного з розподілів.

Середні значення, для зображень склали: 174,1 для першого, 182,1 для другого, 193,7 для третього. Моди, з виключенням нульових відстаней, для зображень склали 190, 174, 201. Медіана: 180, 185, 199. Дисперсії та середні квадратичні відхилення відповідно: 2446,9(49,3), 2417,3(49,1), 2870,6(53,6).

Ці значення, окрім моди, зберігають закономірність невеликої різниці

між першим та другим та їх більшої відстані до третього. Всі вони, не мають суттєвої значущості при класифікації.

Дисперсії в даному випадку сильно відрізняються. Потрібно зауважити, що це не є наслідком того, що розподіли сильно відрізняються. Це відбувається через те, що діапазон розподілу основної кількості дескрипторів більший, ніж у першого і у другого. Перший та другий розподіл мають близькі за розміром діапазони, це можна побачити на рисунку 3.6. Базуючись на цьому, збереження тенденції можна вважати випадковим.

Медіана та середнє значення, як і очікувалося, вказують приблизно на центри розподілу основної кількості відстаней, хоча для першого зображення медіана та середнє сильно відрізняються.

ВИСНОВКИ

У результаті роботи були сформовані, застосовані та досліджені класифікаційні ознаки, створені на основі матриці відстаней для опису зображення у вигляді множини дескрипторів ключових точок.

Порівняння на основі таких ознак продемонструвала результат, схожий з результатом за відстанню Танімото. Було продемонстровано, що точність порівняння зберігається навіть при укрупненні розглянутих діапазонів відстаней.

Практично було встановлена суттєва різниця на діапазоні від 100 до 300. Цей факт дає можливість прискорити обчислення, розглядаючи тільки цей діапазон.

Новизну складає дослідження апарату та моделей формування ознак, основаних на матриці відстаней, що сформована на множині бітових дескрипторів ключових точок зображення. З метою здійснення аналізу таких даних, що застосовувалось для здійснення класифікації зображень, сприяючи спрощенню процесу та зниженню обчислювальних витрат.

Практична значущість полягає у побудові моделей класифікації на підставі матриці відстаней, та підтвердження ефективності їх модифікації на тестових зображеннях. Створення програмного застосунку для впровадження цих класифікаторів у системи.

Перспективи роботи можуть полягати у опрацюванні інших класифікаційних моделей на основі отриманих описів.

Результати роботи апробовано у вигляді статті у журналі Advanced Information Systems та тез доповідей під час Міжнародного молодіжного форуму «РАДІОЕЛЕКТРОНІКА І МОЛОДЬ У XXI СТОЛІТТІ» [38].

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Everything You Ever Wanted To Know About Computer Vision. URL: https://towardsdatascience.com/everything-you-ever-wanted-to-know-about-computer-vision-heres-a-look-why-it-s-so-awesome-e8a58dfb641e (дата звернення 14.04.2023).

2. Image recognition. URL: https:// www.techtarget.com/searchenterpriseai/definition/image-recognition (дата звернення 10.04.2023).

3. What I learned from competing against a ConvNet on ImageNet. URL: http://karpathy.github.io/2014/09/02/what-i-learned-from-competing-against-a-convnet-on-imagenet/ (дата звернення 16.04.2023).

4. Tutorial 2: Image Matching. URL: https://ai.stanford.edu/~syyeung/cvweb/tutorial2.html (дата звернення 26.04.2023).

5. How to choose CNN Architecture MNIST. URL: https://www.kaggle.com/code/cdeotte/how-to-choose-cnn-architecture-mnist (дата звернення 05.05.2023).

6. Keypoints detection and tracking. URL: https://foundationsofdl.com/2020/10/23/keypoints-detection-and-tracking/ (дата звернення 21.04.2023).

7. Human Pose Detection. URL: https://medium.com/@samim/human-pose-detection-51268e95ddc2 (дата звернення 21.04.2023).

8. Liu, C., Xu, J., & Wang, F. (2021). A review of keypoints' detection and feature description in image registration. Scientific Programming, 2021, 1-25.

9. OpenCV Java documentation (4.7.0-dev). URL: https://docs.opencv.org/4.x/javadoc/index.html (дата звернення 10.04.2023).

10. Gorokhovatsky, V. O., Pupchenko, D. V., & Solodchenko, K. G. (2018). Аналіз властивостей, характеристик та результатів застосування новітніх

детекторів для визначення особливих точок зображення. Системи управління, навігації та зв’язку. Збірник наукових праць, 1(47), 93-98.

11. Introduction to SIFT (Scale Invariant Feature Transform). URL: https://medium.com/data-breach/introduction-to-sift-scale-invariant-feature-transform-65d7f3a72d40 (дата звернення 06.05.2023).

12. Andersson, O., & Reyna Marquez, S. (2016). A comparison of object detection algorithms using unmanipulated testing images: Comparing SIFT, KAZE, AKAZE and ORB.

13. Gorokhovatskiy, V. A. (2016). Efficient Estimation of Visual Object Relevance during Recognition through their Vector Descriptions. Telecommunications and Radio Engineering, 75(14).

14. Ibrahim, D. Y., Gorokhovatskyi, V., Tvoroshenko, I., & Zeghid, M. (2022). Cluster representation of the structural description of images for effective classification.

15. Гороховатський, В.А., Путятін, Е.П. (2008) Структурне розпізнавання зображень на основі моделей голосування ознак характерних точок. Реєстрація, зберігання і обробка даних. 4, 75–85.

16. Гороховатський В.О. (2014) Структурний аналіз та інтелектуальне оброблення даних в комп’ютерному зорі: монографія.

17. Gorokhovatskyi, V., Tvoroshenko, I., Kobylin, O., & Vlasenko, N. (2023). Search for Visual Objects by Request in the Form of a Cluster Representation for the Structural Image Description. Advances in Electrical and Electronic Engineering, 21(1), 19-27.

18. Face Key-point Recognition Using CNN. URL: https://www.analyticsvidhya.com/blog/2021/07/face-key-point-recognition-using-cnn/ (дата звернення 25.04.2023).

19. Gorokhovatskyi, O., Peredrii, O., Gorokhovatskyi, V., & Vlasenko, N. (2023). Explanation of CNN image classifiers with hiding parts. In *Explainable Deep Learning AI* (pp. 125-146). Academic Press.

20. Gorokhovatskiy, V. A. (2011). Compression of descriptions in the structural image recognition. Telecommunications and Radio Engineering, 70(15).

21. Daradkeh, Y. I., Gorokhovatskyi, V., Tvoroshenko, I., & Zeghid, M. (2022). Tools for Fast Metric Data Search in Structural Methods for Image Classification. IEEE Access, 10, 124738-124746.

22. Gorokhovatskyi, O., Gorokhovatskyi, V., & Peredrii, O. (2018). Analysis of application of cluster descriptions in space of characteristic image features. Data, 3(4), 52.

23. Shklovets A.V., Axak N.G. (2012) Distance determination between points on the piecewise-smooth Kohonen maps. Bionics of Intelligense: Sci. Mag. 1(78). 63-67.

24. Gorokhovatsky, V., Stiahlyk, N., & Tsarevska, V. (2021). Combination method of accelerated metric data search in image classification problems. Advanced Information Systems, 5 (3), 5-12.

25. Gadetska, S., Gorokhovatskyi, V., Stiahlyk, N., & Vlasenko, N. (2022). Aggregate Parametric Representation of Image Structural Description in Statistical Classification Methods.

26. Szeliski, R. (2022). Computer vision: algorithms and applications. Springer Nature.

27. Гороховатський, В. О., & Творошенко, І. С. (2022). Аналіз багатовимірних даних за описом у формі множини компонент.

28. Курченко О.О., Рабець К.В. (2011). Метричні простори у курсі математичного аналізу: навч. посібник.

29. Flach, P. (2012). Machine learning: the art and science of algorithms that make sense of data. Cambridge university press.

30. Gorokhovatskyi, V., Gadetska, S., & Stiahlyk, N. (2020). Image structural classification technologies based on statistical analysis of descriptions in the form of bit descriptor set.

31. Гороховатський, В. О., & Гадецька, С. В. (2020). Статистичне оброблення та аналіз даних у структурних методах класифікації зображень.

32. Intersection over Union (IoU) in Object Detection & Segmentation. URL: https://learnopencv.com/intersection-over-union-iou-in-object-detection-and-segmentation/ (дата звернення 27.04.2023).

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

34. Піцун, О. Й. (2018). Методи і засоби опрацювання біомедичних зображень в системах автоматизованої мікроскопії (Doctoral dissertation).

35. Tymchyshyn R., Volkov O., Gospodarchuk O., Bogachuk Yu.(2018) Modern Approaches to Computer Vision. Control systems and computers. 6. 46-73.

36. Гороховатський, В. О., Передрій, О. О., Творошенко, І. С., & Марков, Т. Є. (2023). Матриця відстаней для множини компонентів структурного опису як інструмент для створення класифікатора зображень.

37. Yakovleva O., and Nikolaieva K. (2020) Research of descriptor based image normalization and comparative analysis of SURF, SIFT, BRISK, ORB, KAZE, AKAZE descriptors. Advanced Information Systems, 4 (4), 89-101.

38. Марков Т.Є.(2023) Класифікація зображень за частковим поданням.

39. Гороховатський В.О., Передрий Е.О. (2009) Кореляційні методи розпізнавання зображень шляхом голосування систем фрагментів. Радіоелектроніка, інформатика, управління, 1(20), 74–81.

40. Гороховатський, В.О., Путятін, Е.П., Столяров В.С.(2017) Дослідження результативності структурних методів класифікації зображень із застосуванням кластерної моделі даних. Радіоелектроніка, інформатика, управління. 42. 78–85.

41. Гороховатський, В. О., Творошенко, І. С., & Сидоренко, Д. (2021). Класифікація зображень із використанням кластерного подання.

42. Gorokhovatsky, V.O. and Gadetska, S.V. (2019) Determination of Relevance of Visual Object Images by Application of Statistical Analysis of Regarding Fragment Representation of their Descriptions, Telecommunications and Radio Engineering, 78 (3), 211–220.

43. Gorokhovatskyi, V., Gadetska, S., & Stiahlyk, N. (2020). Image structural classification technologies based on statistical analysis of descriptions in the form of bit descriptor set.

44. Alcantarilla, P. F., & Solutions, T. (2011). Fast explicit diffusion for accelerated features in nonlinear scale spaces. IEEE Trans. Patt. Anal. Mach. Intell, 34(7), 1281-1298.

45. New local difference binary image descriptor and algorithm for rapid and precise vehicle visual localization. URL: https:// ietresearch.onlinelibrary.wiley.com/doi/full/10.1049/iet-cvi.2018.5203 (дата звернення 21.04.2023).

46. Gorokhovatskyi, V., Tvoroshenko, I., & Chmutov, Y. (2022). Застосування систем ортогональних функцій для формування простору ознак у методах класифікації зображень. Advanced Information Systems, 6(3), 5-12.

47. Gadetska, S. V., Gorokhovatskyi, V. O., Stiahlyk, N. I., & Vlasenko, N. V. (2021). 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, (4), 58-68.

48. Gorokhovatskyi, V., Gadetska, S., & Ponomarenko, R. (2020). Recognition of visual objects based on statistical distributions for blocks of structural description of image. In Lecture Notes in Computational Intelligence and Decision Making: Proceedings of the XV International Scientific Conference “Intellectual Systems of Decision Making and Problems of Computational Intelligence”(ISDMCI'2019), Ukraine, May 21–25, 2019 15 (pp. 501-512). Springer International Publishing.

49. Gorokhovatskyi V.A., Zamula A.A. (2016) Employment of Intelligent Technologies in Multiparametric Control Systems. Telecommunications and Radio Engineering. 19(75), 1775–1785.

50. Гороховатський В.О., Гадецька С.В., Стяглик Н.І., Власенко Н.В. (2020) Класифікація зображень на підставі ансамблю статистичних розподілів за класами еталонів для компонентів структурного опису. Радіоелектроніка, інформатика, управління, 4 , 85–94.

51. Gorokhovatskyi V.A. (2018) Image Classification Methods in the Space of Descriptions in the Form of a Set of the Key Point Descriptors. Telecommunications and Radio Engineering, 77 (9), 787-797.