

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Західноукраїнський національний університет
Факультет комп'ютерних інформаційних технологій
Кафедра комп'ютерної інженерії

Вдодович Олександр Вікторович

«Порівняння контурів зображень в нечіткій метриці Фреше / Comparison of image contours in the fuzzy Frechet metric»

спеціальність: 123 - Комп'ютерна інженерія
освітньо-професійна програма - Комп'ютерна інженерія
Кваліфікаційна робота

Виконав студент групи КІм-21
Вдодович Олександр Вікторович

Науковий керівник:
д.т.н., проф., Березький О.М.

Кваліфікаційну роботу допущено до захисту:

" ___ " _____ 20__ р.

Завідувач кафедри
_____ О. М. Березький

Тернопіль – 2021

ЗМІСТ

1	Аналіз метрик порівняння контурів зображень.....	6
1.1	Представлення контурів зображень	6
1.2	Поняття нечітких множин та операції над ними	17
1.3	Постановка задач дослідження	22
1.4	Висновки до розділу 1	22
2	Метрики Фреше і Хаусдорфа та їх модифікації	23
2.1	Поняття метричних просторів	23
2.2	Відстань Хаусдорфа	25
2.3	Відстань Громова-Хасудорфа.....	27
2.4	Відстань Канторовича-Рубінштейна	29
2.5	Відстань Фреше	30
2.6	Відстань Громова-Фреше між кривими	32
2.7	Скелетони.....	36
2.8	Відстань Громова-Хаусдорфа.....	38
3	Програмний модуль обчислення відстані між зображеннями.....	45
3.1	Системні вимоги до програмного модуля.....	45
3.2	Архітектура програмного модуля.....	46
3.3	Тестування розробленого програмного модуля	59
3.4	Висновки до розділу 3	68
	Висновки.....	70
	Список використаних джерел.....	71
	Додаток А Довідка про використання	Ошибка! Закладка не определена.
	Додаток Б Світлокопії виданих публікацій	81

ВСТУП

Актуальність теми. Контур – це зовнішня границя об'єкта. Контур несе в собі інформацію про форму об'єкта, його геометричні характеристики. Для багатьох задач форма об'єкта є визначальною для його розпізнавання. Існують різні способи опису зовнішніх границь об'єктів. Вимоги до опису контурів такі: простота, інформативність, зручність працювати з математичною моделлю контуру.

Оскільки контури несуть в собі багато інформативної інформації, то визначення їх подібності є важливою задачею. Не завжди потрібно знати точну подібність зображень. В багатьох випадках досить знати в якісному вимірі подібність. Наприклад, об'єкти можуть бути зовсім неподібні, мало подібні, дуже подібні. При цьому використовують нечітку логіку. Отже, розроблення алгоритмів визначення подібності об'єктів у нечіткій метриці Фреше є актуальною темою дослідження.

В дослідженні опрацьовано праці викладачів кафедри комп'ютерної інженерії Західноукраїнського національного університету, в яких розглядалися аналіз виділення контурів біомедичних зображень [1-5], алгоритми перетворення контурів зображень [6-10], досліджувалася похибка перетворення контурів біомедичних зображень [11-14].

Мета і завдання дослідження. Метою роботи є розроблення та програмна реалізація алгоритмів визначення відстані між контурами зображень в нечіткій метриці Фреше для оцінки похибок сегментації.

Об'єкт дослідження – процес порівняння зображень.

Предмет дослідження – алгоритми порівняння контурів зображень у метричних просторах.

Для досягнення поставленої мети необхідно розв'язати такі задачі:

1. Проаналізувати метрики оцінки відстаней між зображеннями.
2. Проаналізувати основи нечіткої логіки;

3. Розробити алгоритми пошуку дискретної відстані між контурами у нечіткій метриці Фреше;

4. Розробити модуль для пошуку дискретної відстані між контурами у нечіткій метриці Фреше;

5. Провести тестування розробленого модуля.

Методи дослідження базуються на використанні методів комп'ютерного зору, теорії метричних просторів, об'єктно-орієнтованого програмування.

Наукова новизна одержаних результатів. Розроблено алгоритми порівняння контурів зображень у нечіткій метриці Фреше.

Практичне значення отриманих результатів. Розроблено модуль для оцінки похибок сегментації зображень.

Публікації та апробація КР. За результатами кваліфікаційної роботи опубліковані двоє тез доповідей на V-й науково-практичній конференції молодих вчених і студентів «Інтелектуальні комп'ютерні системи та мережі», яка відбулася на кафедрі комп'ютерної інженерії Західноукраїнського національного університету, 2 грудня 2021 р., м. Тернопіль [15, 16].

Кваліфікаційна робота складається із трьох розділів, висновків, списку використаної літератури та додатків [17, 18].

У першому розділі зроблено аналіз контурів зображень та описано основні положення нечіткої логіки, здійснено постановку задач дослідження.

У другому розділі описано метрики Фреше та Хаусдорфа та їх різновиди для використання при порівнянні зображень.

У третьому розділі розроблено програмний модуль обчислення відстані між зображеннями, сформовано системні вимоги до програмного модуля, розроблена архітектура програмного модуля та проведено тестування розробленого програмного модуля.

У додатках приведено, довідку про використання результатів кваліфікаційної роботи, світлокопію виданих публікацій.

1 АНАЛІЗ МЕТРИК ПОРІВНЯННЯ КОНТУРІВ ЗОБРАЖЕНЬ

1.1 Представлення контурів зображень

На сьогоднішній день аналіз форми об'єктів широко використовується в різних галузях виробництва. В багатьох випадках інформації про форму об'єкта достатньо для організації автоматичних або автоматизованих систем. Автоматизація виробництва дозволяє збільшити кількість та зменшити вплив людського фактора на якість виробленої продукції. Під час проведення біомедичних досліджень важливу роль відіграє аналіз форми досліджуваного об'єкта .

Контур об'єкта – це частина об'єкта, яка містить велику кількість інформації про форму об'єкта та слабо залежить від кольору та текстури зображення [19].

Контур на кольоровому зображенні відповідає перепад яскравості. Проте, це визначення виключає контури, пов'язані з різкими змінами колірному тону та насиченості в областях з постійною яскравістю.

Представлення (кодування) контуру – це етап отримання дискретного сигналу, що описує границі оцифрованого зображення.

Вимоги до алгоритмів представлення контурів:

- зменшення обсягів використовуваної пам'яті для зберігання;
- зменшення часу та складності подальшої обробки;
- одержання інформативних ознак об'єкта.

Кодування контуру об'єкта за допомогою ланцюгових кодів Фрімена. Для опису форми об'єкта необхідно намалювати сітку із заданим розміром комірки на аналоговому зображенні контуру та зафіксувати точки перетину контуру з лініями сітки (рисунок 1.1). Потім послідовність вузлів, найближчих до перетинів, кодується вісімковими числами, що вказують напрямок від одного вузла до іншого згідно з кодами.

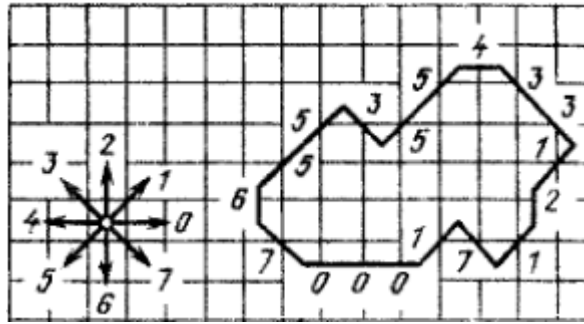


Рисунок 1.1 – Кодування контуру об'єкта за ланцюговими кодами Фрімена

Різновидами кодування контурної функції ланцюговими кодами Фрімена є також кодування, при якому замість цифр 0, 1, ..., 7 використовуються наступні матриці:

$$\begin{array}{ccc}
 -1,-1 & 0,-1 & 1,-1 \\
 -1,0 & \star & 1,0 \\
 -1,1 & 0,1 & 1,1
 \end{array}$$

Для представлення в комплексній площині, використовується матриця:

$$\begin{array}{ccc}
 -1+i & i & 1+i \\
 -1 & \star & 1 \\
 -1-i & -i & 1-i
 \end{array}$$

Перевага такого кодування - простота реалізації, швидкість та точність уявлення контурної функції. До недоліків можна віднести відносно великий обсяг пам'яті, необхідний зберігання опису, а також чутливість до поворотів.

Функція міжцентрової відстані $R(s)$ визначається як відстань від центру об'єкта до кожної точки контуру та розраховується за формулою:

$$R(s) = \sqrt{(x_s - x_c)^2 + (y_s - y_c)^2},$$

де (x_s, y_s) – координати s -тої точки контуру об'єкта;

(x_c, y_c) – координати центру об'єкта.

Кодування елементарними векторами

Елементарний вектор $\gamma(n)$ – це вектор, що сполучає центри сусідніх контурних осередків сітки. Де n – порядковий номер вектора ($n = 0, 1, \dots, k$) (рисунок 1.2). Крок сітки вибирається відповідно до необхідної точності представлення.

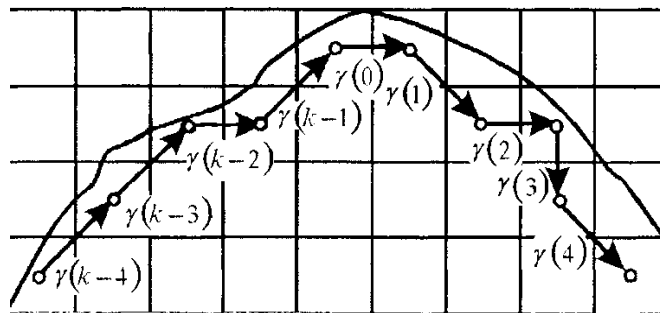


Рисунок 1.2 – Кодування контуру об'єкта елементарними векторами

Елементарне векторне кодування можна розглядати як апроксимацію кривих першого порядку з довжинами стану, що встановився. Перевагами такого підходу до кодування інформації є простота реалізації, низька трудомісткість процесу кодування, невеликі обсяги пам'яті, необхідні зберігання даних, висока точність представлення, простота порівняння двох контурів.

Кодування векторами. Вектори не мають обмеження за довжиною, що дозволяє значно зменшити кількість векторів порівняно з елементарними векторами (рисунок 1.3). Крок сітки залежить від необхідної точності представлення.

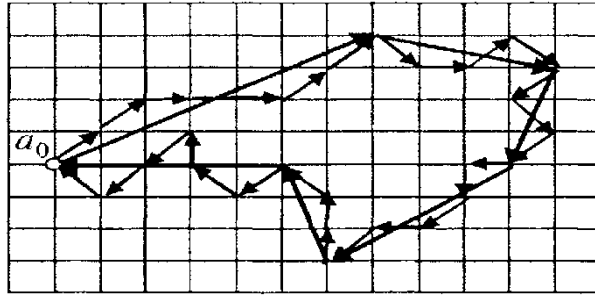


Рисунок 1.3– Кодування контуру об'єкта векторами

Результатом кодування є наступна таблиця характеристик: довжина вектора, напрямок обертання, кут між сусідніми векторами. У певному наближенні опис форми об'єкта за допомогою векторів є наближенням контуру за допомогою кривих першого порядку.

Перевага – невеликий обсяг пам'яті, необхідний для зберігання векторів, і навіть висока точність опису контурної функції. До недоліків відносяться час, що витрачається на обчислення великих векторів, і час, що витрачається на перевірку точності опису.

Побудова опису з урахуванням перетворення Фур'є дуже ефективна для представлення форми замкнутих контурів, тобто періодична функція, отримана при побудові впорядкованої послідовності точок контуру, можна висловити термінами ряду Фур'є. Контур об'єкта можна представити як замкнуту періодичну функцію $y(x)$ з періодом P , де P дорівнює периметру об'єкта, і тому контур можна продовжити за допомогою ряду Фур'є. Тоді дискретне представлення контуру об'єкта з максимальним числом членів N виглядатиме так:

$$x(s) = a_0 + \sum_{n=1}^N [a_n \cos(2\pi ns / P) + b_n \sin(2\pi ns / P)];$$

$$y(s) = c_0 + \sum_{n=1}^N [c_n \cos(2\pi ns / P) + d_n \sin(2\pi ns / P)],$$

де коефіцієнти $a_0, c_0, a_n, b_n, c_n, d_n$ називаються „еліптичними ознаками” та розраховуються наступним чином:

$$a_0 = \frac{1}{N} \sum_{k=1}^N x_k;$$

$$c_0 = \frac{1}{N} \sum_{k=1}^N y_k;$$

$$a_n = \frac{2}{N} \sum_{k=1}^N x_k \cos n\delta;$$

$$b_n = \frac{2}{N} \sum_{k=1}^N x_k \sin n\delta;$$

$$c_n = \frac{2}{N} \sum_{k=1}^N y_k \cos n\delta;$$

$$d_n = \frac{2}{N} \sum_{k=1}^N y_k \sin n\delta,$$

де $\delta = 2\pi \frac{\Delta s}{P}$, Δs - приріст довжини дуги між двома точками контурної послідовності.

Переваги опису форми з використанням коефіцієнтів Фур'є включають відносну простоту реалізації і той факт, що цей опис заснований на розвиненій теорії перетворення Фур'є. Головний недолік - складність опису локальної інформації про форму, а також проблеми з точністю опису симетричних кривих (вони відрізнятимуться лише фазою, яку можна точно розрахувати).

При роботі в комплексній площині контур можна представити за допомогою комплексної координатної функції. Комплексна координатна функція розраховується за такою формулою:

$$Z(s) = (x_s - x_c) + j(y_s - y_c),$$

де (x_s, y_s) – координати s-тої точки контуру об'єкта;

(x_c, y_c) – координати центру об'єкта.

Окрім функції центрових відстаней та комплексних координат також використовується апроксимація контуру кривими першого та другого порядку. Використання кривих третього порядку вимагає виконання більшої кількості обчислювальних операцій, що може сильно впливати на швидкість програми, проте їх використання збільшує якість апроксимації та полегшує процес семантичного аналізу.

Використання даного типу представлення об'єктів дає можливість зменшити кількість необхідної пам'яті для зберігання опису об'єкта. Недоліком даного представлення є велика кількість обчислень, а також похибка. Вхідними даними для подальшої обробки будуть кількість, тип, порядок розташування та коефіцієнти кривих.

Опис за допомогою скелету області. Поняття скелету вперше було запропоноване в роботі Х.Блюма [20-21] як результат перетворення серединних осей або перетворення осей симетрій. В евклідовому просторі скелет визначається як геометричне місце розташування центрів максимальних вписаних в об'єкт кіл для 2D та сфер для 3D об'єктів. Для дискретного випадку поняття скелету можна представити у вигляді метафори „пожежі в степу” [19].

Скелетизація – це процес обробки бінарного зображення об'єкта, результатом якого є скелет. Скелет забезпечує просте і компактне представлення форми, що зберігає більшість топологічних характеристик та розміри оригінальної форми об'єкта. Приклад виділення скелету за допомогою максимально вписаних кіл наведено на рисунку (рисунк 1.4):

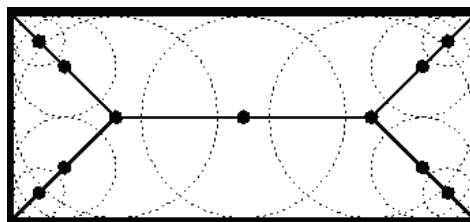


Рисунок 1.4 – Приклад виділення скелету

Основні методи виділення скелетів такі:

- метод просторових трансформацій;
- метод на основі діаграм Вороного;
- метод стоншення.

Метод на основі просторових трансформацій полягає в визначенні ваг точок, які належать об'єкту. Вага точки визначається як мінімальна відстань від точки до контуру об'єкта по горизонталі чи вертикалі (для 4-зв'язності), а також по діагоналях (для 8-зв'язності). Точки з максимальними вагами визнаються такими, що формують скелет об'єкта.

Метод на основі діаграм Вороного характеризується розбиттям вхідного об'єкта на множину підобластей, таких що формуються з точок, які є найближчі до деякої точки. Скелетизація на основі методу діаграм Вороного є трудомістким процесом, особливо на великих та складних об'єктах.

Під час методу стоншення використовуються операції морфологічної обробки зображень. Даний метод полягає у послідовному видаленні точок, які належать контуру об'єкта з метою отримання ліній шириною одну точку. Його перевагами є збереження топології та форми вхідного об'єкта, ширина отриманого скелету рівна одній точці.

Відомі такі алгоритми скелетизації: алгоритм скелетизації на основі визначення топологічних вузлів, на основі визначення просторових моментів, на основі розповсюдження хвилі, алгоритм Зонга-Суня тощо.

Недоліками скелетизації в дискретному зображенні є залежність результатів скелетизації від похибки дискретизації, типу опису зв'язків між сусідніми точками області (4- або 8-зв'язні), характеру (кривизни) контурної лінії області.

Одним з недоліків процесу скелетизації є отримання скелету з великою кількістю надлишкових гілок. Для відсікання таких гілок існують декілька алгоритмів. Зокрема, в роботі пропонується відсікання малоінформативних гілок на основі евристичних правил. Недоліком даного алгоритму є складність

кількісної оцінки роботи алгоритму. В роботі запропонований алгоритм відсікання малоінформативних віток на основі зміни контурної лінії.

У ряді робіт продемонстровано використання скелетизації для роботи з мікрооб'єктами на біомедичних зображеннях: виявлення паразитів (червів) на біомедичних зображеннях, виявлення кров'яних капілярів на зображенні людського мозку та інші.

Перевагами використання опису об'єкта на основі скелетів є: інваріантність до повороту, масштабування, мінімальний вплив дефектів цифрового зображення. Недоліком є складність відновлення об'єкта без додаткових метаданих.

Дескриптори границь.

При аналізі даних отриманих в результаті кодування контурної функції мікрооб'єктів виділяються характерні ознаки контуру (рисунок 1.5) .

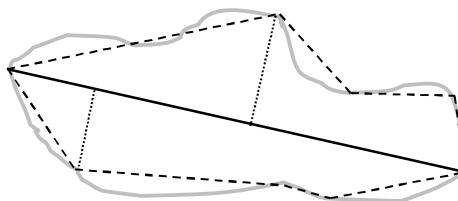


Рисунок 1.5 - Приклад об'єкта та його контурних ознак

Дані ознаки можна розділити на дві групи: характерні ознаки першого та другого порядків. До характерних ознак першого порядку будемо відносити ознаки які обчислюються безпосереднім аналізом контурної функції:

1) Периметр (Perimetr) $P = \sum I_j(x, y)$, де (x, y) - координати контурних точок j -го мікрооб'єкту на зображенні I . Довжина контурної функції. Залежить від типу зв'язності, що використовується під час опису контуру (восьми або чотирьох зв'язність).

2) Площа (Area) $A = \sum I_j(x, y)$, де (x, y) - координати точок j -го мікрооб'єктів на зображенні I . Площа об'єкта – це частина зображення яку обмежує контурна функція.

3) Головна вісь (Main Axis) (H) – відрізок, що з'єднує дві максимально віддалені точки контуру та проходить через центр мас мікрооб'єкту.

4) Моменти – середньозважені інтенсивності пікселів або функції від них. Найбільш поширеними є геометричні моменти, центральні моменти та нормовані центральні моменти, момент інваріанти, - моменти Церніке та моменти Лежандра. В загальному випадку геометричні моменти можна визначити за формулою:

$$m_{pq} = \sum_{x=1}^X \sum_{y=1}^Y x^p y^q f(x, y)$$

де, $f(x, y)$ - значення інтенсивності в точці з координатами x та y .

До ознак другого порядку відносяться ознаки, що отримуються шляхом додаткової обробки або поєднання ознак першого порядку.

5) Співвідношення сторін (Aspect Ratio) $ar = \frac{P_1 + P_2}{H}$. Характеристика, що вказує на протяжність мікрооб'єктів.

6) Колоподібність (Circularity) $cir = \frac{4A}{P^2}$ (рисунок 1.6,а). Характеристика вказує на скільки форма мікрооб'єктів наближається до форми кола.

7) Зміна кута між сусідніми сегментами контуру (Discontinuity Angle Irregularity) $dai = \sqrt{\frac{\sum |\theta_i - \theta_{i+1}|}{2\pi(n-2)}}$. Нормалізоване значення кута абсолютної різниці між кутами сторін багатокутника, що описує контур або його частину.

8) Довжина нерівномірності (Length Irregularity) $lr = \frac{\sum |L_i - L_{i+1}|}{K}$, де $K = 2P$ якщо $n > 3$ або $K = P$ якщо $n = 3$. Нормалізоване значення середньої абсолютної зміни довжини апроксимованого контуру (сегменту) та його первинного представлення.

9) Кількість відрізків/кривих n , необхідних для апроксимації контурної функції мікрооб'єкту. Значення залежить від точності та алгоритмів апроксимації.

10) Складність (Complexity) $com = 10^{\frac{-3}{n}}$. Кількість сегментів у апроксимованому контурі, зміна яких призведе до зміни форми з більшим ефектом в умовах низької складності форми, ніж при високій складності.

11) Прямокутність (Right-Angleness) $RA = \frac{r}{n}$, де r - кількість кутів $\alpha = 90^\circ$ між сусідніми лініями, що описують форму мікрооб'єктів. Характеристика яка визначає подібність форми мікрооб'єкту до прямокутника.

12) Різкість (Sharpness) $sh = \sum \frac{\max(0.1 - (\frac{2|\theta - \pi|}{\pi})^2)}{n}$. Міра різких стрибків зміни контурної функції (понад 90°).

13) Спрямованість (Directedness) $dir = \frac{\sum Paralel(Li)}{\sum Li}$, де $Paralel(Li)$ - сегменти контуру які є паралельні до великої осі.

14) Зовнішній багатокутник. Описує контур у вигляді опуклого багатогранника (рисунок 1.6,б).

15) Зовнішній квадрат (рисунок 1.6,в). Уявний прямокутник, що описує мікрооб'єкт. Серед вторинних ознак які впливають з даної характеристики можна виділити: прямокутник зі сторонами паралельними до осей OX та OY; мінімальний прямокутник що описує об'єкт; координати вершин та центру

описаних прямокутників; площі, периметри та довжини сторін прямокутників тощо.

16) Зовнішній овал (рисунок 1.6,г). Уявний овал, що описує мікрооб'єкт. Серед вторинних ознак які впливають з даної характеристики : координати центру овалу; площа, периметр овалу тощо.

17) Зовнішнє коло (рисунок 1.6,д). Уявне коло, що описує мікрооб'єкт. На основі даної характеристики можна виділити: радіус кола; координати вершин та центру описаного кола; площа, периметр та радіус кола тощо.

18) Максимальний перпендикуляр $Maxep = \max(p_1, p_2)$. Визначає довжину максимального перпендикуляра, опущеного з контуру мікрооб'єкту на велику вісь.

19) Напрямок об'єкта. Кут між великою віссю та віссю OX, але при цьому враховується максимальний перпендикуляр.

20) Апроксимація контуру $Contour = \sum L_i$. Апроксимована контурна функція, відповідає кусково-лінійній апроксимації при кодуванні контуру.

21) Ключові точки. Точки контурної функції в яких відбувається знак приросту функції або кут приросту перевищує деяке порогове значення δ .

22) Відстань між піками контурної функції. Відстань між ключовими точками, які відповідають максимальним значенням.

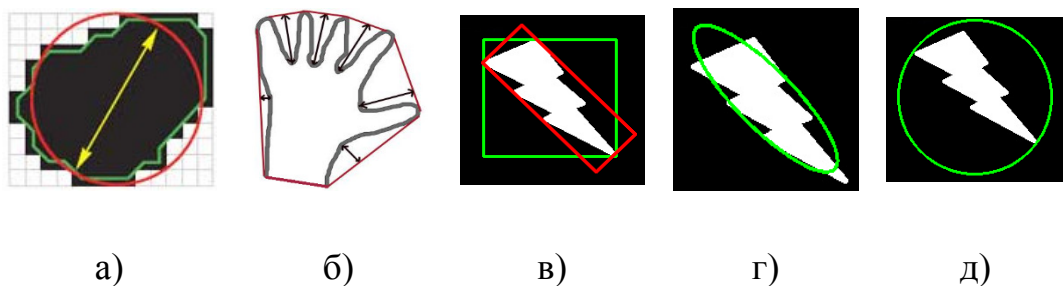


Рисунок 1.6 – Приклади визначення контурних ознак: колоподібність (а), зовнішній багатокутник (б), квадрат (в), овал(г), коло(д)

1.2 Поняття нечітких множин та операції над ними

Нечіткою множиною A в деякому (непустому) просторі \mathbf{X} ($A \subseteq \mathbf{X}$), називається пара, що складається з множини

$$A = \{(x, \mu_A(x)); x \in \mathbf{X}\}, \quad (1.1)$$

де

$$\mu_A: \mathbf{X} \rightarrow [0,1] \quad (1.2)$$

– характеристична або просто функція належності нечіткої множини A . Функція належності вказує рівень належності елемента x до підмножини A . У випадку якщо:

1) $\mu_A = 1$, то значить, що елемент x повністю належить до нечіткої множини A , тобто $x \in A$;

2) $\mu_A = 0$, то – елемент x не належить до нечіткої множини A , тобто $x \notin A$;

3) $0 < \mu_A < 1$, то – елемент x частково належить до нечіткої множини A .

На нечітких множинах визначені наступні операції [22-23]: перетин, сума, доповнення, декартовий добуток, концентрація, розмиття.

1.Перетином двох нечітких множин A , $B \subseteq \mathbf{X}$ називається нечітка множина $C = A \cap B$ з функцією належності:

$$\mu_{A \cap B}(x) = \mu_A(x) \wedge \mu_B(x) = \min(\mu_A(x), \mu_B(x)),$$

для кожного $x \in X$. Графічно перетин зображено на рисунку 1.7.

Перетином декількох нечітких множин A_1, A_2, \dots, A_n називається нечітка множина $A_1 \cap A_2 \cap \dots \cap A_n$ з функцією належності:

$$\mu_{A_1 \cap A_2 \cap \dots \cap A_n}(x) = \bigcap_{i=1}^n \mu_{A_i}(x) = \min(\mu_{A_1}(x), \mu_{A_2}(x), \dots, \mu_{A_n}(x)) \quad (1.3)$$

для кожного $x \in X$.

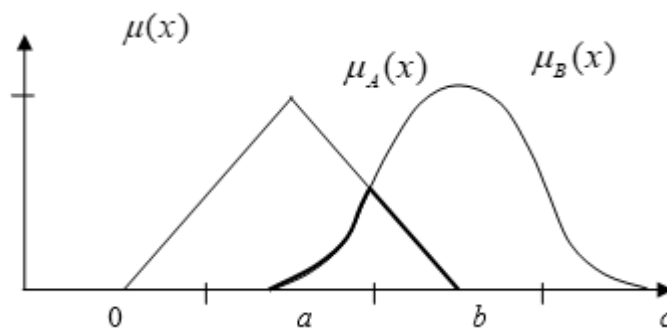


Рисунок 1.7 – Графік перетину нечітких множин A та B

2. Сумою двох нечітких множин $A, B \subseteq X$ називається нечітка множина $C = A \cup B$ з функцією належності:

$$\mu_{A \cup B}(x) = \mu_A(x) \vee \mu_B(x) = \max(\mu_A(x), \mu_B(x))$$

для кожного $x \in X$.

Графік суми зображено на рисунку 1.8.

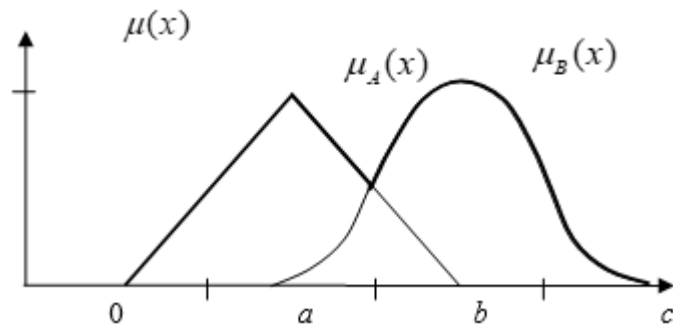


Рисунок 1.8 – Графік суми нечітких множин A та B

Сумою декількох нечітких множин A_1, A_2, \dots, A_n називається нечітка множина $A_1 \cup A_2 \cup \dots \cup A_n$ з функцією належності

$$\mu_{A_1 \cup A_2 \cup \dots \cup A_n}(x) = \bigcup_{i=1}^n \mu_{A_i}(x) = \max(\mu_{A_1}(x), \mu_{A_2}(x), \dots, \mu_{A_n}(x)) \quad (1.4)$$

для кожного $x \in X$.

Приклад

$$X = \{1, 2, 3, 4, 5, 6, 7\}. \quad A = \frac{0,5}{2} + \frac{1}{4} + \frac{0,7}{5}. \quad B = \frac{0,8}{4} + \frac{0,85}{5} + \frac{0,9}{7}.$$

Згідно з означенням

$$A \cap B = \frac{0,8}{4} + \frac{0,7}{5}; \quad A \cup B = \frac{0,5}{2} + \frac{1}{4} + \frac{0,85}{5} + \frac{0,9}{7}.$$

3. Доповненням нечіткої множини $A \subseteq X$ називається нечітка множина \hat{A} з функцією належності:

$$\mu_{\hat{A}}(x) = 1 - \mu_A(x) \quad (1.5)$$

для кожного $x \in X$. Графік доповнення зображено на рисунку 1.9 штриховою лінією.

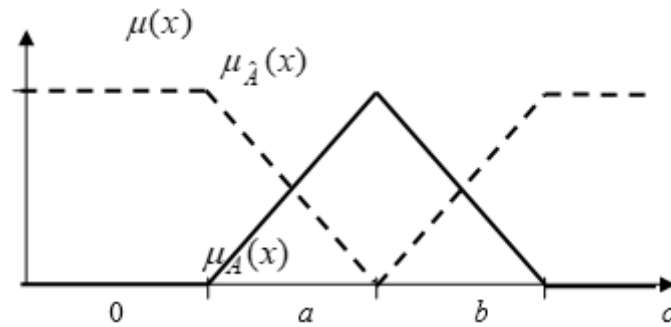


Рисунок 1.9 – Графік доповнення нечіткої множини A

Приклад.

$$\mathbf{X} = \{2,3,4,5,6,7\}. A = \frac{0,5}{2} + \frac{1}{4} + \frac{0,7}{5}. \hat{A} = \frac{0,5}{2} + \frac{1}{3} + \frac{0,3}{5} + \frac{1}{6} + \frac{1}{7}.$$

Треба відзначити, що

$$A \cap \hat{A} = \frac{0,5}{2} + \frac{0,3}{5} \neq \emptyset \text{ та } A \cup \hat{A} = \frac{0,5}{2} + \frac{1}{3} + \frac{1}{4} + \frac{0,7}{5} + \frac{1}{6} + \frac{1}{7} \neq 0.$$

4. Декартовий добуток $A \times B$ двох нечітких множин $A \subseteq \mathbf{X}$ і $B \subseteq \mathbf{Y}$ визначається за формулою:

$$\mu_{A \times B}(x, y) = \mu_A(x) \wedge \mu_B(y) = \min(\mu_A(x), \mu_B(y)) \quad (1.6)$$

або

$$\mu_{A \times B}(x, y) = \mu_A(x) \mu_B(y), \quad (1.7)$$

для кожного $x \in \mathbf{X}$ і $y \in \mathbf{Y}$.

Декартовий добуток $A_1 \times A_2 \times \dots \times A_n$ декількох нечітких множин $A_1 \subseteq \mathbf{X}_1$, $A_2 \subseteq \mathbf{X}_2$, ..., $A_n \subseteq \mathbf{X}_n$ визначається як:

$$\begin{aligned} \mu_{A_1 \times A_2 \times \dots \times A_n}(x_1, x_2, \dots, x_n) &= \mu_{A_1}(x_1) \wedge \mu_{A_2}(x_2) \wedge \dots \wedge \mu_{A_n}(x_n) = \\ &= \min(\mu_{A_1}(x_1), \mu_{A_2}(x_2), \dots, \mu_{A_n}(x_n)) \end{aligned}$$

$$\text{або } \mu_{A_1 \times A_2 \times \dots \times A_n}(x_1, x_2, \dots, x_n) = \mu_{A_1}(x_1) \mu_{A_2}(x_2) \dots \mu_{A_n}(x_n),$$

для кожного $x_1 \in X_1, x_2 \in X_2, \dots, x_n \in X_n$.

Приклад.

$$X = \{2, 3, 4\}, Y = \{3, 4\}, A = \frac{0,5}{2} + \frac{1}{3} + \frac{0,7}{4}, B = \frac{0,1}{3} + \frac{0,2}{4}.$$

Декартовий добуток нечітких множин A та B за формулою (1.4) буде:

$$A \times B = \frac{0,1}{(2,3)} + \frac{0,2}{(2,4)} + \frac{0,1}{(3,3)} + \frac{0,2}{(3,4)} + \frac{0,1}{(4,3)} + \frac{0,2}{(4,4)},$$

за формулою (1.5) отримуємо:

$$A \times B = \frac{0,05}{(2,3)} + \frac{0,1}{(2,4)} + \frac{0,1}{(3,3)} + \frac{0,2}{(3,4)} + \frac{0,07}{(4,3)} + \frac{0,14}{(4,4)}.$$

5. Операція концентрації нечіткої множини $A \subseteq X$ $CON(A)$

визначається як:

$$\mu_{CON(A)}(x) = (\mu_A(x))^2 \quad (1.8)$$

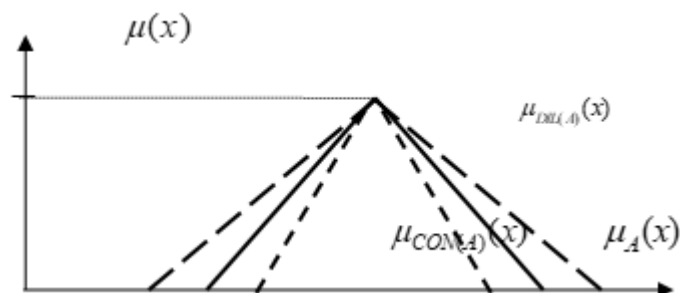
для кожного $x \in X$.

6. Операція розмиття нечіткої множини $A \subseteq X$ $DIL(A)$ визначається як

$$\mu_{DIL(A)}(x) = (\mu_A(x))^{0,5}, \quad (1.9)$$

для кожного $x \in X$.

На графіку рисунку 1.10 показані операції концентрації і розмиття.



1.3 Постановка задач дослідження

Метою кваліфікаційної роботи є розроблення та програмна реалізація алгоритмів визначення відстані між контурами зображень в нечіткій метриці Фреше для оцінки похибок сегментації.

Для досягнення поставленої мети необхідно:

- провести аналіз метрик оцінки відстаней між зображеннями;
- розробити алгоритми пошуку дискретної відстані між контурами у нечіткій метриці Фреше;
- підготувати модуль для пошуку дискретної відстані між контурами у нечіткій метриці Фреше;
- здійснити тестування розробленого модуля.

1.4 Висновки до розділу 1

В розділі 1 проведено аналіз контурів зображень та методи їх опису, введено поняття нечітких множин та проаналізовані операції над ними, здійснено постановку задач дослідження.

2 МЕТРИКИ ФРЕШЕ І ХАУСДОРФА ТА ЇХ МОДИФІКАЦІЇ

2.1 Поняття метричних просторів

Почнемо із поняття метрики.

Метрика на множині X - це функція $d: X \times X \rightarrow \mathbb{R}$, що задовольняю умовам:

1. $d(x, y) \geq 0$, $d(x, y) = 0$,
2. $d(x, y) = d(y, x)$,
3. $d(x, y) \leq d(x, z) + d(z, y)$.

Число $d(x, y)$ називається відстанню між точками x і y .

Якщо умову (1) замінити слабшою умовою (1') $d(x, y) \geq 0$, ми отримаємо поняття псевдометрики. Якщо умову (3) замінити строгішою умовою (3') $d(x, y) \leq \max\{d(x, z), d(z, y)\}$, ми отримуємо поняття ультраметрики. Якщо відкинути умову (2) ми отримуємо поняття квазіметрики.

Різні спеціальні класи метрик і їх властивостей розглянуто у книзі [24].

У метричному просторі (X, d) , для кожного $x \in X$ і $r > 0$, множина $B_r(x) = \{y \in X \mid d(x, y) < r\}$ називається відкритою кулею радіусом r в x . Якщо $A \subset X$, ми говоримо, що $B_r(A) = \cup_{a \in A} B_r(a)$ є r -сусідство A . Множина $U \subset X$ називається обмеженою, якщо вона є в кулі.

Відображення $f: X \rightarrow Y$ з метричного простору в (X, d) метричний простір (Y, ρ) називається ізометрим вкладенням, якщо $\rho(f(x), f(y)) = d(x, y)$ для всіх $x, y \in X$. Відображення метричних просторів називається ізометрією, якщо воно є бієкцією і зберігає відстань. Будь-яке ізометричне вкладення є ізометрією, якщо воно також є відображенням. Ізометрії метричних просторів він утворюють групу щодо операції композиції.

Враховуючи непорожню підмножину в метричному просторі , ми називаємо число (кінечне або) діаметром .

Евклідова відстань між і обчислюється за формулою .

Відображення метричних просторів називається таким, що не розширюється, якщо воно не збільшує відстані між точками.

Враховуючи непорожню множину A у метричному просторі (X, d) , ми називаємо число (скінченне або ∞) $\text{diam} A = \sup\{d(x, y) \mid x, y \in A\}$ діаметр множини A .

Важливими прикладами метричних просторів є евклідові простори \mathbb{R}^n . Евклідова відстань між $x = (x_1, \dots, x_n)$ і $y = (y_1, \dots, y_n)$ обчислюється за

формулою
$$d(x, y) = \left(\sum_{i=1}^n (x_i - y_i)^2 \right)^{1/2}$$
.

Позначимо l^∞ множину всіх обмежених послідовностей дійсних чисел. Відстань на l^∞ задається формулою:

$$d\left(\left(x_i\right)_{i=1}^{\infty}, \left(y_i\right)_{i=1}^{\infty}\right) = \sup_{n \in \mathbb{N}} |x_n - y_n|.$$

Послідовність у метричному просторі називається послідовністю Коші, якщо її діаметр можна зробити довільно малим, видаливши скінченну кількість елементів. Метричний простір називається повним, якщо кожна збіжна послідовність у ньому є збіжною. Повнота простору необхідна для виконання в ньому деяких нескінченних процедур. Найбільш важливі простори в додатках (наприклад, евклідові простори \mathbb{R}^n) є повними.

2.2 Відстань Хаусдорфа

Враховуючи метричний простір, метрика Хаусдорфа дозволяє нам порівнювати підмножину в цьому просторі.

Нехай (X, d) - метричний простір і непусті обмежені підмножини A, C множини X , ми визначаємо відстань Хаусдорфа між ними за формулою:

$$d_H(A, C) = \inf\{r > 0 \mid A \subset B_r(C), C \subset B_r(A)\}.$$

Перш за все, щоб гарантувати існування інфімуму, необхідно розглядати лише обмежені множини. Але навіть тоді, d_H це не є метрикою, оскільки відстань між будь-якою множиною її замиканням дорівнює нулю. Щоб отримати метрику, ми повинні обмежитися закритими обмеженими підмножинами. Маючи на увазі можливі застосування, розглянемо множину

$K(X)$ непустих компактних підмножин of X . Тоді d_H обмежена на $K(X)$ є метрикою (метрикою Хаусдорфа).

Існує еквівалентна формула для метрики Хаусдорфа:

$$d_H(A, C) = \left\{ \sup_{a \in A} \inf_{c \in C} d(a, c), \sup_{c \in C} \inf_{a \in A} d(a, c) \right\}.$$

Зауважимо, що сімейство $\mathcal{F}(X)$ непорожніх скінченних підмножин у X є щільним у топології, породженій метрикою Хаусдорфа. Іншими словами, будь-який компакт можна апроксимувати скінченними множинами. Для будь-якого натурального n будемо визначати $\mathcal{F}_n(X)$ підмножину $\mathcal{F}(X)$, що складається з усіх непорожніх наборів потужності $\leq n$. Множина $\mathcal{F}_n(X)$ замкнена в гіперпросторі X .

Ще один опис метрики Хаусдорфа використовує поняття пари. Враховуючи дві множини A і B , ми говоримо, що підмножина $C \subset A \times B$ є парою між A і B якщо виконується наступне:

1. для кожного $a \in A$, існує $b \in B$ так що $(a, b) \in C$;
2. для кожного $b \in B$, існує $a \in A$ так що $(a, b) \in C$.

Якщо C' є парою між A_1 і A_2 є парою між A_2 і A_3 , то підмножина

$$C = C' \circ C'' = \{(a_1, a_3) \in A_1 \times A_3 \mid (a_1, a_2) \in C', (a_2, a_3) \in C'' \text{ for some } a_2 \in A_2\}$$

є парою між A_1 і A_3 .

Метрика Хаусдорфа d_H , визначається формулою [25]:

$$d_H(A, B) = \min\{\max\{d(a, b) \mid (a, b) \in C\} \mid C \text{ is a coupling between } A \text{ and } B\}.$$

Для кожного натурального n , ми визначаємо $\mathcal{F}_n(X)$ множини $\mathcal{F}(X)$, що складається з множини множин потужністю $\leq n$.

Метрика Хаусдорфа на компактних підмножинах повного метричного простору є повною.

Нагадаємо, що множина в лінійному просторі (зокрема, в евклідовому просторі) є опуклою, якщо для будь-яких двох її точок ця множина містить також лінійний відрізок, який зв'язує ці точки. Неважко довести, що для будь-яких компактних опуклих тіл (тобто опуклих множин з не пустою внутрішньою областю) існує $d_H(A, B) = d_H(\partial A, \partial B)$, де ∂ визначає границю оператора.

Якщо $A, B \subset \mathbb{R}^2$ – полігональні компактні опуклі тіла, то існує алгоритм обчислення відстані Хаусдорфа $d_H(A, B)$ складності $O(m + n)$, де m, n – кількість вершин у відповідно [26].

Відстань Хаусдорфа, а також її модифікації (які не обов'язково є метрикою) мають багато застосувань.

2.3 Відстань Громова-Хаусдорфа

Нехай (X_i, d_i) – два компактних метричних простору. Визначаємо відстань Громова-Хаусдорфа між ними за формулою:

$$d_{GH}(X_1, X_2) = \inf\{d_H(j_1(X_1), j_2(X_2)) \mid j_i: X_i \rightarrow Z, i=1,2 \text{ are embeddings into a metric space } Z\}.$$

Зауважимо, що d_{GH} - це добре визначене, оскільки завжди існує метричний простір Z , який містить ізометричні копії X_1 та X_2 . Дійсно, вибираємо дві базові точки $x_i \in X_i$, і розглянемо букет X_1 і X_2 , $X_1 \vee X_2 = (X_1 \cup X_2) / \{x_1, x_2\}$, де метрика d на $X_1 \vee X_2$ визначається умовою: d збігається з d_i на X_i , $i=1,2$, і якщо $a_i \in X_i \subset X$, $i=1,2$, то $d(a_1, a_2) = d_1(a_1, x_1) + d_2(a_2, x_2)$.

Існує еквівалентне визначення відстані Громова-Хаусдорфа в термінах відповідностей. Відповідність C між будь-якими двома (непорожніми) множинами X і Y є такою підмножиною $C \subset X \times Y$, що:

1. для будь-якого $x \in X$, існує $y \in Y$ так що $(x, y) \in C$,
2. для будь-якого $y \in Y$, існує $x \in X$ так що $(x, y) \in C$.

Позначимо множину $\mathcal{C}(X, Y)$ всіх відповідностей між X і Y .

Тепер, враховуючи метричні простори $(X_i, d_i), i=1,2$, можна визначити відстань Громова-Хаусдорфа між ними за формулою [25]:

$$d_{GH}(X_1, X_2) = \frac{1}{2} \inf_{C \in \mathcal{C}(X_1, X_2)} \sup_{(x_1, y_1), (x_2, y_2) \in C} |d_1(x_1, y_1) - d_2(x_2, y_2)|.$$

Ще один спосіб визначити відстань Громова-Хаусдорфа - це використовувати таку формулу:

$$d_{GH}(X_1, X_2) = \inf \{d_H(j_1(X_1), j_2(X_2)) \mid j_i: X_i \rightarrow \ell^\infty \text{ is an isometric embedding}\}.$$

Відстань Громова-Хаусдорфа приписують М. Громову [27]. Однак А. Тужилін [28] зауважує, що Едвардс [29] використовував це поняття раніше.

Для компактних підпросторів евклідових просторів можна визначити важливу модифікацію відстані Громова-Хаусдорфа ([30]). Позначимо $\text{Iso}(\mathbb{R}^n)$ групу ізометрій евклідового простору \mathbb{R}^n . Визначаємо:

$$d_{H,iso}(A, B) = \inf \{d_H(A, T(B)) \mid T \in \text{Iso}(\mathbb{R}^n)\}.$$

Наступна оцінка для непустих компактних підмножин у n -вимірному евклідовому просторі \mathbb{R}^n доведена в [30]:

$$d_{GH}(A, B) \leq d_{H,iso}(A, B) \leq c'_n (\max\{\text{diam } A, \text{diam } B\} d_{GH}(A, B))^{1/2}$$

(c'_n , що залежить тільки від n)

Наступна оцінка отримана в роботі [31]:

$$d_{H,iso}(A, B) \leq \frac{5}{4} d_{GH}(A, B)$$

для будь-яких компактних підмножин.

Ще один результат з роботи [31] демонструє, що ця оцінка точна: для будь-яких $\delta > 0$ і $\varepsilon \in (0, \frac{1}{4})$ існують компактні підмножини $A, B \subset \mathbb{R}^n$, що

$$d_{GH}(A, B) = \delta \text{ and } d_{H,iso}(A, B) = \left(\frac{5}{4} - \varepsilon \right) \delta.$$

Група ізометрії $Iso(S^n)$ одиничної n -вимірної сфери в евклідовому просторі \check{Y}^{n+1} діє транзитивно на S^n . Виглядає доцільним розглянути ізометричний варіант відстані Громова-Хаусдорфа між непустими компактними підмножинами в S^n .

Є численні публікації, присвячені обчислювальним аспектам відстані Громова-Хаусдорфа. Зокрема, в [32] ця відстань розглядається у випадку метричних дерев. Варіант відстані Громова-Хаусдорфа для ультраметричних просторів досліджено в [33, 34]. Наведено алгоритм поліноміальної складності для оцінки відстані Громова-Хаусдорфа між двома ультраметричними просторами. Також подібні результати отримані для так званих p -метричних просторів.

У [35] визначено відстань Громова-Хаусдорфа для мереж, тобто множини, наділеної функціями $X \times X \rightarrow \check{Y}_+$.

2.4 Відстань Канторовича-Рубінштейна

Нехай (X, d) є метричний простір і $P(X)$ множина всіх регулярних імовірнісних мір на X . Відстань Канторовича (Канторовича-Рубінштейна або Вассерштейна) між $\mu, \nu \in P(X)$ ними обчислюється за формулою:

$$d_K(\mu, \nu) = \left\{ \left| \int_X \varphi d\mu - \int_X \varphi d\nu \right| \mid \varphi: X \rightarrow \mathbb{R} \text{ is nonexpanding} \right\}$$

([36] відстань Канторовича).

Інший спосіб отримати відстань Канторовича між μ і ν полягає в тому, щоб розглянути множину $C(\mu, \nu)$ мір τ на $X \times X$ що задовільняє $\tau(A \times X) = \mu(A)$, $\tau(X \times B) = \nu(B)$ для довільних вимірюваних $A, B \subset X$.

Тоді

$$d_K(\mu, \nu) = \inf_{\tau \in C(\mu, \nu)} \int_{X \times X} d(x, y) d\tau(x, y).$$

У [37] доведено, що відстань Канторовича між двома метриками скінченних опор на (одиничному) колі можна обчислити за лінійний час. Деякі сучасні результати в напрямку складності обчислень можна знайти в [38].

Існують численні публікації, присвячені застосуванню метрики Канторовича до розпізнавання образів і пов'язаним з ними темам [39-45]).

Простір метричної міри (коротко, m -простір) — це трійка (X, d, μ) , де (X, d) — метричний простір і $\mu \in$ мірою ймовірності на X . Зазвичай вважається, що X є компактний.

У роботі [30] розглянуто “гровізацію” відстані Канторовича, а також її застосування до задач комп'ютерного зору. Отриману відстань автор називає відстанню Громова-Вассерштейна. Ця відстань визначена для компактних m просторів.

2.5 Відстань Фреше

Припустимо, що (X, d) це метричний простір. Крива в метричному просторі X є вкладенням $\gamma: [0, 1] \rightarrow X$. Ми отожднюємо γ і $\gamma\alpha$, для кожного гомеоморфізму $\alpha: [0, 1] \rightarrow [0, 1]$. Позначимо множину всіх кривих $C(X)$ у X .

Позначимо множину кривих $C(X)$ у X , тобто вкладення $[0,1]$ в X .
Відстань Фреше між кривими $\gamma_1, \gamma_2 : [0,1] \rightarrow X$ дорівнює

$$d_F(\gamma_1, \gamma_2) = \inf \{ \sup \{ d(\gamma_1(\alpha(t)), \gamma_2(t)) \mid t \in [0,1] \} \mid \alpha : [0,1] \rightarrow [0,1] \text{ is an increasing homeomorphism} \}.$$

Якщо ми не вимагаємо, щоб гомеоморфізм α у визначенні зберігав орієнтацію, то отримаємо неорієнтовану відстань Фреше.

Дискретна відстань Фреше визначається в, і ця відстань знайшла багато застосувань.

Насправді, дискретна відстань Фреше тісно пов'язана з відстанню Фреше між полігональними кривими в евклідових просторах.

Точніше, полігональна крива - це таке відображення $f : [0,1] \rightarrow \check{Y}^k$, що існує розбиття відрізка $[0,1]$ точками $0 = t_0 < t_1 < \dots < t_{n-1} < t_n = 1$, що є лінійним на кожній $[t_i, t_{i+1}]$, $i = 0, 1, \dots, n-1$, тобто

$$f(t) = \frac{t_{i+1} - t}{t_{i+1} - t_i} f(t_i) + \frac{t - t_i}{t_{i+1} - t_i} f(t_{i+1}), t \in [t_i, t_{i+1}].$$

Допустимо $g : [0,1] \rightarrow \check{Y}^k$ ще одна полігональна крива з відповідним розбиттям $0 = \tau_0 < \tau_1 < \dots < \tau_{m-1} < \tau_m = 1$. Пара f і g є послідовністю пар невід'ємних цілих чисел

$$L = ((0,0) = (a_1, b_1), \dots, (a_{q-1}, b_{q-1}), (a_q, b_q))$$

так що $a_{i+1} = a_i$ or $a_{i+1} = a_i + 1$, $b_{i+1} = b_i$ or $b_{i+1} = b_i + 1$, and $a_q \geq n$, $b_q \geq m$. Довжина L є число

$$\|L\| = \max_{i=1, \dots, g} d(f(t_{a_i}), g(\tau_{b_i})).$$

Дискретна відстань Фреше між полігональними кривими f і g дорівнює тоді

$$d_{dF}(f, g) = \min\{\|L\| \mid L \text{ is a coupling of } f \text{ and } g\}.$$

Це доведено в тому, що d_{dF} можна обчислити за $O(nm)$ час.

У роботі [46] показано, що існує алгоритм, який обчислює цю відстань за час $O(mn \log^2(mn))$, де m, n – кількість лінійних відрізків на кривих. Щоб бути більш точним, ми визначаємо дискретну відстань Фреше між полігональними кривими так.

Існує також немонотонна відстань Фреше [46, 47].

Зауважимо, що збіжність відносно відстані Хаусдорфа послідовності метричних кривих не означає її збіжність відносно відстані Фреше.

2.6 Відстань Громова-Фреше між кривими

Наступна версія метрики Фреше введена в [48].

Нехай γ_1, γ_2 метричні криві. Визначимо відстань Громова-Фреше $d_{GF}(\gamma_1, \gamma_2)$ так:

$$d_{GF}(\gamma_1, \gamma_2) = \inf\{d_F(i(S), j(T))$$

$\mid i: S \rightarrow Z, j: T \rightarrow Z \text{ are embeddings into a metric space } Z\}.$

У [48] доведено, що відстань Громова-Фреше між метричними кривими обмежена знизу відстанню Громова-Хаусдорфа між їх опорами.

Позначимо через \mathcal{C} метричний простір класів ізометрії метричних кривих, наділених відстанню Громова-Фреше. У [48] доведено, що \mathcal{C} простір сепарабельний. Останнє означає, що власне, цією щільною підмножиною у \mathcal{C} може служити зліченна щільна підмножина в просторах імовірнісних мір.

Також у [48] побудовано приклад, який демонструє, що \mathcal{C} не є повним метричним простором. Розглянемо послідовність $(\gamma_i)_{i=1}^{\infty}$ кривих для виконання умов:

1. послідовність $(\text{supp}(\gamma_i))_{i=1}^{\infty}$ сходиться до одиничного квадрата $[0,1] \times [0,1]$ в метриці Хаусдорфа;

2. $d_F(\gamma_i, \gamma_{i+1}) \leq 2^{-i}$ для всіх i

(можна взяти як $(\gamma_i)_{i=1}^{\infty}$ послідовність наближень до кривої Гільберта, що заповнює квадрат [49]). Тоді $(\gamma_i)_{i=1}^{\infty}$ є послідовність Коші у просторі \mathcal{C} , і припустимо, що γ це її границя. Зі сказаного вище випливає, що послідовність $(\text{supp}(\gamma_i))_{i=1}^{\infty}$ сходиться до $\text{supp}(\gamma)$ в метриці d_{GH} . Оскільки $d_{GH} \leq d_H$, ми бачимо, що $(\text{supp}(\gamma_i))_{i=1}^{\infty}$ також сходиться до $[0,1] \times [0,1]$ в метриці d_{GH} , і це створює протиріччя.

Можна визначити деякі модифікації відстані Громова-Фреше.

1) Можна також розглянути немонотонний варіант відстані Фреше [46]. Підставляючи цю відстань d_{GF} у формулу на один, отримуємо немонотонну відстань Громова-Фреше.

2) Існує також версія відстані Фреше для замкнутих кривих (тобто гомеоморфних зображень кола S^1). Аналогічно можна визначити відстань Громова-Фреше між метричними замкнутими кривими.

У цих двох випадках можна також розглядати відповідні ізометричні відстані Громова-Фреше.

Зауважимо, що поняття відстані Фреше для параметризованих поверхонь розглядається в [50].

У [48] розглянуто інший підхід. Нехай $\alpha : [0,1]^2 \rightarrow X$ — параметризована поверхня в метричному просторі X . Для будь-якого $s \in [0,1]$, нехай перетворення $\alpha_s : [0,1] \rightarrow X$ визначається так: $\alpha_s(t) = \alpha(t,s)$ $t \in [0,1]$.

Позначимо $\bar{\alpha}$ перетворення s а $\alpha_s : I \rightarrow \text{Curve}(X)$, де $\text{Curve}(X)$ означає метричний простір кривих у X , наділений метрикою Фреше. Тепер відстань Фреше між параметризованими поверхнями α, β визначається за формулою: $D_F(\alpha, \beta) = d_F(\bar{\alpha}, \bar{\beta})$. На основі цієї відстані можна визначити відповідну відстань Громова-Фреше між параметризованими поверхнями в метричних просторах.

Дискретна відстань Фреше визначена в [25, 32, 51]. Щоб визначити дискретну відстань Громова-Фреше, ми припустимо, що розглянуті криві є полігональними в тому сенсі, що вони ізометричні полігональними кривими у нормованих просторах (наприклад, у l^∞).

Аналогічно описаному вище варіанту відстані Громова-Хаусдорфа можна розглядати таке поняття. Припустимо, що γ_1, γ_2 є кривими в \check{Y}^n . Визначимо

$$d_{F,\text{iso}}(\gamma_1, \gamma_2) = \inf \{d_F(\gamma_1, T(\gamma_2)) \mid T \in \text{Iso}(\mathbb{R}^n)\}.$$

Очевидно $d_{GF} \geq d_{F,\text{iso}}$. Ми збираємося продемонструвати в загальному, що $d_{GF} < d_{F,\text{iso}}$. Нехай γ_1 є рівностороннім трикутником зі сторонами одиничної довжини у \check{Y}^2 з невеликим відрізком, видаленим (щоб мати просту криву).

Нехай також γ_2 проста крива \check{Y}^2 досить малого діаметра. Тоді $d_{F,\text{iso}}(\gamma_1, T(\gamma_2))$ приблизно дорівнює $\sqrt{3}/2$. Розглянемо метрику ρ на диз'юнктному об'єднанні γ_1 і $*$ визначається умовою: ρ збігається з вихідною метрикою на γ_1 і $\rho(x, *) = 1/2$ для кожної точки x кривої

Додамо копію γ_2 до $*$. Один із способів визначення метрики на отриманому об'єднанні γ_1 і γ_2 полягає в наступному. Через універсальність простору l^∞ , можна вважати, що $\gamma_1 \cup \{*\}, \gamma_2 \subset l^\infty$ (як метричні підпростори). Тоді розглянемо множину $\gamma_1 \cup (\gamma_2 - a + *)$, де a – довільна точка $\text{supp}(\gamma_2)$. Ця множина містить метричні копії γ_1 і γ_2 , отже $d_{GF}(\gamma_1, \gamma_2)$, близький до $1/2$.

Ми сформулюємо як відкриту задачу точних оцінок для d_{GF} і $d_{F,iso}$ подібних до тих, що отримані для d_{GH} і $d_{H,iso}$. Оцінки явно збігаються у випадку \check{Y}^1 .

2.7 Метрика Громова-Фреше для ізометричних класів дерев

Дерева. Абстрактне дерево – це зв'язний граф без циклів. Кореневе дерево — це пара (T, t_0) , де T є дерево і $t_0 \in T$. Зазвичай ми працюємо з деревами в метричних просторах, тобто із зображеннями вкладень графів у метричні простори.

Зважене кореневе дерево є триплет (T, t_0, w) , де (T, t_0) кореневе дерево і претворення $w: T \rightarrow \check{Y}_+$ є функцією (вага).

Позначимо $e(T)$ множину всіх кінцевих точок T . Для будь-якого $a, b \in e(T)$, позначимо $L(a, b) = L^T(a, b)$ (унікальний) геодезичне з'єднання a та b . Іншими словами, це просто відрізок з кінцевими точками a та b . Власне, множина $M^T = \{L(a, b)^T \mid a, b \in e(T), a \neq b\}$ є множиною всіх максимальних геодезичних у T . Очевидно, $\#M^T = \frac{1}{2}(\#e(T) \cdot (\#e(T) - 1))$, де $\#A$ означає потужність будь-якої множини A .

Розглянемо дерева, вкладені в площину \check{Y}^2 . Очевидно, що наші міркування можна легко поширити на будь-який евклідовий простір.

2.7 Скелетони

Для простоти розглянемо тут лише підмножини в евклідовій площині \check{Y}^2 . Позначимо через d евклідову метрику. Враховуючи непорожню підмножину $A \subset \check{Y}^2$ і $x \in \check{Y}^2$, ми отримаємо $d(x, A) = \inf\{d(x, a) \mid a \in A\}$.

Допустим D є обмежена область в \check{Y}^2 . Позначимо його межу через ∂D . Для будь-якого $x \in D$, позначимо $p(x)$ його метричною проекцією

$$p(x) = \{y \in \partial D \mid d(x, y) = d(x, \partial D)\}.$$

Скелетон (центральна множина) D є множина

$$\text{sk}D = \{x \in D \mid \#p(x) \geq 2\}.$$

У роботі розглянуті [52] метричні та топологічні властивості скелетонів відкритих множин у евклідових просторах.

Є домени, скелети яких демонструють нерегулярну поведінку [53]. Однак це не стосується доменів, які з'являються в застосуваннях. Зокрема, відомо, що скелетом будь-якої області з досить гладкою межею є деформаційний ретракт.

Більше того, скелетами багатогранних простих зв'язаних доменів є дерева.

Функція $w: \text{sk}D \rightarrow \check{Y}_+$, визначена формулою $w(x) = d(x, \partial D)$, є вагою на $\text{sk}D$.

Метрика. Дано криву L в метричному просторі X та функцію $w: L \rightarrow \check{Y}_+$, нехай $\hat{L} = \{(x, w(x)) \mid x \in L\}$. Тоді \hat{L} є крива в просторі $X \times \check{Y}$. Наділяємо $X \times \check{Y}$ макс-метрикою.

Для будь-яких двох зважених кореневих дерев (T, t_0, w) і (S, s_0, u) , ми отримаємо

$$D_F((T, t_0, w), (S, s_0, u)) = \min\{\max\{d_F(\vec{L}_a^T, \vec{L}_b^S) \mid (a, b) \in C\} \mid C \text{ is a coupling between } e(T) \text{ and } e(S)\}.$$

Доведення наступного результату наведено в [54].

Теорема Функція D_F є метрикою на множині зважених кореневих дерев, вкладених у метричний простір X .

Доведення.

Зрозуміло, якщо $D_F((T, t_0, w), (S, s_0, u)) = 0$, то $(T, t_0, w) = (S, s_0, u)$.

Також очевидно, що функція є симетричною.

Ми збираємося довести нерівність трикутника для D_F . Нехай

$$(T, t_0, w), (S, s_0, u), (R, r_0, v)$$

є зважені кореневі дерева. Існує зв'язок C між $e(T)$ і $e(S)$ (відповідно C' між $e(S)$ і $e(R)$) така, що

$$D_F((T, t_0, w), (S, s_0, u)) = \max\{d_F(\vec{L}_a^T, \vec{L}_b^S) \mid (a, b) \in C\}$$

(відповідно

$$D_F((S, s_0, u), (R, r_0, v)) = \max\{d_F(\vec{L}_b^S, \vec{L}_c^R) \mid (b, c) \in C'\}.$$

Визначимо $C'' = C \circ C'$. Тоді

$$D_F((T, t_0, w), (R, r_0, v)) \leq \max\{d_F(\vec{L}_a^T, \vec{L}_c^R) \mid (a, c) \in C''\}.$$

Тоді існує $(a', c') \in C''$ так що

$$D_F((T, t_0, w), (R, r_0, v)) = d_F(\tilde{L}_a^T, \tilde{L}_{c'}^R).$$

Тоді існує $b \in e(S)$ так що $(a', b) \in C$, $(b, c') \in C'$. Робимо висновок, що

$$\begin{aligned} D_F((T, t_0, w), (R, r_0, v)) &= d_F(\tilde{L}_a^T, \tilde{L}_{c'}^R) \leq d_F(\tilde{L}_a^T, \tilde{L}_b^S) + d_F(\tilde{L}_b^S, \tilde{L}_{c'}^R) \\ &\leq \max\{d_F(\tilde{L}_a^T, \tilde{L}_{b'}^S) \mid (a, b') \in C\} \\ &\quad + \max\{d_F(\tilde{L}_{b'}^S, \tilde{L}_{c'}^R) \mid (b', c) \in C'\} \\ &\leq D_F((T, t_0, w), (S, s_0, u)). \end{aligned}$$

Щоб отримати відстань Фреше між деревами без ваги, ми просто припустимо, що $w(x) = 1$ для всіх x розглянутих дерев. Зауважимо, що отримана відстань узгоджується з відстанню Фреше на множині орієнтованих кривих. Дійсно, кожен орієнтований криву можна розглядати як кореневе дерево з початковою точкою коренем і навпаки.

2.8 Відстань Громова-Фреше

Нехай S, T - це метричні дерева. Визначимо відстань Громова-Фреше $d_{GF}(S, T)$ так:

$$\begin{aligned} d_{GF}(S, T) &= \inf\{d_F(i(S), j(T)) \\ &\quad \mid i: S \rightarrow Z, j: T \rightarrow Z \text{ are embeddings into ametric space } Z\}. \end{aligned}$$

Позначимо через \mathfrak{T} множину (ізометричних класів) метричних дерев.

Теорема Функція d_{GF} є метрика на множині \mathcal{T} .

Метрика на множині дерев. Обговорення.

Далі ми не будемо вважати, що дерева, що розглядаються, є кореневими. У роботі [54] автори спробували модифікувати відстань Фреше між кореневими деревами так, щоб отримана функція була метрикою на множині (некорневих) дерев. Насправді, в дерева T , можна вважати довільну його точку коренем. Це дозволяє змінити відстань Фреше таким чином:

$$\tilde{D}_F((T, w), (S, u)) = \inf \{D_F((T, t_0, w), (S, s_0, u)) \mid t_0 \in T, s_0 \in S\}.$$

Однак наступний приклад демонструє, що функція \tilde{D}_F не задовольняє нерівності трикутника, а отже, не є метрикою (ми припускаємо, що ваги на розглянутих деревах однакові і рівні 1, і для простоти ми відкидаємо позначення для ваги).

Нехай

$$T = \{(x, y) \in \mathbb{R}^2 \mid y = r|x| - 1, |y| \leq 1\},$$

$$S = \{(0, y) \mid |y| \leq 1\},$$

$$R = \{(x, -y) \mid (x, y) \in T\},$$

(тут ми припускаємо, що значення $r > 0$ досить мале). Прості геометричні аргументи показують що $\tilde{D}_F(T, S) \geq 1$ і якщо $r < 1/2$ ми отримаємо

$$\tilde{D}_F(T, R) \geq 1 > 2r \geq \tilde{D}_F(S, T) + \tilde{D}_F(S, R),$$

що суперечить нерівності трикутника.

Відстань Фреше. Дані дерева S, T в метричному просторі X ми визначаємо як

$$d_F(S, T) = \min \{ \max \{ d_F(\gamma, \gamma') \mid (\gamma, \gamma') \in C \} \mid C \text{ is a coupling between } M^S \text{ and } M^T \}.$$

Теорема Функція d_F є метрикою на множині $\mathcal{T}(X)$.

Доведення. 1) Допустим $d_F(S, T) = 0$, то існує пара C між M^S і M^T так що $d_F(\gamma, \gamma') = 0$ для всіх $(\gamma, \gamma') \in C$. Більше того, $\gamma = \gamma'$ for all $(\gamma, \gamma') \in C$. Тоді

$$S = \bigcup \{ \gamma \mid (\gamma, \gamma') \in C \text{ for some } \gamma' \in M^T \}$$

і

$$T = \bigcup \{ \gamma' \mid (\gamma, \gamma') \in C \text{ for some } \gamma \in M^S \}.$$

ми робимо висновок, що $S = T$.

2) Очевидно, що функція d_F є симетрична.

3) Доведемо нерівність трикутника для d_F . Допустим R, S, T є дерева в метричному просторі X . Припустимо, що R, S, T є дерева. Допустим, що C_1 є пара між M^R і M^S такі, що

$$d_F(R, S) = \max \{ d_F(\gamma, \gamma') \mid (\gamma, \gamma') \in C_1 \}.$$

Допустимо C_2 є пара між M^S і M^T так, що

$$d_F(S, T) = \max \{ d_F(\gamma, \gamma') \mid (\gamma, \gamma') \in C_2 \}.$$

Нехай

$$C = \{(\gamma, \gamma') \in M^R \times M^T \mid$$

there exists $\gamma'' \in M^S$ such that $(\gamma, \gamma'') \in C_1, (\gamma'', \gamma') \in C_2\}$,

тоді C є пара між M^R і M^T . Враховуючи, що $\gamma \in M^R$, можна знайти таке $\gamma' \in M^T$, що $(\gamma, \gamma') \in C$. Тоді існує таке $\gamma'' \in M^S$, що $(\gamma, \gamma'') \in C_1$ і $(\gamma'', \gamma') \in C_2$. Ми маємо

$$d_F(\gamma, \gamma') \leq d_F(\gamma, \gamma'') + d_F(\gamma'', \gamma') \leq d_F(R, S) + d_F(S, T).$$

Аналогічно можна показати, що для кожного $\gamma \in M^T$, можна знайти таке $\gamma' \in M^R$, що $d_F(\gamma, \gamma') \leq d_F(R, S) + d_F(S, T)$. Підводячи підсумок, робимо висновок $d_F(R, T) \leq d_F(R, S) + d_F(S, T)$.

Зауваження 2 Іншими словами, відстань Фреше між двома метричними деревами — це відстань Хаусдорфа між множинами максимальних геодезичних відносно метрики Фреше в просторі метричних кривих. Насправді присвоєння T а M^T є вкладенням $T(X)$ у $\mathcal{F}(C(X))$.

Примітка 3. Введена відстань, очевидно, є розширенням відстані Фреше між (неорієнтованими) кривими.

Для будь-якого $n \in \Gamma$ позначимо $\mathcal{T}_n(X)$ множину всіх дерев T у X таких, що $\#e(T) \leq n$. Очевидно, $\mathcal{C}(X) = \mathcal{T}_2(X)$.

Твердження. Для кожного натурального n простір $\mathcal{T}_n(X) \setminus \mathcal{T}_{n-1}(X)$ є замкнутим у просторі $\mathcal{T}(X)$.

Доведення. Припустимо навпаки і нехай $(T_i)_{i=1}^\infty$ є послідовність у просторі $\mathcal{T}_n(X) \setminus \mathcal{T}_{n-1}(X)$, що збігається до $T \in \mathcal{T}_{n-1}(X)$. За визначенням існують $a_i, b_i \in e(T_i)$, $a_i \neq b_i$, $i \in \Gamma$, і $a \in e(T)$ такі, що $\lim_{i \rightarrow \infty} a_i = \lim_{i \rightarrow \infty} b_i = a$. Існує геодезична крива γ_i , що з'єднує a_i і b_i , $i \in \Gamma$. Тоді послідовність $(\gamma_i)_{i=1}^\infty$ сходиться до максимальної геодезичної γ в просторі T . Однак, дві

кінцеві точки γ повинні збігатися з a і ми отримуємо протиріччя з максимальністю.

Наслідок Множина $\mathcal{C}(X) = \mathcal{T}_2(X)$ є замкненою підмножиною $\mathcal{T}(X)$.

Пропозиція Простір $\mathcal{T}(X)$ не є повний.

Доведення. Результат є наслідком пропозиції 9.5 і неповноти простору кривих у метриці Фреше.

Наступний приклад демонструє, що множина всіх $T' \in \mathcal{T}(X)$ гомеоморфних $T \in \mathcal{T}(X)$ не обов'язково замкнута в $\mathcal{T}(X)$.

Приклад. Нехай $i \in \Gamma$. Розглянемо дерево T_i у \check{Y}^2 , що складається з таких сегментів:

$$[(1,1), (1/i, 0)], [(1, -1), (1/i, 0)], [(-1, 1), (-1/i, 0)], \\ [(-1/i, 0), (1/i, 0)].$$

Очевидно, що всі T_i є взаємно гомеоморфні. Послідовність $(T_i)_{i=1}^\infty$ сходиться до дерева, що складається з відрізків, що з'єднують точки $(1, 1)$, $(1, -1)$, $(-1, 1)$, $(-1, -1)$ з початком координат. Отримане дерево не є гомеоморфним жодному з T_i .

Твердження 8. Простір $\mathcal{T}(X)$ є сепарабельним.

Доведення. Оскільки простір $\mathcal{C}(X)$ сепарабельний [48] і $\mathcal{F}(Y)$ є сепарабельним для кожного сепарабельного Y , отримуємо результат.

Нагадаємо, що тріод — це топологічний простір, гомеоморфний конусу, над дискретним простором, що складається з 3 точок. Ми розглядаємо тріоди як кореневі дерева, припускаючи, що вершина конуса є коренем. Позначимо $\mathcal{T}^0(X)$ множину всіх тріодів у метричному просторі X . Нагадаємо, що в [54] визначено варіант відстані Фреше на множині корневих дерев. Нижче ми будемо позначати це через d_F^0 . Наступна формула для d_F^0 еквівалентна формулі в [54]:

$$d_F^0((T, t_0), (S, s_0)) = d_H(\{[t_0, t] \mid t \in e(T)\}, \{[s_0, s] \mid s \in e(S)\})$$

(тут d_H означає метрику Хаусдорфа в множині (орієнтованих) кривих у X , породжених метрикою Фреше).

Наступний результат наведено без доведення.

Теорема. Топології на $T^0(X)$, індуковані метрикою d_F^0 і d_F (в останньому випадку ми розглядаємо дерева як некореневі), збігаються.

Метрика Громова-Фреше на ізометричних класах дерев. До метрики Фреше можна застосувати процедуру «громовізації».

Нехай S, T це метричні дерева. Визначимо відстань Громова-Фреше $d_{GF}(S, T)$ так:

$$d_{GF}(S, T) = \inf \{d_F(i(S), j(T)) \mid i: S \rightarrow Z, j: T \rightarrow Z \text{ are embeddings into a metric space } Z\}.$$

Теорема. Функція d_{GF} є метрикою на множині \mathfrak{T} .

Позначимо через \mathfrak{T} множиною (ізометричних класів) метричних дерев.

Теорема. Простір \mathfrak{T} є сепарабельним.

Доведення. Спочатку опишемо скінченний набір метричних дерев. Позначимо через \mathfrak{M} множини всіх скінченних метричних просторів з раціональними відстанями. Зрозуміло, що множина \mathfrak{M} є скінченна. Для будь-якого $X \in \mathfrak{M}$ розглянемо X включення Куратовського в нормований простір $K(X)$. Позначимо через \mathcal{T}_X множини метричних дерев в просторі $K(X)$, утворених лінійними відрізками з вершинами в X . Оскільки кожне \mathcal{T}_X є скінченне, ми робимо висновок, що $\mathfrak{T}' = \cup \{\mathcal{T}_X \mid X \in \mathfrak{M}\}$ є скінченним.

2.6 Висновки до розділу 2

1. Проаналізовано в загальному метричні простори що дало можливість використати метрики Фреше та Хаусдорфа для визначення подібності зображень.

2. Проаналізовано метрику Фреше та метрику Громова-Фреше для порівняння контурів зображень.

3. Проаналізовано метрику Хаусдорфа, метрику Громова-Хаусдорфа для порівняння областей зображень.

3 ПРОГРАМНИЙ МОДУЛЬ ОБЧИСЛЕННЯ ВІДСТАНИ МІЖ ЗОБРАЖЕННЯМИ

3.1 Системні вимоги до програмного модуля

Даний програмний модуль оцінює якість сегментації зображень різними алгоритмами у порівнянні з експертною сегментацією [55].

На вхід модуля подаються шляхи до зображень (експертна сегментація та сегментація алгоритмами). На виході отримується середньозважена оцінка подібності зображень.

Сформулюємо основні вимоги до системної підтримки програмного модуля:

- 1.1. Робота із зображеннями в JPEG, PNG та GIF форматах;
- 1.2. Система зобов'язана приймати на вхід два зображення та проводити їхню паралельну обробку;
- 1.3. Реалізація порівняння контурів зображень в метриці Фреше;
- 1.4. Результат роботи програми має відображатися у консоль;
- 1.5. Введення шляхів до зображень вводяться через консоль.
2. Вимоги до надійності:
 - 2.1. Програма повинна працювати із великими за розміром зображеннями, без збоїв;
 - 2.2. Запобігання помилкам при переповненні пам'яті;
 - 2.3. Передбачити виведення помилок при їхньому виникненні.
3. Вимоги до робочої станції:
 - 3.1. Система повинна працювати на IBM-сумісних робочих станціях;
 - 3.2. Мінімальні вимоги до робочих станцій: процесор із частотою від 1.8 ГГц із кількістю ядер більше 1, клавіатура, монітор;
 - 3.3. Оперативна пам'ять від 1 Гб;
 - 3.4. Наявність дискретної відео карти nVidia з підтримкою технології CUDA.

4. Вимоги до програмного забезпечення:

4.1. Операційна система – система Unix, Windows;

4.2. Наявність Java;

4.3. Розмір програми до 10 Мб;

4.4. Код програмних модулів повинен містити необхідні для його розуміння коментарі.

3.2 Архітектура програмного модуля

Архітектуру програмного модуля представимо за допомогою діаграми класів, яка наведена на рисунку 3.1.

Діаграма класів пакету Utils наведено на рисунку 3.2.

Деталізуємо вміст пакетів та класів.

Клас Main:

public class Main – точка входу в модуль.

Експертне зображення - private static final String img1

Після порогової сегментації - private static final String img2

Сегментація методом водорозподілу - private static final String img3

```
Mat img*** = ImageOperations.prepareImage (img***,  
"THRESH_TRIANGLE"); - виклик функції для перетворення зображення в  
формат opencv Mat
```

```
List<MatOfPoint> contours*** = ImageOperations.prepareContours (img***);
```

- виклик функції для виділення контурів на зображенні

```
MainComparator mainComparator = new MainComparator();
```

```
// add new comparator
```

```
mainComparator.add(new HausdorffComparator()); - додавання нового  
компаратора
```

mainComparator.compare(contour 1, contour 2) – передача контурів зображення для порівняння.

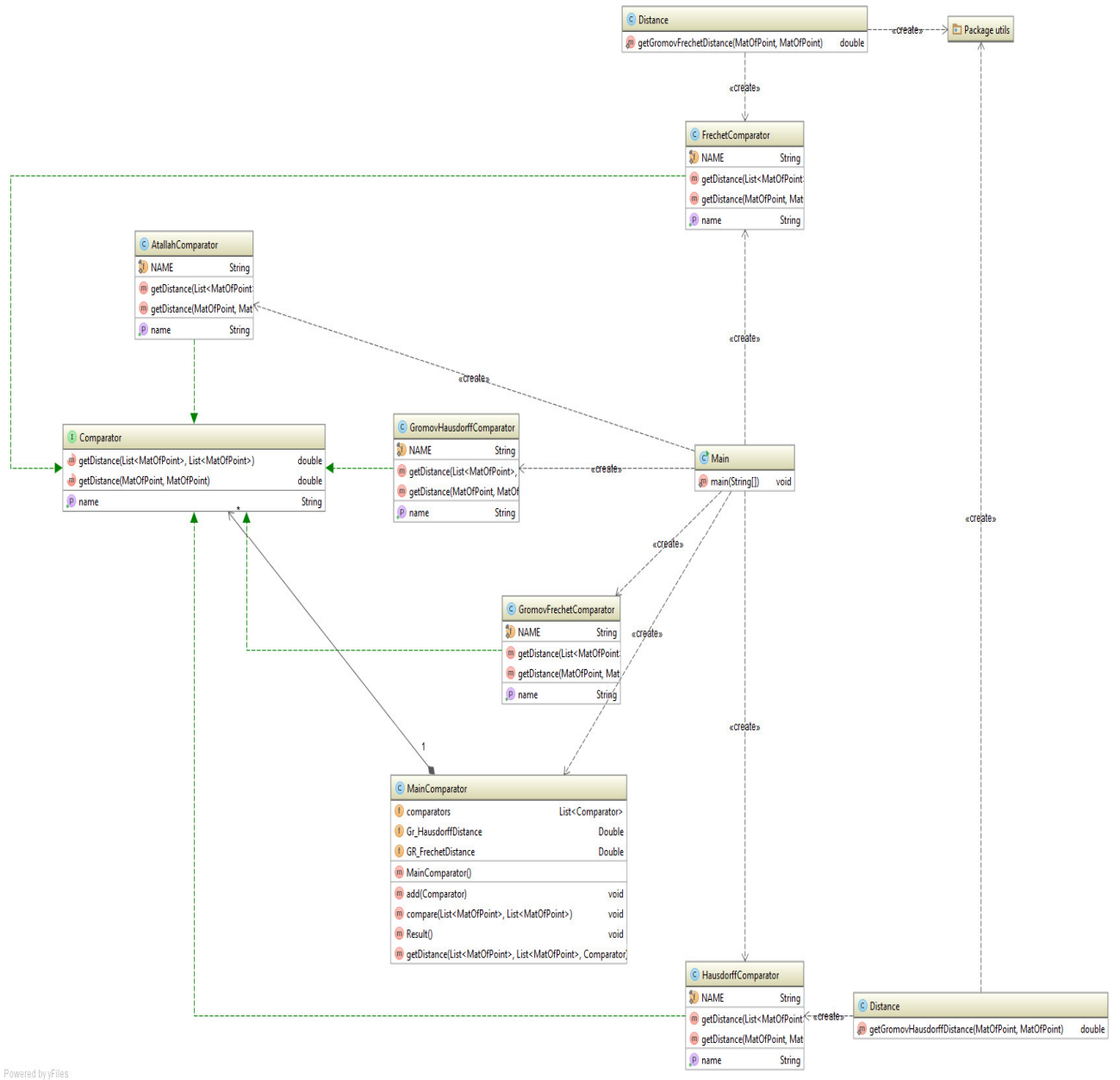


Рисунок 3.1 – Діаграма класів модуля оцінки сегментації

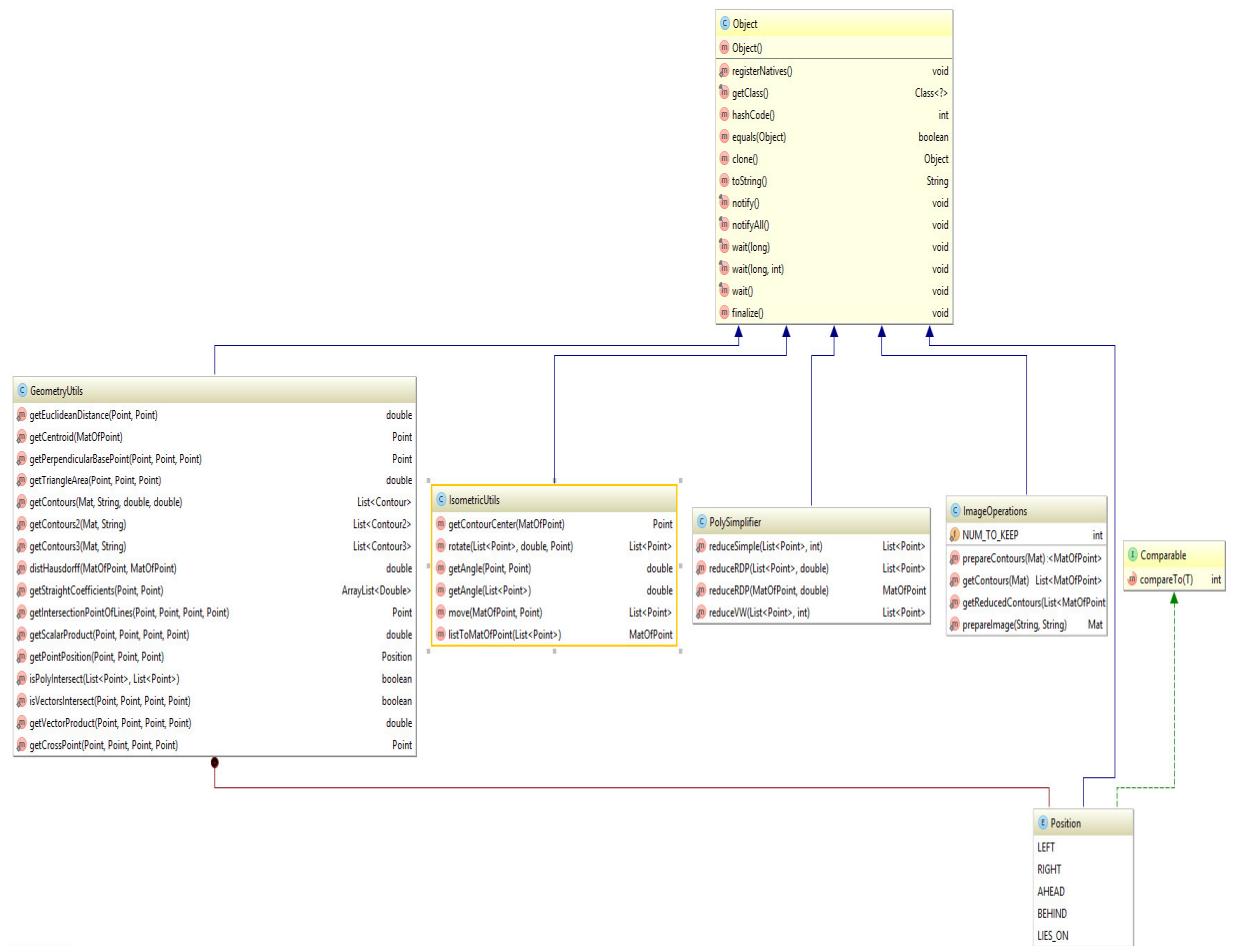


Рисунок 3.2 – Діаграма класів пакету Utils

Кожний компаратор повинен імплементувати інтерфейс Comparator та перевизначити його методи.

Реалізація кожного компаратора знаходиться в окремому пакеті з відповідною назвою.

Допоміжні класи та методи знаходяться в пакеті Utils.

Опис основного інтерфейсу приведений в таблиці 3.1.

Опис основного компаратора приведений в таблиці 3.2.

Package Comparator.

Таблиця 3.1 – Опис основного інтерфейсу

Інтерфейс	<pre>public interface Comparator</pre> <p>інтерфейс, котрий повинні унаслідувати усі компаратори</p>
Методи	<pre>double getDistance(List<MatOfPoint> image1, List<MatOfPoint> image2);</pre> <p>повертає результат (відстань). Тип вхідних параметрів: List<MatOfPoint></p> <pre>double getDistance(MatOfPoint contour1, MatOfPoint contour2);</pre> <p>повертає результат (відстань). Тип вхідних параметрів: MatOfPoint</p> <pre>String getName();</pre> <p>Повертає назву компаратора</p>

Таблиця 3.2 – Опис основного компаратора

Клас	MainComparator
Методи	<pre>public void add(Comparator comparator)</pre> <p>додавання нового методу</p> <pre>public void compare(List<MatOfPoint> image1, List<MatOfPoint> image2)</pre> <p>порівнює зображення всіма доступними компараторами image1 – список контурів експертного зображення image2 – список контурів сегментованого зображення</p> <pre>private double getDistance(List<MatOfPoint> contours1, List<MatOfPoint> contours2, Comparator comparator)</pre>

Продовження таблиці 3.2

	<p>Алгоритм порівняння двох зображень (може бути змінений на будь-який інший)</p> <ul style="list-style-type: none">* Кожен контур першого зображення порівнюється з кожним контуром другого зображення* 1. Шукаємо найменшу відстань від контура зображення 1 до контурів зображення 2. Зберігаємо це значення* 2. Вибираємо найбільше значення зі всіх збережених на кроці 1.
--	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Опис класу визначення відстані на основі метрики Хаусдорфа приведений в таблиці 3.3.

Package Hausdorff

Клас HausdorffComparator

Перевизначає усі методи інтерфейсу Comparator

package comparator.hausdorff.core;

Опис класу визначення відстані на основі алгоритму Аталаха приведений в таблиці 3.4.

Package comparator.atallah;

Клас AtallahComparator перевизначає методи інтерфейсу Comparator.

Крім цього повертає відстань в метриці Хаусдорфа, базується на алгоритмі Аталаха, працює тільки для опуклих фігур

Таблиця 3.3 – Опис класу визначення відстані на основі метрики Хаусдорфа

Клас	Distance Відстань на основі метрики Хаусдорфа * використовується повний перебір точок
Методи	<pre> public static double getHausdorffDistance(MatOfPoint contour1, MatOfPoint contour2) повертає відстань MatOfPoint contour1 – контур ядра експертного зображення MatOfPoint contour2 – контур ядра сегментованого зображення private static double getDeviation(Point p, List<Point> contour) відхилення точки від контура (для Хаусдорфа), повертає відстань найменшу відстань до множини private static double getDeviation(List<Point> contour1, List<Point> contour2) відхилення контура від контура (для Хаусдорфа), повертає відстань // порядок аргументів має значення getDeviation(c1, c2) != getDeviation(c2, c1) </pre>

```
package comparator.atallah.core;
```

Таблиця 3.4 – Опис класу визначення відстані на основі алгоритму Аталаха

Клас	Distance Алгоритм Аталаха
Методи	<pre> public static double getHausdorffDistanceModDisjoint(MatOfPoint contour1, MatOfPoint contour2) Модифікований алгоритм пошуку відстані Хаусдорфа </pre>

Продовження таблиці 3.4

	<p>Обов'язково для розділених багатокутників (таких що не перетинаються і один не міститься в іншому)</p> <pre>public static double getHausdorffDistanceModDisjoint(MatOfPoint contour1, MatOfPoint contour2)</pre> <p>Модифікований алгоритм пошуку відстані Хаусдорфа</p> <p>Обов'язково для розділених багатокутників (таких що не перетинаються і один не міститься в іншому)</p> <pre>private static double getAttalahDisjointDeviation(List<Point> contour1, List<Point> contour2)</pre> <p>Повертає відхилення однієї множини від іншої</p> <p>// використовується алгоритм Аталаха для РОЗДІЛЕНИХ багатокутників</p> <pre>private static int getNextVertexIndex(int curPosition, int polySize, boolean clockwise)</pre> <p>повертає наступну точку контура</p>
--	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Опис класу визначення відстані на основі метрики Фреше приведений в таблиці 3.5.

Package

comparator.frechet

Клас FrechetComparator перевизначає методи інтерфейсу Comparator

package comparator.frechet.core;

Опис класу визначення відстані на основі метрики Громова – Фреше приведений в таблиці 3.6.

Таблиця 3.5 – Опис класу визначення відстані на основі метрики Фреше

Клас	Distance
Методи	<pre> public static double getFrechetDistance(MatOfPoint contour1, MatOfPoint contour2) повертає відстань Фреше private double frechet(List<Point> contourA, List<Point> contourB) реалізація самого алгоритму Фреше </pre>

Package comparator.gromovFrechet;

Клас GromovFrechetComparator перевизначає методи інтерфейсу Comparator

package comparator.gromovFrechet.core;

Таблиця 3.6 – Опис класу визначення відстані на основі метрики Громова - Фреше

Клас	Distance
Методи	<pre> public static double getGromovFrechetDistance(MatOfPoint contour1, MatOfPoint contour2) повертає відстань Громова-Фреше private Point getContourCenter(MatOfPoint contour) повертає центри контурів public List<Point> rotate(List<Point> contour, double angle, Point center) виконує поворот public double getAngle(List<Point> contour) повертає кут private MatOfPoint listToMatOfPoint(List<Point> contour) конвертує List<Point> в тип MatOfPoint </pre>

Опис класу для прорідження контура приведений в таблиці 3.7.

Package utils;

Таблиця 3.7 – Опис класу для прорідження контура

Клас	<p>PolySimplifier Вміщує методи для прорідження контура</p>
Методи	<pre>public static List<Point> reduceSimple(List<Point> points, int step) List<Point> points – масив точок Int step - крок викидає точки з масиву з певним кроком</pre> <pre>public static List<Point> reduceRDP(List<Point> inputPointList, double epsilon) алгоритм Дугласа-Рамера-Пекара epsilon - максимальна довжина відхилення від перпендикуляра (чим більше тим більше спрощення)</pre> <pre>public static List<Point> reduceVW(List<Point> inputPointList, int numberToKeep) Алгоритм Visvalingam-Whyatt numberToKeep - кількість точок у вихідній ламаній</pre>

Опис класу для знаходження контура, прорідження контура приведений в таблиці 3.8.

Таблиця 3.8 – Опис класу для знаходження контура

Клас	<p>ImageOperations знаходження контурів, прорідження і т.д. кількість точок, до якої буде спрощено вхідний контур private static int NUM_TO_KEEP = 30;</p>
------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Продовження таблиці 3.8

Методи	<pre>public static List<MatOfPoint> prepareContours(Mat img) img- вхідне зображення вихід в форматі List<MatOfPoint> конвертує Mat в List private static List<MatOfPoint> getContours(Mat mat) повертає список контурів вихід в форматі List<MatOfPoint> конвертує Mat в List public static Mat prepareImage(String path, String treshType) String path – шлях до зображення String treshType – застосовується для порогової сегментації конвертує вхідне зображення в формат opencv Mat</pre>
--------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Опис класу допоміжних методів для геометричних обчислень в таблиці 3.9.

Таблиця 3.9 – Опис класу допоміжних методів для геометричних обчислень

Клас	<p>GeometryUtils Допоміжні методи для геометричних обчислень</p>
Методи	<pre>public static double getEuclideanDistance(Point p1, Point p2) повертає евклідову відстань між двома точками public static Point getPerpendicularBasePoint(Point fromPoint, Point startSeg, Point endSeg) шукає координати основи перпендикуляра опущеної з точки на відрізок повертає null - якщо перпендикуляр провести не можна</pre>

Продовження таблиці 3.9

	<pre> public static double getTriangleArea(Point a, Point b, Point c) площа трикутника за 3 точками, формула Герона public static List<Contour> getContours(Mat image, String msg, double distCoef, double overlapCoef) повертає список контурів List<Contour> public static ArrayList<Double> getStraightCoefficients(Point A, Point B) повертає коефіцієнти рівняння прямої що проходить через дві точки (A,B) public static Point getIntersectionPointOfLines(Point A1, Point B1, Point A2, Point B2) повертає точку перетину прямої (що проходить через A1,B1) і відрізка (A2,B2), якщо така є static public double getScalarProduct(Point startA, Point endA, Point startB, Point endB) повертає скалярний добуток векторів // можливі випадки розташування точки відносно вектора public enum Position { LEFT, // точка зліва від напрямку вектора RIGHT, // точка справа від напрямку вектора AHEAD, // точка спереду вектора (на одній прямій) BEHIND, // точка позаду вектора (на одній прямій) LIES_ON // точка лежить на векторі </pre>
--	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Продовження таблиці 3.9

	<pre> } static public Position getPointPosition(Point point, Point vectorStart, Point vectorEnd) повертає позицію точки відносно напрямку вектора </pre>
--	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Опис класу конструктора приведений в таблиці 3.10.

Package qualityEstimator;

Таблиця 3.10 – Опис класу конструктора

Клас	MainEstimator
Конструктор	<pre> public void Estimator(Mat im1, Mat im2) приймає 2 зображення в форматі opencv mat float distCoef = (float)0.5; float overlapCoef = (float)0.6; вказуються коефіцієнти double[] dist1 = byLongestChorde(im1, im2); double[] dist2 = byLongestChordeAndPerpendicular(im1.clone(), im2.clone()); double[] dist3 = bySecantMethod(im1.clone(), im2.clone()); double[] dist4 = byListOfChordes(im1.clone(), im2.clone(),distCoef,overlapCoef); double[] dist5 = byFullSearch(im1.clone(), im2.clone()); виклик 5 методів ізометричних перетворень Далі відбувається реалізація 5 методів,викликаних вище. Контури порівнюються – кожний з кожним. </pre>

Опис класу методу повороту приведений в таблиці 3.11.

package qualityEstimator.core;

Таблиця 3.11 – Опис класу методу повороту

Клас	IsometricTranslation
Методи	<p data-bbox="794 461 1525 539">public static double distBy360Rotation (Contour contour1, Contour contour2)</p> <p data-bbox="890 589 1453 712">метод повороту на 360 градусів. contour1, contour2 – вхідні контури Тип повернутого значення - double</p> <p data-bbox="794 759 1525 882">public static double distBySecantMethod(Contour3 contour1, Contour3 contour2)</p> <p data-bbox="890 931 1453 1010">метод січних Тип повернутого значення - double</p> <p data-bbox="794 1059 1525 1182">public static double distByLongestChord(Contour contour1, Contour contour2)</p> <p data-bbox="890 1187 1453 1265">відстань за найбільшою хордою Тип повернутого значення - double</p> <p data-bbox="794 1314 1525 1438">public static double distByListOfChordes(Contour contour1, Contour contour2)</p> <p data-bbox="890 1487 1453 1565">за списком хорд Тип повернутого значення - double</p> <p data-bbox="794 1615 1525 1738">public static double distByLongestChordAndPerpendicular(Contour2 contour1, Contour2 contour2)</p> <p data-bbox="794 1787 1525 1910">за найдовшою хордою та перпендикуляром Тип повернутого значення - double</p>

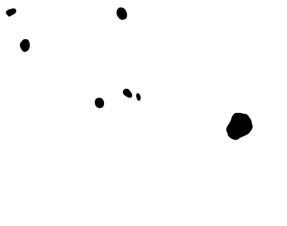

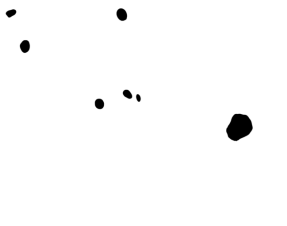

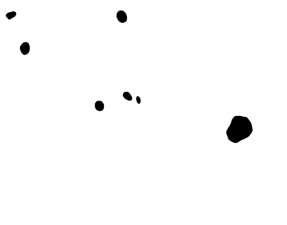
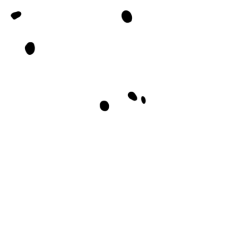
package qualityEstimator.core.Contour;

Вміщує класи для роботи з контурами
package qualityEstimator.core.Chorde;
Вміщує класи для роботи з хордами.

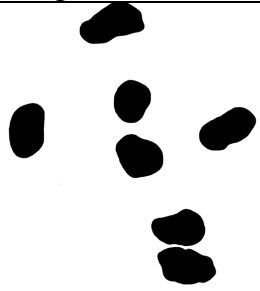
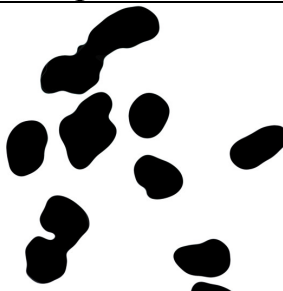
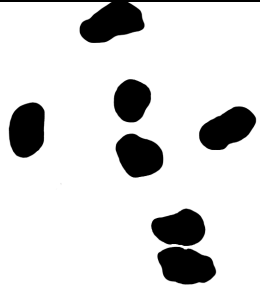

3.3 Тестування розробленого програмного модуля

Тестування проводилося на прикладі цитологічних зображень [56-58]. У таблицях 3.12 – 3.21 приведені результати тестування модуля оцінки якості сегментації з використанням сегментації експертів та сегментації на основі алгоритмів порогової сегментації та алгоритму водоподілу.

Таблиця 3.12 – Результати тестування модуля на зображеннях типу 1

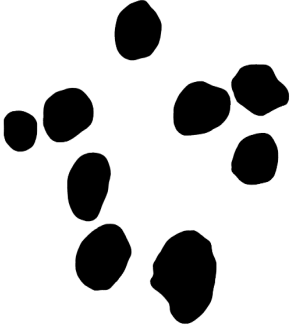
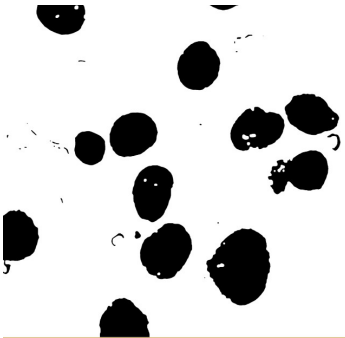
Зображення 1	Зображення 1	Результат
 <p data-bbox="320 517 451 555">Експерт</p>	 <p data-bbox="576 506 767 584">Порогова сегментація</p>	<pre data-bbox="863 286 1533 855"> ===== Comparator Module/images/1_threshold.png => Comp Hausdorff. distance - 249.0020080240318 Time - 4 millis Attalah. distance - 249.0020080240318 Time - 1 millis Gromov-Hausdorff. distance - 1138.0281191605065 Time - 2 millis Frechet. distance - 249.0020080240318 Time - 1 millis Gromov-Frechet. distance - 2442.218868160673 Time - 2 millis Result: 1790.1234936605897 </pre>
 <p data-bbox="225 1113 355 1151">Експерт</p>	 <p data-bbox="576 1079 767 1122">Водоподілу</p>	<pre data-bbox="863 882 1533 1460"> ===== Comparator Module/images/1_watershed.png => Co Hausdorff. distance - 796.0025125588486 Time - 1 millis Attalah. distance - 3651.0913436943756 Time - 10 millis Gromov-Hausdorff. distance - 788.2042882400476 Time - 2 millis Frechet. distance - 796.0025125588486 Time - 2 millis Gromov-Frechet. distance - 788.2042882400476 Time - 4 millis Result: 788.2042882400476 </pre>
 <p data-bbox="320 1718 451 1756">Експерт</p>	 <p data-bbox="671 1718 802 1756">Експерт</p>	<pre data-bbox="863 1464 1533 1982"> Comparator Module/images/1_expert.png => Comp Hausdorff. distance - 0.0 Time - 9 millis Attalah. distance - 0.0 Time - 0 millis Gromov-Hausdorff. distance - 0.0 Time - 11 millis Frechet. distance - 0.0 Time - 8 millis Gromov-Frechet. distance - 0.0 Time - 4 millis Result: 0.0 </pre>

Таблиця 3.13 – Результати тестування модуля на зображеннях типу 2

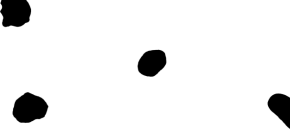
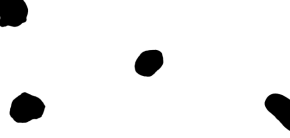

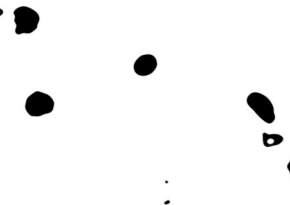
Зображення 2	Зображення 2	Результат
 <p data-bbox="320 548 453 584">Експерт</p>	 <p data-bbox="692 548 847 584">Порогова</p>	<pre data-bbox="975 282 1490 763"> ===== Comparator Module/images/3_threshold.png => Comp Hausdorff. distance - 5.385164807134504 Time - 16 millis Attalah. distance - 5.385164807134504 Time - 1 millis Gromov-Hausdorff. distance - 1148.015679335435 Time - 1 millis Frechet. distance - 3260.0001533742297 Time - 1 millis Gromov-Frechet. distance - 2440.2004016063925 Time - 2 millis Result: 1794.1080404709137 ===== </pre>
 <p data-bbox="320 1059 453 1095">Експерт</p>	 <p data-bbox="608 1137 794 1173">Водоподілу</p>	<pre data-bbox="975 792 1490 1274"> ===== Comparator Module/images/3_watershed.png => Co Hausdorff. distance - 4.242640687119285 Time - 0 millis Attalah. distance - 4.242640687119285 Time - 1 millis Gromov-Hausdorff. distance - 1143.010936080666 Time - 1 millis Frechet. distance - 4.242640687119285 Time - 0 millis Gromov-Frechet. distance - 2443.6000081846455 Time - 2 millis Result: 1793.3054721326557 ===== </pre>

Таблиця 3.14 – Результати тестування модуля на зображеннях типу 3

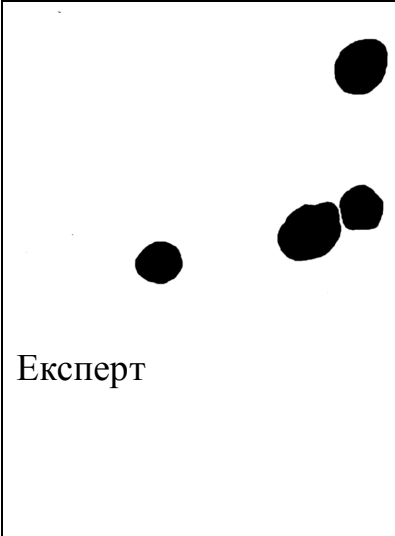
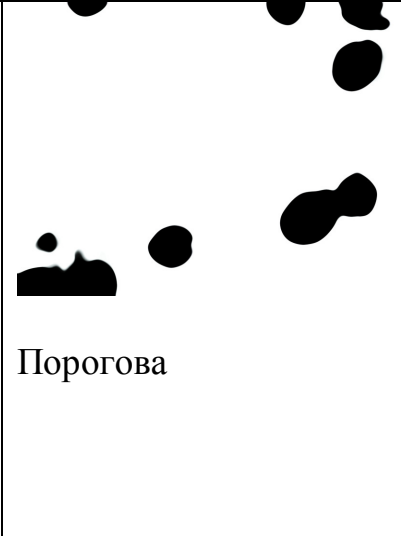
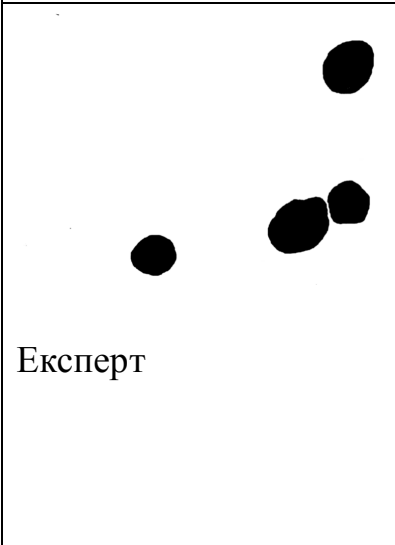
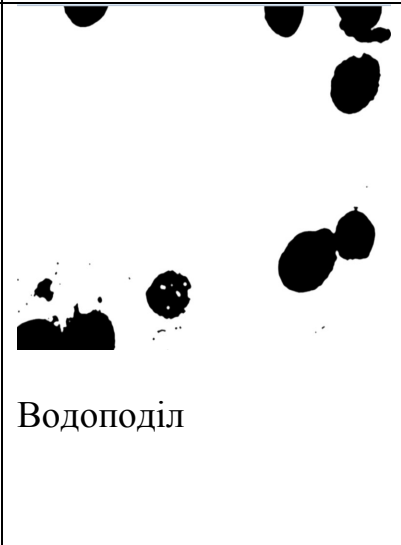
Зображення 3	Зображення 3	Результат
 <p data-bbox="320 1883 453 1919">Експерт</p>	 <p data-bbox="587 1883 742 1919">Порогова</p>	<pre data-bbox="960 1599 1452 1960"> Comparator Module/images/4_threshold.png => Comparato Hausdorff. distance - 1603.718491506536 Time - 5 millis Attalah. distance - 1603.718491506536 Time - 12 millis Gromov-Hausdorff. distance - 1603.2831939492162 Time - 13 millis Frechet. distance - 3013.0097908901657 Time - 1 millis Gromov-Frechet. distance - 3011.389712408542 Time - 3 millis Result: 2307.3364531788793 </pre>

 <p>Експерт</p>	 <p>Водоподілу</p>	<pre> Comparator Module/images/4_watershed.png => Compa Hausdorff. distance - 917.357618380095 Time - 1 millis Attalah. distance - 917.357618380095 Time - 0 millis Gromov-Hausdorff. distance - 917.357618380095 Time - 0 millis Frechet. distance - 917.357618380095 Time - 1 millis Gromov-Frechet. distance - 917.357618380095 Time - 0 millis Result: 917.357618380095 </pre>
--------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

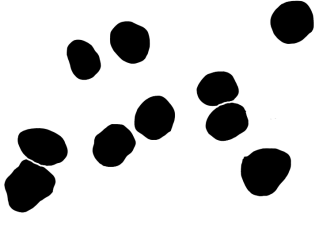
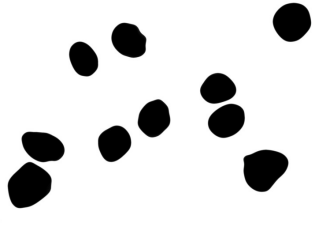
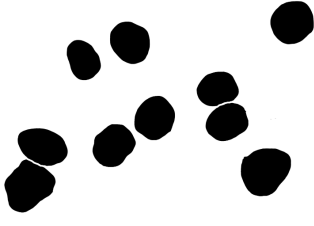

Таблиця 3.15 – Результати тестування модуля на зображеннях типу 4

Зображення 4	Зображення 4	Результат
 <p>Експерт</p>	 <p>Експерт</p>	<pre> Comparator Module/images/5_expert.png => Compa Hausdorff. distance - 0.0 Time - 9 millis Attalah. distance - 0.0 Time - 1 millis Gromov-Hausdorff. distance - 0.0 Time - 10 millis Frechet. distance - 0.0 Time - 5 millis Gromov-Frechet. distance - 0.0 Time - 11 millis Result: 0.0 </pre>
 <p>Експерт</p>	 <p>Порогова</p>	<pre> Comparator Module/images/5_threshold.png => Compa Hausdorff. distance - 1602.7189398019855 Time - 16 millis Attalah. distance - 1602.7189398019855 Time - 11 millis Gromov-Hausdorff. distance - 1612.3603815524618 Time - 3 millis Frechet. distance - 3012.010126144997 Time - 3 millis Gromov-Frechet. distance - 3619.958563298757 Time - 3 millis Result: 2616.1594724256092 </pre>

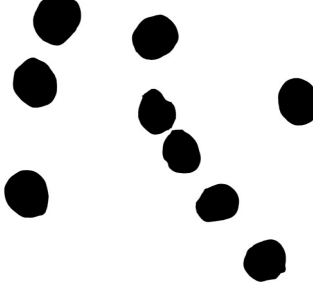
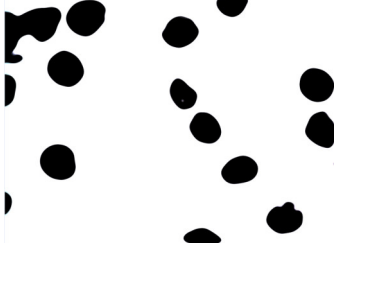
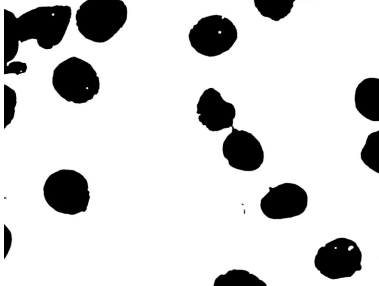
Таблиця 3.16 – Результати тестування модуля на зображеннях типу 5

Зображення 5	Зображення 5	Результат
<p>Експерт</p> 	<p>Порогова</p> 	<pre> ===== Comparator Module/images/8_threshold.png => Co Hausdorff. distance - 1117.0040286408998 Time - 4 millis Attalah. distance - 1117.0040286408998 Time - 2 millis Gromov-Hausdorff. distance - 1264.227827569066 Time - 2 millis Frechet. distance - 2020.0 Time - 1 millis Gromov-Frechet. distance - 1277.283445441927 Time - 1 millis Result: 1270.7556365054966 </pre>
<p>Експерт</p> 	<p>Водоподіл</p> 	<pre> ===== Comparator Module/images/8_watershed.png => Com Hausdorff. distance - 664.0 Time - 2 millis Attalah. distance - 664.0 Time - 1 millis Gromov-Hausdorff. distance - 663.6000301386371 Time - 2 millis Frechet. distance - 755.0 Time - 0 millis Gromov-Frechet. distance - 753.6000265392777 Time - 2 millis Result: 708.6000283389574 </pre>

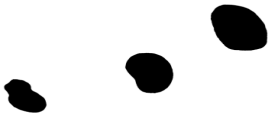
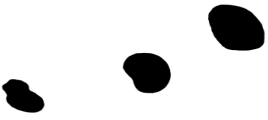


Таблиця 3.17 – Результати тестування модуля на зображеннях типу 6

Зображення 6	Зображення 6	Результат
 <p data-bbox="225 633 359 674">Експерт</p>	 <p data-bbox="596 593 751 633">Порогова</p>	<pre data-bbox="986 347 1535 817"> ===== Comparator Module/images/9_threshold.png => Com Hausdorff. distance - 996.5139236357915 Time - 5 millis Attalah. distance - 996.5139236357915 Time - 1 millis Gromov-Hausdorff. distance - 995.3140207994661 Time - 2 millis Frechet. distance - 996.5139236357915 Time - 5 millis Gromov-Frechet. distance - 995.3140207994661 Time - 4 millis Result: 995.3140207994661 ===== </pre>
 <p data-bbox="225 1184 359 1225">Експерт</p>	 <p data-bbox="596 1184 778 1225">Водоподіл</p>	<pre data-bbox="986 896 1535 1361"> ===== Comparator Module/images/9_watershed.png => Compa Hausdorff. distance - 41.677331968349414 Time - 1 millis Attalah. distance - 41.677331968349414 Time - 1 millis Gromov-Hausdorff. distance - 41.23105625617661 Time - 3 millis Frechet. distance - 41.677331968349414 Time - 1 millis Gromov-Frechet. distance - 41.23105625617661 Time - 2 millis Result: 41.23105625617661 ===== </pre>





Таблиця 3.18 – Результати тестування модуля на зображеннях типу 7

Зображення 7	Зображення 7	Результат
 <p>Експерт</p>	 <p>Порогова</p>	<pre> Comparator Module/images/10_threshold.png => Co Hausdorff. distance - 590.0279654389273 Time - 6 millis Attalah. distance - 590.0279654389273 Time - 10 millis Gromov-Hausdorff. distance - 589.8966011090419 Time - 3 millis Frechet. distance - 2583.5827836552867 Time - 3 millis Gromov-Frechet. distance - 2582.8892736623457 Time - 3 millis Result: 1586.3929373856938 </pre>
<p>Експерт</p>	 <p>Водподіл</p>	<pre> Comparator Module/images/10_watershed.png => Co Hausdorff. distance - 1091.3807768143986 Time - 1 millis Attalah. distance - 1091.3807768143986 Time - 2 millis Gromov-Hausdorff. distance - 1515.7747194091871 Time - 2 millis Frechet. distance - 1519.5795471116344 Time - 2 millis Gromov-Frechet. distance - 1515.7747194091871 Time - 1 millis Result: 1515.7747194091871 </pre>

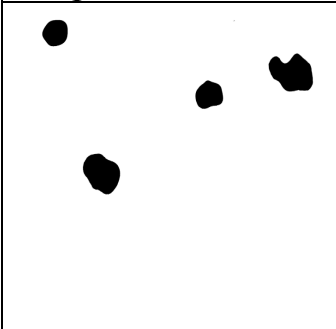
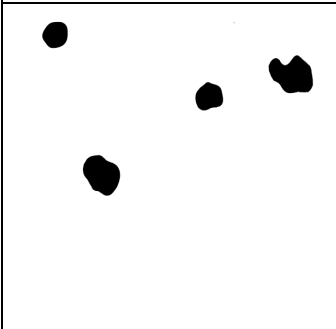
Таблиця 3.19 – Результати тестування модуля на зображеннях типу 8

Зображення 8	Зображення 8	Результат
 <p data-bbox="225 544 357 584">Експерт</p>	 <p data-bbox="604 544 737 584">Експерт</p>	<pre data-bbox="991 264 1513 714"> Comparator Module/images/11_expert.png => Comparat Hausdorff. distance - 0.0 Time - 6 millis Attalah. distance - 0.0 Time - 1 millis Gromov-Hausdorff. distance - 0.0 Time - 9 millis Frechet. distance - 0.0 Time - 5 millis Gromov-Frechet. distance - 0.0 Time - 5 millis Result: 0.0 </pre>
 <p data-bbox="225 999 357 1039">Експерт</p>	 <p data-bbox="604 1061 756 1102">Порогова</p>	<pre data-bbox="991 719 1513 1225"> Comparator Module/images/11_threshold.png => Comparator Hausdorff. distance - 1199.9741663885936 Time - 5 millis Attalah. distance - 1199.9741663885936 Time - 19 millis Gromov-Hausdorff. distance - 1200.0854136268802 Time - 2 millis Frechet. distance - 3261.2827231014485 Time - 0 millis Gromov-Frechet. distance - 3260.1818967658846 Time - 1 millis Result: 2230.1336551963823 </pre>

Таблиця 3.20 – Результати тестування модуля на зображеннях типу 9

Зображення 9	Зображення 9	Результат
 <p data-bbox="225 663 357 696">Експерт</p>	 <p data-bbox="647 663 780 696">Експерт</p>	<pre data-bbox="1034 338 1513 842"> Comparator Module/images/12_expert.png Hausdorff. distance - 0.0 Time - 7 millis Attalah. distance - 0.0 Time - 0 millis Gromov-Hausdorff. distance - 0.0 Time - 10 millis Frechet. distance - 0.0 Time - 4 millis Gromov-Frechet. distance - 0.0 Time - 4 millis Result: 0.0 </pre>
 <p data-bbox="225 1193 357 1227">Експерт</p>	 <p data-bbox="647 1238 802 1272">Порогова</p>	<pre data-bbox="1034 866 1519 1361"> Comparator Module/images/12_threshold.png => Comp Hausdorff. distance - 927.0 Time - 6 millis Attalah. distance - 927.0 Time - 8 millis Gromov-Hausdorff. distance - 924.0779187925659 Time - 3 millis Frechet. distance - 927.0 Time - 2 millis Gromov-Frechet. distance - 924.4003461704242 Time - 3 millis Result: 924.2391324814951 </pre>

Таблиця 3.21 – Результати тестування модуля на зображеннях типу 10

Зображення 10	Зображення 10	Результат
 <p data-bbox="212 649 549 891">Експерт</p>	 <p data-bbox="549 649 868 891">Експерт</p>	<pre data-bbox="868 322 1540 891"> Comparator Module/images/13_expert.png => C Hausdorff. distance - 0.0 Time - 8 millis Attalah. distance - 0.0 Time - 0 millis Gromov-Hausdorff. distance - 0.0 Time - 11 millis Frechet. distance - 0.0 Time - 4 millis Gromov-Frechet. distance - 0.0 Time - 17 millis Result: 0.0 </pre>
 <p data-bbox="212 1218 549 1545">Експерт</p>	 <p data-bbox="549 1218 868 1545">Порогова</p>	<pre data-bbox="868 891 1540 1545"> ===== Comparator Module/images/13_threshold.png => C Hausdorff. distance - 1202.5643433929013 Time - 4 millis Attalah. distance - 1202.5643433929013 Time - 8 millis Gromov-Hausdorff. distance - 1201.697549302652 Time - 2 millis Frechet. distance - 1202.5643433929013 Time - 1 millis Gromov-Frechet. distance - 1201.697549302652 Time - 2 millis Result: 1201.697549302652 </pre>

3.4 Висновки до розділу 3

В розділі сформовано системні вимоги до програмного модуля, розроблено його архітектуру. За допомогою цього спроектовано та програмно реалізовано модуль порівняння контурів зображень в метриці Фреше.

Проведено тестування програмного модуля та комп'ютерні експерименти на прикладі цитологічних зображень, що дало можливість оцінювати відстань між контурами зображень.

ВИСНОВКИ

1. Проведено аналіз контурів зображень та методи їх опису, введено поняття нечітких множин та операції над ними, проаналізовані, здійснено постановку задач дослідження.
2. Проаналізовано в загальному метричні простори, що дало можливість використати метрики Фреше та Хаусдорфа для визначення подібності зображень.
3. Проаналізовано метрику Фреше та метрику Громова-Фреше для порівняння контурів зображень.
4. Проаналізовано метрику Хаусдорфа, метрику Громова-Хаусдорфа для порівняння областей зображень.
5. Розроблено алгоритм порівняння контурів зображень в метриці - Фреше на основі алгоритму визначення дискретної відстані Фреше i , що дало можливість оцінити відстань між двома контурами.
6. Сформовано системні вимоги до програмного модуля, розроблено його архітектуру, що дало можливість спроектувати та програмно реалізувати модуль порівняння контурів зображень в метриці Фреше.
7. Проведено тестування програмного модуля та комп'ютерні експерименти на прикладі цитологічних зображень, що дало можливість оцінювати відстань між контурами зображень.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Березький О. Аналіз алгоритмів виділення контурів зображень / О. Березький, Ю. Батько // Вісник Національного університету «Львівська політехніка». Комп'ютерні науки та інформаційні технології. – 2006. – № 565. – С. 212-216.
2. Berezsky O. Contour analysis of images / O. Berezsky // Information technologies and Systems. – 2006. – V.9, №1. – P.5-12.
3. Березький О. М. Аналіз контурів зображень у глобальних координатах / О. М. Березький, К. М. Березька // Оптико-електронні інформаційно-енергетичні технології. – 2006. – №2(12). – С.83-92.
4. Berezsky O. Algorithm of determination of image contours of biological nature / Oleg Berezsky, Yu Bat'ko // Proceedings of the International conference «Modern problems of radio engineering, telecommunications and computer science» TCSET2006 ' 2006. Lviv-Slavske. – Lviv, 2006. – P. 642-644.
5. Berezsky O. M. Image contour analysis in local coordinates / O. M. Berezsky, K.M. Berezska, O.P. Adamiv // Proceedings of the 4-th IEEE workshop on intelligent data advanced computing systems: technology and applications (IDAACS-2007). – Dortmund, Germany, 2007. – P. 393-398
6. Березький О. М. Перетворення контурів зображень в афінному просторі / О. М. Березький, К.М. Березька, Ю.М. Батько, Г.М.Мельник // Матеріали Міжнародної конференції «Інтелектуальні системи прийняття рішень і проблеми обчислювального інтелекту». Євпаторія, 2008. Т.3, Ч.1. С. 51-55.
7. Березький О. М. Методи та алгоритми перетворення контурів зображень в афінному просторі / О. М. Березький // Вісник Національного університету «Львівська політехніка». Комп'ютерні науки та інформаційні технології. – 2009. – № 638. – С. 185-189.

8. Березький О. М. Методи та алгоритми перетворення контурів зображень в афінному просторі / О. М. Березький // Вісник Національного університету «Львівська політехніка». Комп'ютерні науки та інформаційні технології. – 2009. – № 638. – С. 185-189.

9. Березький О. М. Алгоритм проходження контуром об'єкта з використанням зворотного ходу / О. М. Березький, Ю.М. Батько // Штучний інтелект – 2009. – №3. – С.116-122.

10. Березский О. Н. Топологические методы и алгоритмы преобразования контуров и областей плоских изображений / О. Н. Березский // Проблемы информатики и управления. – 2010. – № 5. – С.123-131.

11. Berezsky O. Contour correspondence points detection for biomedical images analysis / Berezsky O., Batko Yu., Melnyk G. // Proceedings of the VIIIth International Conference on Perspective Technologies and Methods in MEMS Design (MEMSTECH'2012), Lviv – Polyana, April 18-21, 2012. – Lviv, 2012. – P. 57-59.

12. Berezsky O. Image contours transformation errors / Berezsky Oleg, Melnyk Grygoriy, Batko Yuriy // Elektronica. – 2013. – №8. – С.49-51.

13. Березький О. М. Дослідження похибки перетворення контурів біомедичних зображень / О. М. Березький // Науковий вісник НЛТУ України: зб. наук.-техн. праць. – Львів: РВВ НЛТУ України. – 2013. – Вип. 23.17. – С. 352-359.

14. Березький О. М. Похибки перетворення типу «контур-контур» в автоматизованих системах опрацювання біомедичних зображень / О. М. Березький // Вісник Національного університету «Львівська політехніка». Комп'ютерні науки та інформаційні технології. – 2013. – № 771. – С. 11-16.

15. Вдодович О. В., Клімовський Д. Б., Полагнюк І. В., Бучинський Т. Б. Алгоритми порівняння зображень в метричних просторах. Інтелектуальні комп'ютерні системи та мережі : тези доп. V Наук.-практ. конф. молодих вчених і студентів (2 груд. 2021 р.). Тернопіль : ЗУНУ, 2021. С. 11.

16. Полагнюк І. В., Клімовський Д. Б., Вдодович О. В., Бучинський Т. Б. Сегментація зображень з використанням метричних мір. Інтелектуальні комп'ютерні системи та мережі : тези доп. V Наук.-практ. конф. молодих вчених і студентів (2 груд. 2021 р.). Тернопіль : ЗУНУ, 2021. С. 14.

17. Березький О.М., Дубчак Л.О., Мельник Г.М. Методичні рекомендації до виконання кваліфікаційної роботи з освітнього ступеня "Магістр". Спеціальність: 123 - Комп'ютерна інженерія. Магістерська програма - Комп'ютерна інженерія". Тернопіль: ЗУНУ, 2021. 32 с.

18. Методичні вказівки до оформлення курсових проектів, звітів про проходження практики, випускних кваліфікаційних робіт для студентів спеціальності «Комп'ютерна інженерія» / І.В. Гураль, Л.О. Дубчак / Під ред. О.М. Березького. Тернопіль: ТНЕУ, 2019. 33 с.

19. Гонсалес Р. Цифровая обработка изображений / Р. Гонсалес, Р. Вудс – М.: Техносфера. – 2005.–1072 с.

20. Fackler M.J., Veigh M.M., Mehrotra J., Blum M.A., Lange J., Lapidis A., Garrett E., Argani P., Sukumar S. (2004) *Cancer Res.*, 64, 4442–4452.

21. Blum H. A transformation for extracting new descriptors of shape. In *Proc. Symposium Models for the perception of speech and visual form*. MIT Press, Cambridge, MA, 1967.

22. Pierre Soille, Martino Pesaresi, and Georgios K. Ouzounis, editors. *Mathematical Morphology and Its Applications to Image and Signal Processing - 10th International Symposium, ISMM 2011, Verbania-Intra, Italy, July 6-8, 2011. Proceedings*, volume 6671 of *Lecture Notes in Computer Science*. Springer, 2011.

23. Рутковская Д., Пилинский М., Рутковский Л. Нейронные сети, генетические алгоритмы и нечеткие системы. М.: Горячая линия –Телеком, 2004. 452 с.

24. Michel Marie Deza and Elena Deza. *Encyclopedia of distances*, pages 1–583. Springer, 2009.

25. D. Burago, Y. Burago, and S. Ivanov, A Course in Metric Geometry, vol. 33 of Graduate Studies in Mathematics. Providence, Rhode Island: American Mathematical Society, June 2001.
26. M. J. Atallah, A linear time algorithm for the Hausdorff distance between convex polygons, Inform. Process. Lett., 17: 207–209, 1983.
27. M. Gromov, Groups of Polynomial growth and Expanding Maps, Publications mathematiques I.H.E.S., 53, 1981.
28. Alexey A. Tuzhilin, Who Invented the Gromov-Hausdorff Distance?, arXiv:1612.00728v1.
29. D.A. Edwards The Structure of Superspace, published in: Studies in Topology, Academic Press, 1975.
30. Facundo Mémoli, Gromov-Hausdorff distances in Euclidean spaces, in 2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, (Anchorage, AK, USA), pp. 1–8, IEEE, June 2008.
31. Sushovan Majhi, Jeffrey Vitter, Carola Wenk, Approximating Gromov-Hausdorff Distance in Euclidean Space, arXiv:1912.13008v1.
32. P. K. Agarwal, K. Fox, A. Nath, A. Sidiropoulos, and Y. Wang, Computing the Gromov-Hausdorff Distance for Metric Trees, ACM Trans. Algorithms, vol. 14, pp. 24:1–24:20, Apr.2018.
33. Facundo Mémoli, Zane Smith, and Zhengchao Wan, Gromov-Hausdorff distances on P -metric spaces and ultrametric spaces, arXiv:1912.00564v3.
34. Ihor Zarichnyi, Gromov-Hausdorff ultrametric, arXiv preprint math/0511437, 2005.
35. Samir Chowdhury. Metric and Topological Approaches to Network Data Analysis. PhD thesis, The Ohio State University, 2019.
36. A.M. Vershik, Kantorovich Metric: Initial History and Little-Known Applications. J Math Sci 133, 1410–1417 (2006).
37. Carlos A. Cabrelli, Ursula M.Molter, The Kantorovich metric for probability measures on the circle, Journal of Computational and Applied Mathematics, Volume 57, Issue 3, 1995, 345–361.

38. Federico Basseti, Stefano Gualandi, Marco Veneroni, On the Computation of Kantorovich-Wasserstein Distances between 2D-Histograms by Uncapacitated Minimum Cost Flows, arXiv:1804.00445v3.
39. F. de Goes, D. Cohen-Steiner, P. Alliez, and M. Desbrun. An optimal transport approach to robust reconstruction and simplification of 2D shapes. *Computer Graphics Forum*, 30(5):1593–1602, 2011.
40. O. Frigo, N. Sabater, V. Demoulin, and P. Hellier. Optimal transportation for example-guided color transfer. In *Asian Conference on Computer Vision (ACCV'14)*, V. 9005, 655–670, 2014.
41. W. Gangbo and R. J. McCann. Shape recognition via wasserstein distance. *Quarterly of Applied Mathematics*, LVIII(4):705–737, 2000.
42. J. Lellmann, D. A. Lorenz, C. Schönlieb, and T. Valkonen. Imaging with Kantorovich-Rubinstein discrepancy. *SIAM Journal on Imaging Sciences*, 7(4):2833–2859, 2014.
43. Y. Rubner, C. Tomasi, and L. Guibas. A metric for distributions with applications to image databases. In *IEEE International Conference on Computer Vision (ICCV'98)*, 59–66, 1998.
44. J. Rabin and N. Papadakis. Convex color image segmentation with optimal transport distances. In *Scale Space and Variational Methods in Computer Vision (SSVM'15)*, 241–252, 2015.
45. Facundo Mémoli, Gromov–Wasserstein Distances and the Metric Approach to Object Matching, *Found Comput Math* (2011) 11:417–487.
46. Alt H., Godau M. Computing the Fréchet distance between two polygonal curves, *Int. J. of Computational Geometry and Applications*. 1995. - 5. 75–91.
47. L.E.Bazylevych, M.M. Zarichnyi, On metrization of the hyperspace of oriented curves. *Visn. Lviv. Univ. Ser. mekh.-mat.* 1996. V. 43, 3–5.
48. O. Berezhsky, M. Zarichnyi. Gromov-Fréchet distance between curves, *Mat. Stud.* 50 (2018), 88–92.
49. J. Alber, R. Niedermeier, On multidimensional curves with Hilbert property, *Theory of Computing Systems*, 33 (2000), No 4, 295–312.

50. H. Alt, M. Buchin, Can we compute the similarity between surfaces? *Discrete Comput. Geom.*, 43 (2010), No 1, 78–99.

51. J.G. Camarena, V. Gregori, S. Morillas, A. Sapena, Fast detection and removal of impulsive noise using peer groups and fuzzy metrics, *Journal of Visual Communication and Image Representation* 19 (2008) 20-29.

52. D.H. Fremlin, Skeletons and Central Sets, *Proc. London Math. Soc.*, Volume 74 (1997), Issue 3, 701–720.

53. C.J. Bishop, H. Hakobyan, A central set of dimension 2, *Proc. Amer. Math. Soc.* 136 (2008), no. 7, 2453–2461.

54. O. Berezska, M. Zarichnyi. Fréchet distance between weighted rooted trees, *Mat. Stud.* 48 (2017), 165–170.

55. Berezska, O. Gromov-Fréchet distance between curves / O. Berezska, M. Zarichnyi // *Matematychni Studii.* – 2018. – Vol. 50, No.1. – P. 88-92.

56. Березький О. М. База даних цитологічних та гістологічних зображень ауто- та ксеногенних тканин / О. М. Березький, Г.М. Мельник, Т.В. Дацко, С. О. Вербовий // *Науковий вісник НЛТУ України: зб. наук.-техн. праць.* – Львів: РВВ НЛТУ України. – 2014. – Вип. 24.10. – С. 338-345.

57. База даних цифрових гістологічних та цитологічних зображень різних форм раку молочної залози «CIFDB» («CIFDB») / Березький О. М., Мельник Г.М., Николук В.Д., Дацко Т.В.: Свідоцтво про реєстрацію авторського права на твір № 52743 від 23.12.2013 р.

58. Свідоцтво про реєстрацію авторського права на твір №75359. База даних цифрових гістологічних та цитологічних зображень передракових та ракових станів молочної залози «ВРСІ2100». / О.М. Березький, Г.М. Мельник, С.О. Вербовий, О.Й. Піцун, В.Д. Николук, Т.В. Дацко. Дата реєстрації 14.12.2017 р.