

# ВЕРИФІКАЦІЯ ВІДБИТКІВ ПАЛЬЦІВ З ВИКОРИСТАННЯМ РІШЕННЯ ЗАДАЧІ КОМІВОЯЖЕРА І ДЕКОМПОЗИЦІЇ ОТОЧЕННЯ МІНУЦІЙ

Ольга Мелкозьорова, Сергій Малахов, Валерія Гайкова

Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків, 61022, Україна  
[olha.melkozerova@karazin.ua](mailto:olha.melkozerova@karazin.ua), [mailgate@meta.ua](mailto:mailgate@meta.ua), [valeriagaikova98@gmail.com](mailto:valeriagaikova98@gmail.com)

Рецензент: В'ячеслав Калашников, д.ф.-м.н., проф., Технологічний університет Монтеррея,  
64849 Монтеррей, Нуево-Леон, Мексика  
[kalash@itesm.mx](mailto:kalash@itesm.mx)

Надійшла: Липень 2020.

**Анотація.** У статті наведено приклад верифікації бази відбитків пальців методом рішення задачі комівояжера з використанням декомпозиції околиць найближчих мінуцій. Рішення цієї задачі має стійкість до лінійних, кутових деформацій, перемішуванню точок. Цей метод забезпечує правильне рішення для невеликої кількості точок, для великої кількості точок виникають перетин контурів, рішення при цьому не є оптимальним. Тому для зменшення часу обробки та розрахунку метрики наведено змінений алгоритм рішення задачі методом гілок та границь, а саме вирівнювання та виключення дуг на кожному циклі пошуку оптимального маршруту. Верифікація базується на створенні локальних структур для кожної мінуції відбитку, тому що саме локальні структури мають стійкість до деформацій. Побудова глобальних структур дуже часто не призводить до гарних показників якості, так як виникає проблема при центруванні всього зразку. Проведено повний перебір випробувань шаблонів бази даних відбитків пальців при їх верифікації цим методом. Використання декомпозиції характерних ознак забезпечує більшу стійкість при дописуванні помилкових та стираних справжніх мінуцій. В результатах статті наведено значення парних порівнянь двох шаблонів для справжніх та помилкових випробувань. Досліджено показники помилкової відмови ( $FRR$  – false rejection rate), помилкового доступу ( $FAR$  – false acceptance rate), єдиної еквівалентної помилки ( $EER$  – equal error rate).

**Ключові слова:** відбитки пальців; мінуція; найкоротша відстань; локальні ознаки; оптимальний маршрут.

## 1 Вступ

На поточний момент часу рішення задачі, що пов'язана з верифікацією відбитків пальців все ще продовжує залишатися актуальною в багатьох сферах діяльності, наприклад, таких як, криміналістика, інформаційні технології, банківська справа та ін.

У представленій роботі розглядається метрика, яка заснована на побудові локальної структури найменшої відстані між точками. Локальна структура - це така структура, система координат якої, безпосередньо пов'язана з конкретною мінуцією, а не з системою координат для відбитка в цілому. Важливо відзначити, що локальні метрики мають стійкість до всіх типів спотворень, в тому числі лінійним і кутовим спотворенням. При цьому деякі з них дозволяють виключити проблему дописування помилкових мінуцій і видалення справжніх, що є найбільш суттєвою проблемою при рішенні задачі верифікації відбитків пальців [1-3].

Для побудови структури, авторами роботи використаний базовий алгоритм вирішення задачі комівояжера [4]. Однак, як відомо, при його використанні виникають проблема перебічення контурів [5] і для нього не описана можливість виключення дуг на маршруті при великій кількості ітерацій. Тому в даній статті розглянуто опис, яке дозволяє виключити зазначені труднощі і зменшити час на обробку всіх «дерев» при рішенні зазначеного вище завдання. Також, в рамках роботи, досліджені показники помилкової відмови ( $FAR$  – false acceptance rate) та єдиної еквівалентної помилки ( $EER$  – equal error rate) при повному переборі бази даних відбитків пальців.

## 2 Алгоритм вирішення задачі комівояжера

У даній роботі використаний алгоритм вирішення задачі комівояжера, основні етапи, якого представлені наступним чином:

1. Складання матриці взаємних відстаней, утворення матриці  $n \times n$ ;
2. Знаходження мінімальних значень за рядками;
3. Віднімання значень, які знайдені в пункті 2, з відповідних рядків;
4. Знаходження мінімальних значень взаємних відстаней в стовпцях матриці;
5. Віднімання значень, що отримані в п. 4, з відповідних стовпців (*т.ч., принаймні, один з елементів рядка стає нулем*);
6. Для кожного нуля знаходимо суму констант приведення, мінімальне значення по рядках і стовпцях;
7. З усіх нульових елементів вибираємо нульовий елемент з максимальною сумою констант приведення;
8. Включаємо дугу  $(i, j)$  в маршрут;
9. Виходячи з дуги, яка була включена в маршрут, виключаємо деяку дугу, шляхом введення знака  $\infty$  (*механізм виключення дуги описаний нижче*);
10. Викреслюємо  $i$ -й рядок,  $j$ -й стовець з матриці;
11. Повторюємо пункти 2-10 до тих пір, поки матриця не матимемо розмір  $(n-2) \times (n-2)$ ;
12. Дві інші дуги вибираємо з тих елементів, де не була встановлена заборона;
13. Переглядаємо отриманий маршрут на предмет наявності пересічень;
14. Видаляємо пересічення шляхом заміни двох точок на дугах (*детальний опис та приклади розглянуті нижче*).

## 3 Виключення дуги з маршруту

У 9-му пункті представлено вище опису алгоритму розв'язання задачі комівояжера, є т.зв. «виключення дуги з маршруту». Слід звернути увагу на те, що виключення цієї дуги є дуже важливим елементом алгоритму. Для великого числа точок, що входять в маршрут, цей аспект потребує додавання. Включення дуги  $(i, j)$  веде до утворення деякого пов'язаного шляху, який з'єднує, наприклад, точки  $p$  та  $q$ , забороняється включати в маршрут дугу  $(q, p)$ . У найпростішому випадку, після вибору дуги  $(i, j)$ , варто виключити дугу  $(j, i)$ , тобто елемент  $d_{j,i}$  слід замінити на значення  $\infty$ . Ця ситуація ілюструється нижче, на рис. 1. Так, наприклад, якщо в ході рішення отримана дуга  $(0,4)$ , то замінювати на нескінченне значення, слід довжину дуги  $(4,0)$ . Якщо ж в результаті рішення обрана послідовність дуг –  $(0,4)$ ,  $(2,3)$ , а наступною є дуга  $(0,2)$ , то виключати варто елемент  $(4,3)$  (рис. 2).

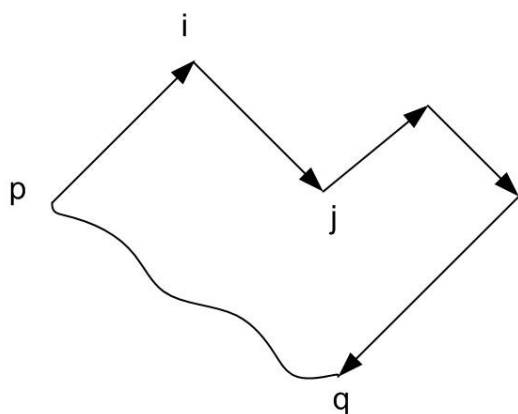


Рис. 1 – Пояснення до процедури виключення дуги

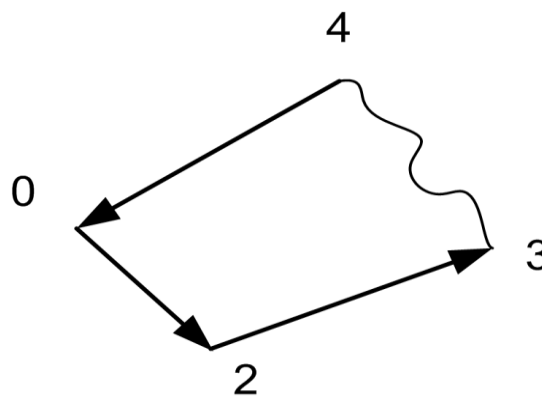


Рис. 2 – Виключення дуги  $(4, 3)$

В основному алгоритмі описаний випадок для асиметричної матриці, яка дозволяє отримувати замкнутий контур. Однак, в разі якщо матриця симетрична і кількість точок велике, то точки, що приходять на маршрут надходять не по порядку, а довільно, і таким чином виникають складнощі з пошуком дуг на заборону. Тому, в даному випадку, слід мати на увазі, що для установки на заборону дуг, необхідно розглянути весь маршрут, який складається на даній ітерації.

Так, якщо крайні точки дуги  $(i, j)$  були в маршруті один раз, то на заборону необхідно виставити дугу  $(j, i)$ . Якщо ж одна з точок була в маршруті кілька разів, то тоді, спочатку, слід видалити всі повторювані точки. Після цього слід видалити дуги на заборону, які були на попередніх ітераціях і обидві точки яких залишилися в послідовності точок без повтору. Таким чином, точки що залишилися і будуть тими точками дуги, котру слід заборонити.

Таблиця 1 – Приклад виключення дуги з маршруту

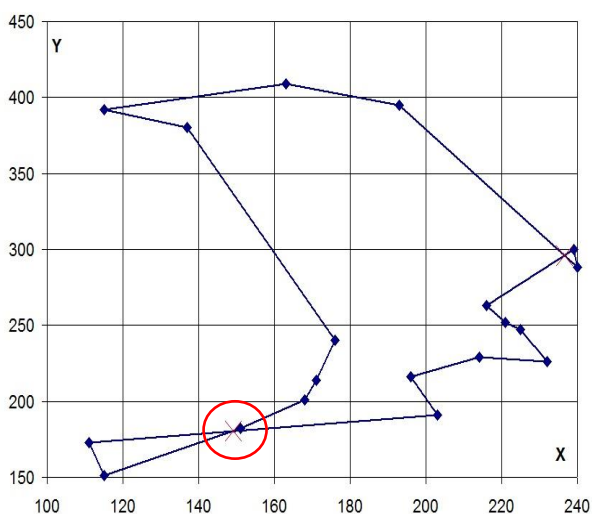
Номер ітерації	Дуга на маршруті	Дуга для заборони
0	14, 15	15, 14
1	16, 18	18, 16
2	3, 11	11, 3
3	11, 5	3, 5
4	4, 6	6, 4
5	6, 16	4, 18

У табл. 1 наведено приклад, який дозволяє продемонструвати, логіку вибору дуги на заборону. Так, на ітерації 0, маршрут 14, 15 за правилом, описаним вище, дуга на заборону 15, 14, то ж на ітераціях 1, 2. На ітерації 3, після видалення всіх повторів з маршруту, залишаються точки 14, 15, 16, 18, 3, 5. З цього ряду видаляємо 14, 15, 16, 18 (це дуги, які були заборонені на ітераціях 0 і 1), залишається дуга 3, 5, яка і є шуканою. На ітерації 5, після виключення всіх повторів з маршруту, залишаються точки 14, 15, 18, 3, 5, 4. З цього переліку видаляємо точки 14, 15, 3, 5, внаслідок

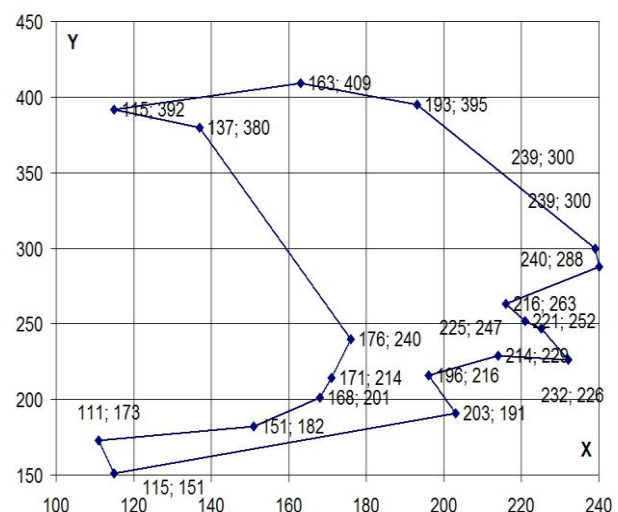
чого залишається дуга 4, 18, котру слід заборонити.

#### 4 Видалення пересічень контуру за базовим алгоритмом

Як вже було зазначено раніше, при вирішенні завдання комівояжера за базовим алгоритмом, який описаний в [1] виникають проблеми, що пов'язані з пересіченням відрізків маршруту (див. рис. 3(а) і Табл. 2). Так, наприклад, пересікаються відрізок з координатами (239, 300) – (216, 263) з відрізком (193, 395) – (240, 288), а також перетинається відрізок (203, 191) – (111, 173) з відрізком (115, 151) – (151, 182).



а) – рішення, що отримане за базовим алгоритмом;



б) – рішення, яке отримане після видалення пересічень.

Рис. 3 – Виключення пересічень з маршруту

Таблиця 2 – Дані для 20 точок  $X_i, Y_i$ 

Точки на маршруте	X	Y	X'	Y'	X''	Y''
0	176	240	240	288	239	300
1	171	214	239	300	240	288
2	168	201	216	263	216	263
3	111	173	221	252	221	252
4	137	380	225	247	225	247
5	151	182	232	226	232	226
6	115	392	214	229	214	229
7	196	216	196	216	196	216
8	216	263	203	191	203	191
9	221	252	111	173	115	151
10	214	229	115	151	111	173
11	115	151	151	182	151	182
12	225	247	168	201	168	201
13	203	191	171	214	171	214
14	240	288	176	240	176	240
15	239	300	137	380	137	380
16	163	409	115	392	115	392
17	232	226	163	409	163	409
18	193	395	193	395	193	395
19			240	288	239	300

-  $(X'_i, Y'_i)$  - одержане рішення;

-  $(X''_i, Y''_i)$  - одержане оптимальне рішення.

Ці пересічення збільшують маршрут, а рішення при цьому не є оптимальним, тобто довжина шляху 762, а без пересічень - 739. Точки пересічення при цьому (236, 295) і (149, 180) відповідно.

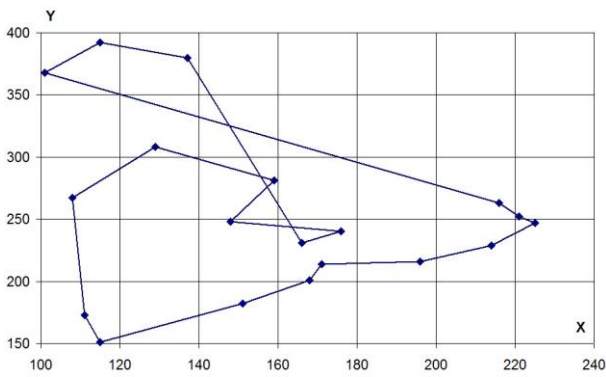
Вирішити цю проблему можна шляхом перестановки найближчих крайніх точок цих відрізків. Для цього необхідно точку з координатами (239, 300) одного відрізка, замінити на найближчу в контурі точку (240, 288). І навпаки, точку з координатами (240, 288), замінити на (239, 300). Аналогічну процедуру слід виконати з точками (111, 173) та (115, 151). Таким чином, буде сформований маршрут  $X'Y'$ , який є оптимальним маршрутом (див. рис. 3(б)).

Правило, для виключення таких пересічень працює і для більш складних контурів, дані по яких зведені в таблиці 3. Приклади контурів див. рис. 4.

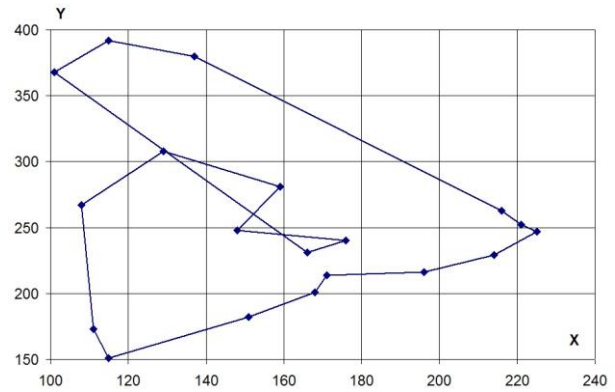
В даному випадку «вирівнювання» контуру відбувається за 5 ітерацій, але правило, яке описане вище, однозначно працює, і його можна з успіхом застосовувати для розв'язання задачі комівояжера.

Таблиця 3– Одержане рішення  $(X'_i, Y'_i)$  і одержане оптимальне рішення  $X_i''''', Y_i'''''$ 

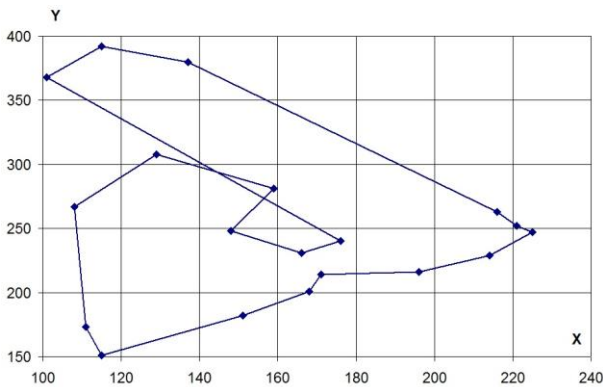
Точки на маршруті	X'	Y'	X''	Y''	X'''	Y'''	X''''	Y''''	X'''''	Y'''''	X''''''	Y''''''
0	111	173	111	173	111	173	111	173	111	173	111	173
1	115	151	115	151	115	151	115	151	115	151	115	151
2	151	182	151	182	151	182	151	182	151	182	151	182
3	168	201	168	201	168	201	168	201	168	201	168	201
4	171	214	171	214	171	214	171	214	171	214	171	214
5	196	216	196	216	196	216	196	216	196	216	196	216
6	214	229	214	229	214	229	214	229	214	229	214	229
7	225	247	225	247	225	247	225	247	225	247	225	247
8	221	252	221	252	221	252	221	252	221	252	221	252
9	216	263	216	263	216	263	216	263	216	263	216	263
10	101	368	137	380	137	380	137	380	137	380	137	380
11	115	392	115	392	115	392	115	392	115	392	115	392
12	137	380	101	368	101	368	101	368	101	368	101	368
13	166	231	166	231	176	240	148	248	129	308	129	308
14	176	240	176	240	166	231	166	231	166	231	159	281
15	148	248	148	248	148	248	176	240	176	240	176	240
16	159	281	159	281	159	281	159	281	159	281	166	231
17	129	308	129	308	129	308	129	308	148	248	148	248
18	108	267	108	267	108	267	108	267	108	267	108	267
19	111	173	111	173	111	173	111	173	111	173	111	173



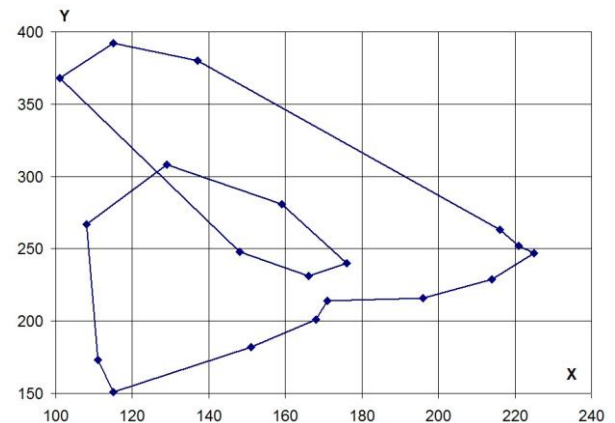
а) – опорний план  $X_i', Y_i'$



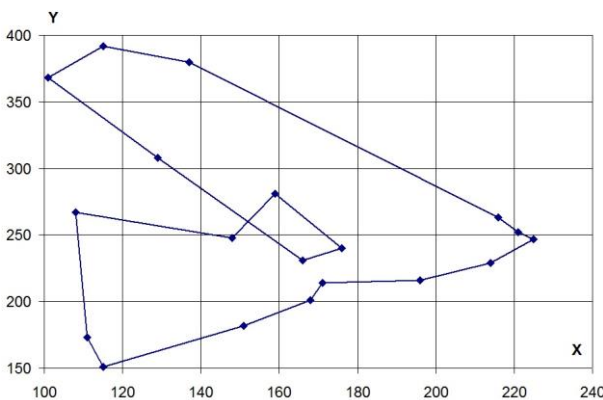
б) – перша ітерація видалення пересічень,  
маршрут  $X_i'', Y_i''$



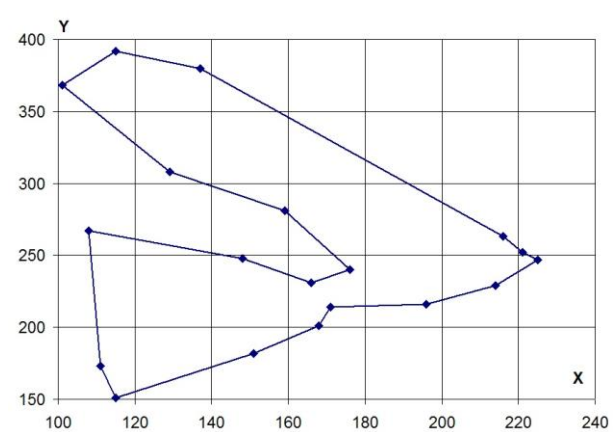
в) – друга ітерація видалення пересічень,  
маршрут  $X_i''', Y_i'''$



г) – третя ітерація видалення пересічень,  
маршрут  $X_i'''', Y_i''''$



д) – четверта ітерація видалення пересічень,  
маршрут  $X_i''''', Y_i'''''$



е) – п'ята ітерація видалення пересічень,  
маршрут  $X_i'''''', Y_i''''''$

Рис. 4 – Послідовність виключення пересічень маршруту ((а) → ... → (е))

## 5 Результати експерименту

В рамках декількох циклів моделюючих досліджень, було проведено 78400 «помилкових» випробувань та 1372 «істинних» (це повний перебір). За їх результатами було проведено розрахунок для десяти найближчих точок кожної з мінучій. Крім того була використана декомпозиція, в результаті чого для кожної мінучій побудовано три контури з найближчих мінучій (див. Табл. 4), одна мінучія представлена рядком з трьох чисел.

Таблиця 4 – Значення метрики

406.000	422.000	484.000
424.000	448.000	495.000
400.000	470.000	466.000
456.000	534.000	511.000
385.000	434.000	416.000
371.000	412.000	416.000
364.000	334.000	353.000
450.000	449.000	449.000
314.000	334.000	346.000
381.000	388.000	431.000
406.000	396.000	384.000

Нижче, на рис. 5-6 зображені графіки розподілів по метриці декомпозиції околиць (оточення) мінуцій.

У таблиці 5 представлені значення порівняння шаблонів для кількох «помилкових» і «дійсних» (істинних) випробувань.

На рисунках 7 та 8 наведено залежність *FAR/FRR* (де, *False Acceptance Rate* - рівень помилкового прийняття (червоний), а *False Rejection Rate* - рівень помилкової відмови (синій)), за якими можна визначити, що  $EER \approx 33\%$  (*Equal Error Rate* – величина еквівалентної помилки).

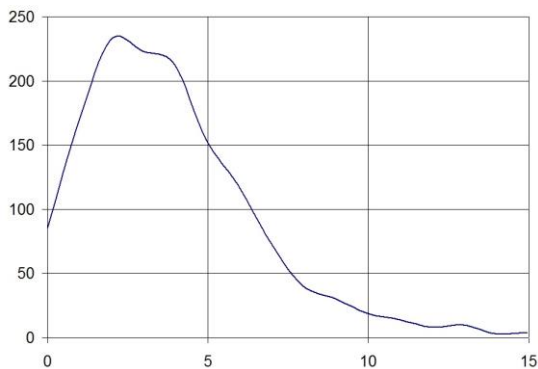


Рис. 5 – Гістограма розподілу істинних випробувань

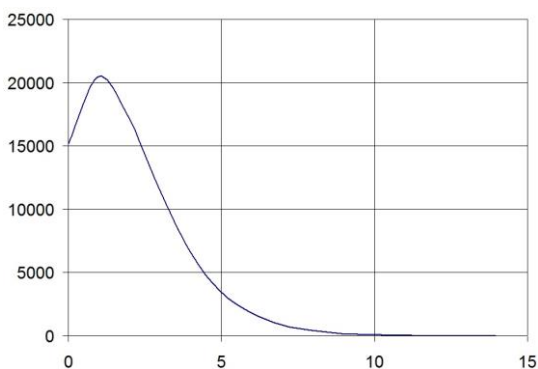
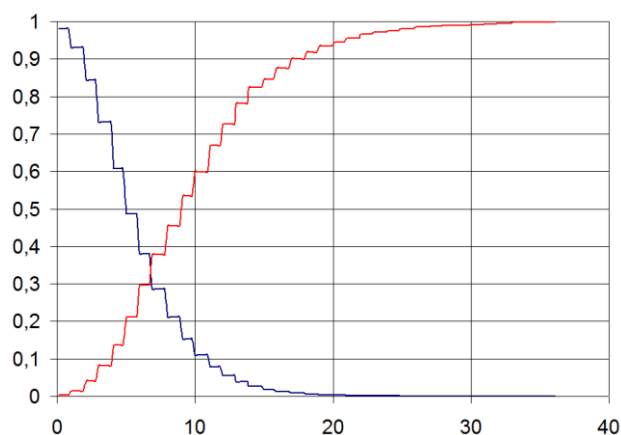
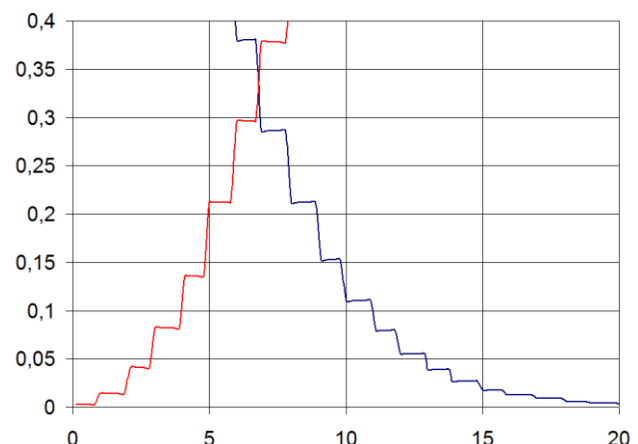


Рис. 6 – Гістограма розподілу помилкових випробувань

Рис. 7 – Залежності *FAR* і *FRR*

Таблиця 5 – Результати порівняння шаблонів для помилкових і істинних випробувань

Значення метрики для помилкових випробувань			Значення метрики для істинних випробувань		
0_0	1_2	1.0	0_3	0_4	11.0
0_0	1_3	3.0	0_3	0_5	9.0
0_0	1_4	2.0	0_3	0_6	2.0
0_0	1_5	1.0	0_3	0_7	6.0
0_0	1_6	1.0	0_4	0_5	4.0
0_0	1_7	0.0	0_4	0_6	13.0
0_0	2_0	2.0	0_4	0_7	8.0
0_0	2_1	1.0	0_5	0_6	6.0
0_0	2_2	0.0	0_5	0_7	4.0
0_0	2_3	1.0	0_6	0_7	7.0
0_0	2_4	2.0	1_0	1_1	2.0
0_0	2_5	1.0	1_0	1_2	2.0
0_0	2_6	0.0	1_0	1_3	0.0
0_0	2_7	0.0	1_0	1_4	5.0

Рис. 8 – Залежності *FAR* і *FRR* ( $EER = 33\%$ )

## 6 Висновки

В роботі розглянута актуальна на даний момент тема – рішення задачі верифікації відбитків *методом знаходження найменшої найкоротшої відстані*, який дозволяє синтезувати локальні структури для мінуцій, які входять в шаблон.

За результатами моделювання можна стверджувати, що даний метод придатний для вирішення завдання верифікації.

До переваг даного методу слід віднести:

- відносну простоту реалізації;

- швидкість обробки бази даних. В ході експериментів час на обробку всієї бази даних становило 42 с. У порівнянні з циліндричним кодом, повний перебір тієї ж бази, займає 30 хвилин;

- повний перебір при постановці завдання верифікації становить 2 хвилини, тоді як при обробці циліндричного коду це може зайняти 24 години.

До недоліків описаного методу слід віднести низьку точність отриманих результатів ( $EER = 33\%$ ). Однак, в зв'язку з цим, справедливо буде зауважити, що на отримання оптимальних умов для тестування програми необхідні серйозні витрати часу і ресурсів.

## Посилання

- [1] Jin Zhe, Andrew Teoh Beng Jin, Fingerprint template protection with Minutia Vicinity Decomposition. Article, 2011. [https://www.researchgate.net/publication/261431544\\_Fingerprint\\_template\\_protection\\_with\\_Minutia\\_Vicinity\\_Decomposition](https://www.researchgate.net/publication/261431544_Fingerprint_template_protection_with_Minutia_Vicinity_Decomposition) - 20.06.2020.
- [2] Wajih Ullah Baig, Umar Munir, Waqas Ellahi, Adeel Ejaz, Kashif Sardar Minutia texture cylinder codes for fingerprint matching <https://arxiv.org/pdf/1807.02251.pdf> , Article 2018 - 20.06.2020.
- [3] Melkozerova, O., Shlokin, V., Malakhov, S. Mathematical model of the biometric system of fingerprint authentication. Problems of informatization: abstracts of the reports of the seventh international conference on November 13-15, 2019, Pages. 92.
- [4] Мудров В.И. Задача о коммивояжере. Издательство «Знание» Москва 1969, 61с.
- [5] Melkozerova, O., Rassomakhin, S. Identification of fingers on the basis of Hamiltonian cycles of local features. the Bulletin of KNU Series "Mathematical Modeling. IT. ACS". Bulletin of V. Karazin Kharkiv National University series «Mathematical Modelling. Information Technology. Automated Control Systems». 2019. Issue 44. Pages 51–65. <https://periodicals.karazin.ua/mia/article/view/15767> - 20.06.2020.

**Reviewer:** Vyacheslav Kalashnikov, Doctor of Sciences (Physics and Mathematics), Full Prof., Department of Systems and Industrial Engineering, Campus Monterrey, Monterrey, Mexico.

E-mail: [kalash@itesm.mx](mailto:kalash@itesm.mx)

Received on July 2020.

### Authors:

Olha Melkozerova, Ph.D., Senior Lecturer, Department of Security of Information Systems and Technologies, V. N. Karazin Kharkiv National University, Kharkiv, Ukraine.

E-mail: [olha.melkozerova@karazin.ua](mailto:olha.melkozerova@karazin.ua)

Serhii Malakhov, Ph.D., Senior Researcher, Associate Professor, Department of Security of Information Systems and Technologies, V. N. Karazin Kharkiv National University, Kharkiv, Ukraine.

E-mail: [mailgate@meta.ua](mailto:mailgate@meta.ua)

Valeriia Haikova, CSD Student, V.N. Karazin Kharkiv National University, Ukraine.

E-mail: [valeriagaikova98@gmail.com](mailto:valeriagaikova98@gmail.com)

### Fingerprint verification using the traveling salesman problem solution and decomposition of the vicinity of the minutiae.

**Abstract.** The article presents an example of verification of the fingerprint database by the method of solving the problem of a salesman using the decomposition of the neighborhood of the nearest minutes. The solution of this problem is resistant to linear, angular deformations, mixing of points. This method provides the correct solution for a small number of points, for a large number of points there is a cross section of the contours, the solution is not optimal. Therefore, to reduce the processing time and calculate the metric, a modified algorithm for solving the problem by the method of branches and boundaries, namely the alignment and exclusion of arcs on each cycle of the optimal route. Verification is based on the creation of local structures for each minute of the imprint, because it is the local structures that are resistant to deformation. Building global structures very often does not lead to good quality indicators, as there is a problem with the centering of the entire sample. A complete list of tests of fingerprint database templates during their verification by this method has been carried out. The use of decomposition of characteristic features provides greater stability when adding false and erasing true minutes. The results of the article show the values of pairwise comparisons of two templates for true and false tests. The indicators of false rejection rate (FRR), false access rate (FAR), single equivalent error rate (EER) were studied.

**Keywords:** Fingerprints; Minutia; The Shortest Distance; Local Signs; The Optimal Route.

**Рецензент:** Вячеслав Калашников, д.ф.-м.н., проф., Технологический университет Монтеррея, Монтеррей, Мексика.

E-mail: [kalash@itesm.mx](mailto:kalash@itesm.mx)

Поступила: Июль 2020.

**Авторы:**

Ольга Мелкозерова, к.т.н., доцент кафедры Безопасности информационных систем и технологий, Харьковский национальный университет имени В. Н. Каразина, Харьков, Украина.

E-mail: [olha.melkozerova@karazin.ua](mailto:olha.melkozerova@karazin.ua)

Сергей Малахов, к.т.н., с.н.с., доцент кафедры Безопасности информационных систем и технологий, Харьковский национальный университет имени В. Н. Каразина, Харьков, Украина.

E-mail: [mailgate@meta.ua](mailto:mailgate@meta.ua)

Валерия Гайкова, студентка факультета компьютерных наук, Харьковский национальный университет имени В. Н. Каразина, Харьков, Украина.

E-mail: [valeriagaikova98@gmail.com](mailto:valeriagaikova98@gmail.com)

**Верификация отпечатков пальцев с использованием решения задачи коммивояжера и декомпозиции окрестностей минуций.**

**Аннотация:** В статье проведена верификация базы отпечатков методом решения задачи коммивояжера с применением декомпозиции окрестностей близлежащих минуций. Решение данной задачи обладает устойчивостью к линейным, угловым искажениям, перемешиванию точек. Данный метод обеспечивает правильное решение для небольшого числа точек, при большом количестве точек возникают пересечения контуров, решение при этом не является оптимальным. Поэтому для уменьшения времени обработки данных и расчета метрики проведено изменение алгоритма решения задачи методом ветвей и границ, а именно выравнивание контура и исключение дуг на каждом цикле поиска оптимального маршрута. Верификация основывается на создании локальных структур для каждой минуции отпечатка, так как именно локальные структуры обладают устойчивостью к искажениям всякого рода. Построение глобальных структур зачастую не приводит к хорошим показателям точности, так как возникает проблема при центрировании всего образца. Проведен полный перебор испытаний образцов базы данных отпечатков пальцев при их верификации этим методом. Применение декомпозиции характерных признаков обеспечивает большую устойчивость при дописывании ложных и стирании истинных минуций. В результатах статьи приведены значения парных сравнений двух шаблонов для истинных и ложных испытаний. Исследованы показатели ложного отказа (*FRR – false rejection rate*), ложного доступа (*FAR – false acceptance rate*), единой эквивалентной ошибки (*EER – equal error rate*).

**Ключевые слова:** отпечатки пальцев; минуции; кратчайшее расстояние; локальные признаки; оптимальный маршрут.