

Міністерство освіти і науки України
Національний університет «Одеська політехніка»
Інститут інформаційної безпеки, радіоелектроніки та телекомунікацій
Кафедра кібербезпеки та програмного забезпечення

Кирилюк Владислав Олександрович,
студент групи РЗ-171

КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА

Удосконалення універсального методу виявлення порушення цілісності
цифрового зображення

Спеціальність:
125 Кібербезпека

Спеціалізація, освітня програма:
Кібербезпека

Керівник:
Кобозєва Алла Анатоліївна,
д.т.н., професор

Одеса – 2022

Національний університет «Одеська політехніка»
Інститут інформаційної безпеки, радіоелектроніки та телекомунікацій
Кафедра кібербезпеки та програмного забезпечення

Рівень вищої освіти другий (магістерський)
Спеціальність 125 Кібербезпека
Спеціалізація, освітня програма Кібербезпека

ЗАТВЕРДЖУЮ
Завідувач кафедри КБПЗ

д.т.н., проф. А.А.Кобозєва
_____ 202_р.

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

Кирилюку Владиславу Олександровичу

1.Тема роботи: *Удосконалення універсального методу виявлення порушення цілісності цифрового зображення,*

керівник роботи *Кобозєва Алла Анатоліївна, д.т.н., проф.*

затверджені наказом ректора від „_____” _____ 20__ р. №_____ .

2.Зміст роботи: *аналіз джерел з даної теми, удосконалення універсального методу виявлення порушення цілісності цифрового зображення, дослідження ефективності удосконаленого методу, порівняння з оригінальним методом, створення користувацького інтерфейсу для демонстрації роботи удосконаленого універсального методу.*

3. Перелік ілюстративного матеріалу: *таблиці результатів тестування розробленого методу, зображення створеного інтерфейсу, слайди презентації*

4. Консультанти розділів роботи

| Розділ | Прізвище, ініціали та посада консультанта | Підпис, дата | |
|--------|---|----------------|------------------|
| | | Завдання видав | Завдання прийняв |
| | | | |

5. Дата видачі завдання “ _____ ” _____ 2022 р.

КАЛЕНДАРНИЙ ПЛАН

| № з/п | Назва етапів кваліфікаційної роботи | Строк виконання | Примітка |
|-------|--|-----------------|-----------------|
| 1 | <i>Аналіз джерел з теми випускної кваліфікаційної роботи</i> | 29.08.2022 | <i>виконано</i> |
| 2 | <i>Обробка отриманих даних</i> | 05.09.2022 | <i>виконано</i> |
| 3 | <i>Удосконалення універсального методу виявлення порушення цілісності цифрового зображення</i> | 07.10.2022 | <i>виконано</i> |
| 4 | <i>Аналіз та вибір мови програмування для реалізації розробленого методу</i> | 14.10.2022 | <i>виконано</i> |
| 5 | <i>Дослідження ефективності удосконаленого методу, порівняння з оригіналом</i> | 04.11.2022 | <i>виконано</i> |
| 6 | <i>Створення користувацького інтерфейсу для демонстрації роботи методу</i> | 11.11.2022 | <i>виконано</i> |
| 7 | <i>Підготовка тексту роботи</i> | 18.11.2022 | |
| 8 | <i>Підготовка презентації та доповіді</i> | 25.11.2022 | <i>виконано</i> |
| 9 | <i>Попередній захист</i> | 02.12.2022 | <i>виконано</i> |
| 10 | <i>Нормоконтроль, рецензування</i> | 15.12.2022 | <i>виконано</i> |
| 11 | <i>Перевірка на антиплагіат</i> | 16.12.2022 | <i>виконано</i> |
| 12 | <i>Занесення роботи в електронний архів</i> | 19.12.2022 | <i>виконано</i> |
| 13 | <i>Допуск до захисту</i> | 20.12.2022 | <i>виконано</i> |

Здобувач вищої освіти _____

Кирилюк В. О.

Керівник роботи _____

Кобозєва А.А.

АНОТАЦІЯ

Кваліфікаційна робота на тему «Удосконалення універсального методу виявлення порушення цілісності цифрового зображення» на здобуття другого (магістерського) рівня вищої освіти за спеціальністю 125 Кібербезпека, освітня програма: Кібербезпека, містить 12 рисунків в основному тексті, 3 таблиці, 1 додаток, 32 літературних джерел за переліком посилань. Робота виконана на 63 сторінках загального тексту і 46 сторінках основного тексту.

Метою роботи є підвищення ефективності виявлення порушень цілісності цифрового зображення шляхом удосконалення існуючого універсального експертного методу.

У результаті виконання кваліфікаційної роботи запропоновано удосконалення універсального методу виявлення порушення цілісності зображення, теоретичний базис якого включає обґрунтування способу симетризації матриці блоку. Удосконалений метод заснований на аналізі власних значень та власних векторів симетричних блоків матриці зображення, що ставляться у відповідність оригінальним блокам. Результати порівняння запропонованого методу з прототипом показали підвищення ефективності експертизи цілісності у сенсі: обчислювальних (часових) витрат на експертизу одного зображення – більше ніж на 15%; точності виявлення порушення цілісності зображення в середньому – більше ніж на 3.5%.

Результати роботи можуть бути використані для подальшої модернізації удосконаленого методу шляхом знаходження для нього ідеальних параметрів перевірки порушення цілісності цифрового зображення.

ЦИФРОВЕ ЗОБРАЖЕННЯ, ПОРУШЕННЯ ЦІЛІСНОСТІ, ВЛАСНИЙ ВЕКТОР, ВЛАСНЕ ЗНАЧЕННЯ.

ABSTRACT

Qualification work on the topic "Improvement of the universal method of detection the integrity violation of a digital image" to obtain the second (master's) level of higher education in the specialty 125 Cybersecurity, educational program: Cybersecurity, contains 12 figures in the main text, 3 tables, 1 appendix, 32 literature sources on the list of references. The work is performed on 63 pages of general text and 46 pages of main text.

The aim of the work is to increase the efficiency of detecting violations of the integrity of a digital image by improving the existing universal expert method.

As a result of the qualification work, an improvement of the universal method of detecting a violation of the integrity of the image is proposed, the theoretical basis of which includes the justification of the method of symmetrization of the block's matrix. The improved method is founded from the analysis of eigen values and eigen vectors of symmetric blocks of image matrix, which are placed in correspondence to the original blocks. The results of the comparison of the proposed method with the prototype showed an increase in the effectiveness of expertise integrity in the sense of: computing (time) costs for examination of one image - by more than 15%; the accuracy of detecting a violation of image integrity on average - by more than 3.5%.

The results of the work can be used for further modernization of the improved method by finding ideal parameters for checking the integrity of a digital image.

DIGITAL IMAGE, INTEGRITY VIOLATION, EIGEN VECTOR, EIGEN VALUE.

ЗМІСТ

| | |
|--|--|
| ВСТУП..... | 7 |
| 1. АНАЛІЗ СУЧАСНОГО СТАНУ ЗАДАЧІ ВИЯВЛЕННЯ ПОРУШЕНЬ ЦІЛІСНОСТІ ЦИФРОВОГО ЗОБРАЖЕННЯ | 10 |
| 1.1. Методи виявлення порушення цілісності ЦЗ | 10 |
| 1.2. Спрямовані методи детектування порушення цілісності цифрового зображення..... | 12 |
| 1.3. Універсальні методи виявлення порушення цілісності цифрового зображення..... | 14 |
| 2. УДОСКОНАЛЕННЯ УНІВЕРСАЛЬНОГО МЕТОДУ ВИЯВЛЕННЯ ПОРУШЕННЯ ЦІЛІСНОСТІ ЦИФРОВОГО ЗОБРАЖЕННЯ | 17 |
| 2.1. Теоретичні основи удосконалення універсального методу виявлення порушення цілісності цифрового зображення..... | 17 |
| 2.1.1. Обґрунтування способу симетризації матриці блоку цифрового зображення | 17 |
| 2.1.2. Обґрунтування можливості використання симетризованих блоків матриці цифрового зображення для експертизи його цілісності | 20 |
| 2.2. Удосконалений метод експертизи цілісності цифрового зображення..... | 25 |
| 2.3. Аналіз ефективності удосконаленого універсального методу | 26 |
| 3. РОЗРОБКА ПРОГРАМНОЇ РЕАЛІЗАЦІЇ УДОСКОНАЛЕНОГО УНІВЕРСАЛЬНОГО МЕТОДУ ВИЯВЛЕННЯ ПОРУШЕННЯ ЦІЛІСНОСТІ ЦИФРОВОГО ЗОБРАЖЕННЯ | 33 |
| 3.1. Вибір мови програмування..... | 33 |
| 3.2. Особливості реалізації програмного продукту..... | 34 |
| 3.3. Інтерфейс програми..... | 44 |
| ВИСНОВКИ..... | 52 |
| ПЕРЕЛІК ПОСИЛАНЬ | 53 |
| Додаток А. Лістинг програмного коду..... | Ошибка! Закладка не определена. |

ВСТУП

Стрімкий розвиток інформаційних технологій, проникнення їх у всі сфери людської діяльності привів сучасне суспільство до стану, при якому несанкціоновані зміни інформації – порушення її цілісності можуть привести до негативних наслідків як для окремих людей, підприємств, банків, фірм, так і до катастроф для людства в цілому, якщо несанкціоновані зміни відбудуться з інформацією, що становить державну таємницю, містить дані зі сфери військової галузі, атомної енергетики, хімічної промисловості тощо, що може поставити під загрозу життя людей в усьому світі [2] і що є критично актуальним для нашої держави сьогодні, під час повномасштабного вторгнення Росії в Україну.

Питання виявлення порушень цілісності інформації – одного з критеріїв її захищеності, зокрема цифрових зображень (ЦЗ), що є її поширеним представленням, стає сьогодні одним з основних для фахівців в області інформаційної та кібербезпеки [3,4]. Ці порушення можуть проводитися різними способами, мати різні цілі. Так організація прихованого (стеганографічного) каналу зв'язку, де порушення цілісності контейнера є результатом вбудови в нього додаткової інформації, може сприяти безпосередньо витоку секретної інформації, привести до матеріального, репутаційного збитку підприємств, фірм, банків, до наслідків державного масштабу [5]; застосування засобів графічних редакторів (AdobePhotoshop, Gimp та ін.) дозволяє, навіть не маючи спеціальної кваліфікації, обробляти, змінювати ЦЗ, цифрові відео, результатом чого може стати, зокрема усунення зі сцени ЦЗ (кадрів відео) окремих предметів, персонажів чи їх штучне додавання, що кардинально змінить зміст цифрового контенту та наслідки від його використання [6], може виявитися критичним при застосуванні таких контентів в судових справах, засобах масової інформації тощо. Тільки своєчасне виявлення неоригінальності цифрового контенту дозволить тут уникнути негативних наслідків, що говорить про актуальність і важливість розробки, модифікації, удосконалення відповідних методів, спрямованих на виявлення порушення цілісності інформаційного контенту, зокрема ЦЗ.

Метою роботи є підвищення ефективності виявлення порушень цілісності ЦЗ шляхом удосконалення універсального експертного методу, запропонованого в [7].

Для досягнення поставленої мети необхідно вирішити наступні **задачі**:

- а) Обґрунтувати спосіб симетризації матриці блоку ЦЗ, що дасть можливість заміни формальних параметрів блоку (СНЧ, СНВ), що використовуються в процесі експертизи в [7], на власні вектори і власні значення, отримання яких є менш обчислювально затратним;
- б) Обґрунтувати математично можливість використання симетризованих блоків матриці ЦЗ для експертизи його цілісності;
- в) Розробити удосконалення методу [7] та його алгоритмічну реалізацію;
- г) Провести оцінку ефективності запропонованої алгоритмічної реалізації, зокрема, порівняльну з аналогом-прототипом;
- д) Створити програмну реалізацію удосконаленого методу.

Об'єкт дослідження – процеси виявлення порушення цілісності ЦЗ.

Предмет дослідження – універсальні методи виявлення порушення цілісності ЦЗ.

Методи дослідження. При формуванні теоретичного базису удосконаленого універсального методу виявлення порушення цілісності ЦЗ використано: матричний аналіз, обчислювальна лінійна алгебра, теорія збурень; при формуванні та оцінці ефективності запропонованого удосконалення - обчислювальні методи, методи обробки цифрових зображень, теорія алгоритмів.

Наукова новизна отриманих результатів полягає в наступному:

Удосконалено універсальний метод виявлення порушення цілісності цифрового зображення шляхом обґрунтування можливості симетризації блоків матриці досліджуваного зображення практично без втрати інформації про первісні блоки, вибору способу цієї симетризації і заміни набору досліджуваних при експертизі формальних параметрів на власні вектори і власні значення, що дало можливість підвищити ефективність виявлення порушень цілісності ЦЗ.

Практичне значення отриманих результатів. Практична цінність роботи полягає в доведенні здобувачем отриманих результатів до конкретного методу та алгоритму, що може бути використаним як складова систем захисту інформації, інформаційних систем різного наповнення будь-якого закладу, підприємства. Алгоритмічна реалізація удосконаленого універсального методу виявлення порушення цілісності ЦЗ дозволила підвищити ефективність експертизи цілісності ЦЗ у сенсі: обчислювальних (часових) витрат на експертизу одного зображення – більше ніж на 15%; точності виявлення порушення цілісності зображення в середньому – більше ніж на 3.5%.

В додатку А представлений лістинг програмного коду удосконаленого методу.

Публікації. Матеріали магістерської роботи були опубліковані в [1].

1. АНАЛІЗ СУЧАСНОГО СТАНУ ЗАДАЧІ ВИЯВЛЕННЯ ПОРУШЕНЬ ЦІЛІСНОСТІ ЦИФРОВОГО ЗОБРАЖЕННЯ

1.1. Методи виявлення порушення цілісності ЦЗ

Цілісність інформації, зокрема ЦЗ, є одним з критеріїв її захищеності.

Для виявлення наявності порушення цілісності ЦЗ потрібно визначити, які зміни з зображенням відбулися. Якщо не має змоги визначити, як саме цілісність ЦЗ була порушена, потрібно визначити сам факт порушення цілісності ЦЗ. Методи експертизи цілісності ЦЗ розподіляються на дві великі групи: активні і пасивні (або «сліпі») [8,9]. Активні методи потребують інформацію про оригінальне ЦЗ, на відміну від пасивних, для яких така інформація не потрібна.

Існує багато інструментів, за допомогою яких можна вносити зміни до ЦЗ. Маніпуляція може відбуватися за допомогою багатьох операцій обробки, таких як розмиття, масштабування, регулювання яскравості, обертання тощо, а також комбінації даних маніпуляцій.

До активних методів, що виявляють такі маніпуляції слід віднести методи, які використовують електронний цифровий підпис або цифрові водяні знаки.

Цифровий водяний знак (ЦВЗ) - спеціальна мітка, що вбудовується в зображення під час його створення. ЦВЗ бувають видимими й невидимими. Видимі ЦВЗ використовуються на зображенні або відео, і являють собою текст або логотип, який ідентифікує автора. Як правило, використовувані невидимі ЦВЗ, які вбудовуються в зображення, можуть бути виявлені й перевірені тільки спеціальним програмним забезпеченням. ЦВЗ є частиною ЦЗ й змінюються разом з ним, що і дозволяє використовувати їх для визначення цілісності ЦЗ або встановлення її порушення. ЦВЗ можуть містити деякий автентичний код, інформацію про власника, або яку-небудь керуючу інформацію.

Друга категорія, заснована на цифровому підписі (ЦП), використовує методи, які виділяють ознаки (властивості) зображень, такі як колір, форми або текстури; потім отримані дані перетворюються в якийсь код, який шифрується секретним ключем, утворюючи цифровий підпис зображення. При проведенні

експертизи цілісності ЦЗ витягають унікальні властивості досліджуваного зображення й отримують його код. Цифровий підпис, отриманий з оригінального зображення, з використанням відкритого ключа дешифрується й у результаті є код, відповідний до оригінального зображення. Отримані коди досліджуваного зображення й оригінального порівнюються, і робиться висновок про цілісність досліджуваного ЦЗ.

Істотним недоліком активних методів є те, що при використанні ЦВЗ вбудовувана інформація повинна бути занурена в цифровий об'єкт безпосередньо під час створення цього об'єкта, що обмежує область застосування означеного підходу тільки для механізмів генерації ЦЗ, що мають вбудовані можливості занурення ЦВЗ, чого більша частина широко використовуваних відеокамер на сьогоднішній день не має. Крім цього, на формальному рівні вбудова ЦВЗ сама по собі приводить до порушення його цілісності. При використанні ж ЦП необхідно зберігати цифровий підпис еталонного зображення, що також є суттєвим недоліком, оскільки еталонне зображення взагалі може бути відсутнім у експерта під час проведення дослідження.

На сьогоднішній день саме пасивні експертні методи займають провідні позиції для розв'язку задачі, що розглядається [9-11], хоча організація «сліпого» детектування результатів порушення цілісності ЦЗ є більш складною.

Всі пасивні методи, в свою чергу, можна розподілити на:

- спрямовані (налаштовані на конкретні збурні дії, які враховують особливості, властивості тих збурень, що є результатом таких дій);
- універсальні (налаштовані на виявлення наявності відмін досліджуваного контенту від оригінального незалежно від того, яким чином ці зміни були отримані).

Як правило, спрямовані методи мають більшу ефективність проти конкретного типу збурення. Це пояснюється тим, що, маючи одне конкретне збурення, можна його дослідити – та знайти закономірності в зміні ЦЗ. Хоча існує гіпотеза, згідно з якою методи подроби зображення під час фальсифікації ЦЗ не залишають видимого відбитку. Багато існуючих методів локалізують подробилену

область на цифровому зображенні, ідентифікуючи підробку та сліди, залишені методами фальсифікації [10].

1.2. Спрямовані методи детектування порушення цілісності цифрового зображення

В наші дні зусилля вчених найчастіше спрямовані на виявлення результатів конкретних збурних дій: зміни яскравості [12], розмиття ЦЗ чи його частини [13], накладання шуму [14], результатів стеганоперетворення конкретними стеганоалгоритмами [15,16] тощо. Математичний базис таких методів формується з врахуванням особливостей, які вносять саме ці конкретні збурні дії в параметри оригінального ЦЗ.

Так в [12] знайдене формальне представлення результату корекції яскравості ЦЗ у вигляді корекції максимального сингулярного числа σ_1 його матриці яскравості Y , яке і використане в відповідному експертному методі, що між іншим дало змогу визначати клоновані області ЦЗ.

Локалізація області підробки в ЦЗ стає все більш складною при експертизі ЦЗ. У роботі [13], спираючись на CNN (convolution neural network) представлено детектор на основі RFM (reliability fusion map). Результати експериментів, представлені в роботі, доводять ефективність запропонованого в ній методу.

В [14] запропоновано модифікацію методу, спрямованого на виявлення наслідків шуму Перлінні, метою якого є моделювання на ЦЗ реалістичного в природі явища – неба та хмар. Модифікація проведена шляхом зміни математичного базису методу і полягає у використанні дискретної функції швидкості зміни відношення сингулярних чисел блоку матриці ЦЗ, отриманого шляхом її стандартної розбивки, замість існуючої раніше – дискретної функції значень сингулярних чисел блоку.

В [15,16] запропоновані стеганоаналітичні методи, спрямовані на виявлення результатів стеганоперетворення методом модифікації найменшого значущого біта (LSB-методом). Ці методи не тільки налаштовані на конкретний

стеганографічний метод, а ще й, як більшість існуючих аналогів, накладають обмеження на величину пропускнуї спроможності прихованого каналу зв'язку, що піддається експертизі, стаючи неефективними у випадку пропускнуї спроможності менше, ніж 0.1 біт/піксель.

В [17] запропонований метод виявлення результатів штучного підвищення різкості в ЦЗ, розглянутий конкретний фільтр, що використовується для цієї операції в графічному редакторі Adobe Photoshop – «Інтелектуальна різкість». Для відокремлення ЦЗ, що піддалися обробці таким фільтром, від таких, що не піддалися, використовується оцінка відношення кількості близьких пар кольорів до загальної кількості пар кольорів, яка специфічно змінюється при застосуванні саме згаданого фільтра, для якої імперічно визначене порогове значення.

В [18] описується алгоритм для виявлення розмиття, яке можна застосувати до зображення в графічному редакторі Adobe Photoshop фільтром «Motion blur». Розроблений алгоритм базується на сингулярному розкладанні блоків зображення, та подальшому аналізі швидкості росту сингулярних чисел.

В [19] було запропоновано модернізацію методу штучного підвищення різкості в ЦЗ, було розглянуто фільтр, який використовується у графічному редакторі GIMP – «Unsharp Mask». Авторами аналізуються кількості нульових пікселів та пікселів, значення яскравості яких дорівнюють одиниці, для кожної кольорової складової, з погляду на гіпотезу про «обнулення» пікселів, яка знайшла своє підтвердження на практиці. Запропонований модифікований метод виявився ефективним як для зображень з втратами, так і для зображень без втрат.

Існує ціла низка експертних методів, які застосовуються для аналізу ЦЗ у форматі Jpeg, заснованих на виявленні «ефекту подвійного квантування», що виникає в гістограмах частотних коефіцієнтів зображення при первісному та повторному його збереженні в Jpeg, які мають значне поширення в силу широкого використання цього формату сьогодні для збереження ЦЗ [11,20], але очевидно, що «форматоорієнтованість» значно обмежує область застосування таких методів для експертизи цілісності і не дає можливості віднести їх до групи універсальних.

В [20] висувається ідея щодо виявлення результатів вбудови одного ЦЗ в інше за допомогою дискретного вейвлет перетворення. Запропонований метод працює з інформацією про контури зображення, про краї, вбудованого ЦЗ, які не до кінця узгоджені з областю вбудови, а тому піддаються розпізнаванню за допомогою представленого методу. Також даний метод продемонстрував стійкість до операцій масштабування та геометричних параметрів орієнтації.

Таким чином, на сьогодні існують ефективні спрямовані методи виявлення порушень цілісності ЦЗ, але при відсутності у експерта інформації про ймовірні/застосовані збурні дії ці методи можуть виявитися неефективними, даючи висновок про збереження цілісності ЦЗ там, де її порушення відбулося, але використані експертні методи були просто не розраховані на застосовані конкретні збурні дії.

1.3. Універсальні методи виявлення порушення цілісності цифрового зображення

Враховуючи те, що на практиці обізнаність експерта про можливі збурні дії не завжди присутня, наявний у експерта арсенал програмних засобів є обмеженим та принципово не може (у випадку спрямованих методів) забезпечити «готовність» до всіх збурних дій, надзвичайно актуальним на сьогодні є наявність, розробка, удосконалення саме універсальних методів для розв'язку задачі, що розглядається.

І хоча розробки в цьому напрямку ведуться [7,21], при цьому найчастіше – в межах стеганоаналізу [22,23], на сьогоднішній день універсальні методи виявлення порушення цілісності ЦЗ майже відсутні.

Мала їх кількість зумовлена, в першу чергу, значною складністю задачі: при їх розробці потрібно розглядати збурні дії на ЦЗ без врахування їх конкретики, а лише як інструмент, результатом якого є отримання ЦЗ, яке відрізняється від оригінального. Тут потрібно розуміти, що реальні зміни в ЦЗ можуть вноситися як в області перетворення, так і в просторовій, при цьому досліджуване

зображення може зберігатися/створюватися в різних форматах (з втратами, без втрат).

Одним з невеликої кількості існуючих універсальних пасивних методів виявлення порушення цілісності ЦЗ є метод, запропонований в [7]. Основою методу є доведена для більшості $l \times l$ -блоків оригінального ЦЗ, отриманих шляхом стандартної розбивки його матриці, рівність:

$$\angle(u_1, \bar{\sigma}) \approx \angle(v_1, \bar{\sigma}) \approx \angle(n^\circ, e_1), \quad (1.1)$$

де u_1, v_1 – ортонормовані сингулярні вектори (СНВ) блоку, що відповідають найбільшому сингулярному числу σ_1 ; $\bar{\sigma} = \sigma / \|\sigma\|$, де $\sigma = (\sigma_1, \sigma_2, \dots, \sigma_i)^T$, $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_i \geq 0$ – сингулярні числа (СНЧ) блоку; $n^\circ = (1/\sqrt{l}, 1/\sqrt{l}, \dots, 1/\sqrt{l})^T \in R^l$ – n -оптимальний вектор простору R^l , $\angle(u_1, \bar{\sigma})$, $\angle(v_1, \bar{\sigma})$, $\angle(n^\circ, e_1)$ – величини кутів між векторами u_1 і $\bar{\sigma}$, v_1 і $\bar{\sigma}$, n° і вектором стандартного базису $e_1 = (1, 0, \dots, 0)$ простору R^l , що відповідає додатному напрямку осі Ox_1 , відповідно. Порушення цілісності оригінального ЦЗ, зміна його матриці приведе до зменшення кількості блоків, у яких рівність в (1.1) має місце, і, як наслідок, може взагалі привести до порушення співвідношення (1.1) для більшості блоків ЦЗ. Значимою перевагою цього методу є те, що він залишається ефективним, незалежно від конкретики та сили збурної дії, в результаті якої відбувається порушення цілісності ЦЗ, від формату досліджуваного зображення. Але орієнтованість його на аналіз властивостей СНЧ, СНВ блоків, для отримання яких використовується їх нормальні сингулярні розкладання, які є достатньо «дорогими» в обчислювальному сенсі, робить актуальним питання його удосконалення при збереженні всіх переваг, як і метода, запропонованого в [24].

Таким чином, на основі проведеного аналізу наукових джерел встановлено, що проблема виявлення порушень цілісності ЦЗ не є вирішеною остаточно. Абсолютна більшість існуючих пасивних методів, що займають провідні позиції для розв'язку задачі, що розглядається, мають значні недоліки, серед яких: орієнтованість здебільшого на конкретну збурну дію, на формат ЦЗ, величину збурення, що зазнає оригінальний контент в результаті збурної дії, значні часові

витрати, залишаючи актуальною задачу підвищення ефективності процесу виявлення порушень цілісності ЦЗ, зокрема в сенсі універсальної експертизи [1].

2. УДОСКОНАЛЕННЯ УНІВЕРСАЛЬНОГО МЕТОДУ ВИЯВЛЕННЯ ПОРУШЕННЯ ЦІЛІСНОСТІ ЦИФРОВОГО ЗОБРАЖЕННЯ

2.1. Теоретичні основи удосконалення універсального методу виявлення порушення цілісності цифрового зображення

2.1.1. Обґрунтування способу симетризації матриці блоку цифрового зображення

Нехай формальним представлення ЦЗ, яке піддається експертизі цілісності, є $n \times n$ -матриця F . У випадку кольорового зображення ця матриця може відповідати будь-якій кольоровій складовій (схема RGB) чи є матрицею яскравості (схема YUV).

Як показники ефективності експертного методу в роботі розглядаються:

- обчислювальні (часові) витрати на експертизу одного ЦЗ;
- точність виявлення порушення цілісності [25] (*accuracy (ACC)*), яка

визначається відповідно до формули:

$$ACC = (TP + TN) / (TP + FN + TN + FP), \quad (2.1)$$

де:

- TP (*TruePositive*) – число правильно виявлених ЦЗ, цілісність яких була порушена;
- TN (*TrueNegative*) – число правильно виявлених оригінальних ЦЗ;
- FP (*FalsePositive*) – число оригінальних ЦЗ, помилково прийнятих за такі, цілісність яких була порушена;
- FN (*FalseNegative*) – число ЦЗ, цілісність яких була порушена, помилково визнаних оригінальними.

Матриця F стандартним чином [26] розбивається на непересічні $l \times l$ -блоки, довільний з яких позначимо A . Для матриці A , у якій відсутні кратні СНЧ, маємо єдине нормальне сингулярне розкладання [27]:

$$A = U \Sigma V^T = \sum_{i=1}^l \sigma_i u_i v_i^T, \quad (2.2)$$

де U, V – ортогональні $l \times l$ -матриці, стовпці яких $u_i, v_i, i = \overline{1, l}$, є лівими і правими СНВ A відповідно, при цьому ліві СНВ додатково є лексикографічно додатними; $\Sigma = \text{diag}(\sigma_1, \dots, \sigma_l), \sigma_1 \geq \dots \geq \sigma_l \geq 0$ – СНЧ A . Права частина (2.2) представляє сингулярне розкладання A у формі зовнішніх добутоків [28].

Якщо матриця A є симетричною, то всі її власні значення (ВЗ) є дійсними, при цьому для неї можливо побудувати єдине нормальне спектральне розкладання у випадку відсутності ВЗ з однаковими абсолютними значеннями:

$$A = W \Lambda W^T = \sum_{i=1}^l \lambda_i w_i w_i^T,$$

(2.3)

де W – ортогональна $l \times l$ -матриця, стовпці якої $w_i, i = \overline{1, l}$, є лексикографічно додатними власними векторами (ВВ) A , $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_l), \lambda_1, \dots, \lambda_l$ – ВЗ A . Права частина рівності (2.3) представляє спектральне розкладання A у формі зовнішніх добутоків [28].

Спектральне і сингулярне розкладання симетричної матриці пов'язані між собою [28]. При цьому очевидним є те, що обчислювальна складність процесу побудови сингулярного розкладання A , яке передбачає визначення елементів матриць U, V і діагональної матриці Σ і оцінюється як $O(l^3)$, приблизно вдвічі більше, ніж спектрального, де передбачається обчислення елементів лише матриці W і діагональної матриці Λ (таке ж саме співвідношення буде мати місце і для запитів до пам'яті). Таким чином, одним з шляхів удосконалення методу з [7], що очікувано сприятиме зменшенню часових витрат на експертизу ЦЗ, є побудова експертизи не на аналізі СНЧ і СНВ реальних блоків, а на аналізі ВЗ і ВВ для симетричних матриць блоків (якщо цю симетричність можливо буде забезпечити без втрати точності виявлення порушення цілісності АСС та без виникнення обмежень на застосування відповідного методу). Дійсно, хоча обчислювальна складність будь-якого блокового методу, яким є і метод [7], визначається кількістю $l \times l$ -блоків ЦЗ і для $n \times n$ -матриці F становить $C \begin{bmatrix} n \\ l \end{bmatrix} \times \begin{bmatrix} n \\ l \end{bmatrix} = O(n^2)$, де C

не залежить від n , незалежно від того, яка кількість операцій використовується для роботи з одним блоком, але кількість операцій обробки блоку відіб'ється на коефіцієнті при n^2 і при застосуванні спектрального розкладання для матриці блоку цей коефіцієнт очевидно буде менше, чим при використанні сингулярного розкладання. Але в ЦЗ блок, як правило, не є симетричним [1].

Розглянемо декілька можливих способів отримання симетричного виду блоку B , що буде ставитися у співвідношення реальному блоку A ЦЗ при його експертизі:

$$A \rightarrow B = A^T A, \quad (2.4)$$

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1l} \\ a_{21} & a_{22} & \dots & a_{2l} \\ \dots & \dots & \dots & \dots \\ a_{l1} & a_{l2} & \dots & a_{ll} \end{pmatrix} \rightarrow B = \begin{pmatrix} a_{11} & a_{21} & \dots & a_{l1} \\ a_{21} & a_{22} & \dots & a_{l2} \\ \dots & \dots & \dots & \dots \\ a_{l1} & a_{l2} & \dots & a_{ll} \end{pmatrix} \vee B = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1l} \\ a_{12} & a_{22} & \dots & a_{2l} \\ \dots & \dots & \dots & \dots \\ a_{l1} & a_{2l} & \dots & a_{ll} \end{pmatrix}, \quad (2.5)$$

$$A \rightarrow B = \frac{A + A^T}{2}.$$

(2.6)

Обчислювальна складність симетризації (2.4) визначається як $O(l^3)$, а для (2.6) – $O(l^2)$. Використання одної з матриць B , що визначаються (2.5), взагалі не вимагає проведення жодної арифметичної операції для перерахування елементів B відносно A , а може бути сформована лише за $O(l^2)$ операцій присвоювання, виконання яких вимагає меншого часу, ніж будь-яка арифметична операція. На перший погляд, з точки зору часових витрат, перевагу треба віддати способу (2.5), але для кожного з варіантів (2.5) матриця B несе в собі інформацію лише про нижній/верхній трикутник оригінальної матриці A , при цьому інформація про верхній/нижній трикутник губиться, а загалом губиться майже половина інформації про досліджуване ЦЗ, що є неприпустимим з урахуванням специфіки задачі, що розглядається в роботі. Передбаченим результатом цього, який підтверджено практично, є, на фоні зменшення часу експертизи, значне зменшення ефективності експертизи (ACC) в порівнянні з [7], що, з урахуванням вищенаведеного, робить пріоритетним спосіб (2.6) для симетризації блоку, що і

використовується нижче, який хоча дещо і спотворює інформацію про реальні значення матриці A , але зберігає її в цілому про блок A .

2.1.2. Обґрунтування можливості використання симетризованих блоків матриці цифрового зображення для експертизи його цілісності

Як доведено в [29], для СНВ u_1, v_1 блоку оригінального ЦЗ, що відповідають максимальному СНЧ σ_1 :

$$u_1 \approx n^\circ, \quad v_1 \approx n^\circ. \quad (2.7)$$

Враховуючи форму (2.2) сингулярного розкладання, з (2.6) для симетричної матриці B маємо:

$$\begin{aligned} B &= \frac{A+A^T}{2} = \frac{U\Sigma V^T + V\Sigma U^T}{2} = \frac{1}{2} \left(\sum_{i=1}^l \sigma_i u_i v_i^T + \sum_{i=1}^l \sigma_i v_i u_i^T \right) = \frac{1}{2} \sum_{i=1}^l \sigma_i (u_i v_i^T + v_i u_i^T) = \\ &= \frac{1}{2} \left(\sigma_1 (u_1 v_1^T + v_1 u_1^T) + \sum_{i=2}^l \sigma_i (u_i v_i^T + v_i u_i^T) \right). \end{aligned} \quad (2.8)$$

Підставимо (2.7) в (2.8):

$$B = \frac{1}{2} \left(\sigma_1 \left(n^\circ (n^\circ)^T + n^\circ (n^\circ)^T \right) + \sum_{i=2}^l \sigma_i (u_i v_i^T + v_i u_i^T) \right) = \sigma_1 n^\circ (n^\circ)^T + \frac{1}{2} \sum_{i=2}^l \sigma_i (u_i v_i^T + v_i u_i^T). \quad (2.9)$$

Для симетричної матриці B існує певний зв'язок між її сингулярним і спектральним розкладанням [28]. Якщо у відповідності до (2.3) позначити спектральне розкладання B : $B = \bar{U} \bar{\Lambda} \bar{U}^T$ ($\bar{\Lambda} = \text{diag}(\bar{\lambda}_1, \bar{\lambda}_2, \dots, \bar{\lambda}_l)$ – матриця ВЗ, $\bar{U} = (\bar{u}_1, \bar{u}_2, \dots, \bar{u}_l)$ – ортогональна матриця ВВ), то сингулярне розкладання для B виглядає: $B = \bar{U} \bar{\Sigma} \bar{V}^T$ ($\bar{\Sigma} = \text{diag}(\bar{\sigma}_1, \bar{\sigma}_2, \dots, \bar{\sigma}_n)$ – матриця СНЧ, $\bar{V} = (\bar{v}_1, \bar{v}_2, \dots, \bar{v}_l)$ – ортогональна матриця правих СНВ), де

$$\bar{\sigma}_i = |\bar{\lambda}_i|, \quad \bar{v}_i = \text{sign}(\bar{\lambda}_i) \bar{u}_i. \quad (2.10)$$

З урахуванням (2.7) для B при отриманні її сингулярного розкладання: $\bar{u}_1 \approx n^\circ, \quad \bar{v}_1 \approx n^\circ$, це означає, що для власного вектора B , що отримується при спектральному розкладанні B і відповідає максимальному за модулем ВЗ, яке за теоремою Фробеніуса [30] є додатним: $\bar{u}_1 \approx n^\circ$.

Якщо б для B будувалось спектральне розкладання у формі зовнішніх добутків, то ми б отримали:

$$B = \sum_{i=1}^l \bar{\lambda}_i \bar{u}_i (\bar{u}_i)^T = \bar{\lambda}_1 \bar{u}_1 (\bar{u}_1)^T + \sum_{i=2}^l \bar{\lambda}_i \bar{u}_i (\bar{u}_i)^T = \bar{\lambda}_1 n^\circ (n^\circ)^T + \sum_{i=2}^l \bar{\lambda}_i \bar{u}_i (\bar{u}_i)^T. \quad (2.11)$$

А якщо б для B будувалось сингулярне розкладання у формі зовнішніх добутків, воно б мало вигляд:

$$B = \sum_{i=1}^l \bar{\sigma}_i \bar{u}_i (\bar{v}_i)^T = \bar{\sigma}_1 \bar{u}_1 (\bar{v}_1)^T + \sum_{i=2}^l \bar{\sigma}_i \bar{u}_i (\bar{v}_i)^T = \bar{\sigma}_1 n^\circ (n^\circ)^T + \sum_{i=2}^l \bar{\sigma}_i \bar{u}_i (\bar{v}_i)^T. \quad (2.12)$$

Враховуючи ортогональність ВВ, СНВ B , можна стверджувати, що серед власних векторів, як і серед лівих і правих СНВ B є тільки по одному, що дорівнюють n° . Це вектори, що відповідають найбільшому ВЗ/найбільшому СНЧ. Таким чином, перший доданок в правій частині (2.9) можна розглядати, як добуток максимального власного значення на відповідний власний вектор. Порівнюючи праві частини (2.9), (2.11) і (2.12), враховуючи (2.10), маємо:

$$\bar{\lambda}_1 = \bar{\sigma}_1 = \sigma_1. \quad (2.13)$$

Основою співвідношення (1.1) разом з (2.7) було в [29] співвідношення $\bar{\sigma} \approx e_1$, отримане з урахуванням того, що для блоків ЦЗ максимальне СНЧ є набагато більшим за всі інші СНЧ. Операція (2.6), усереднюючи значення яскравості пікселів, розмиває блок, зменшуючи його високочастотну складову. Для СНЧ, враховуючи зв'язок між сингулярними тройками (блоку) матриці ЦЗ та її частотними коефіцієнтами, що полягає в тому, що сингулярні тройки, що відповідають максимальним/мінімальним/середнім СНЧ несуть в собі інформацію, головним чином, про низькочастотну/високочастотну/середньочастотну складову сигналу, це приводить до сукупного зменшення найменших і, можливо, середніх значень СНЧ, що, враховуючи (2.10), (2.13), приведе до того, що $\bar{\lambda}_1$ буде мати більшу абсолютну відокремленість

$$gap_{abs}(1, B) = \min_{i \neq 1} \left| \bar{\lambda}_1 - \bar{\lambda}_i \right|,$$

(2.14)

ніж відокремленість $svdgap(1, A) = \min_{i \neq 1} |\sigma_1 - \sigma_i|$ СНЧ σ_1 в A .

Наслідком цього, в свою чергу, буде те, що нормований вектор модулів власних значень B

$$\bar{\lambda} = \frac{\lambda}{\|\lambda\|}, \quad (2.15)$$

де $\lambda = (\bar{\lambda}_1, \bar{\lambda}_2, \dots, \bar{\lambda}_n)^T$, буде ближче до e_1 , чим нормований вектор $\bar{\sigma}$ СНЧ A , тобто:

$$\angle(\bar{\lambda}, e_1) < \angle(\bar{\sigma}, e_1).$$

(2.16)

Співвідношення (2.16) приведе до того, що для матриці F ЦЗ буде більше блоків у вигляді B , для яких

$$\angle(\bar{u}_1, \bar{\lambda}) \approx \angle(n^\circ, e_1), \quad (2.17)$$

де \bar{u}_1 для матриці $B \in \mathbb{V}\mathbb{V}$, який відповідає максимальному власному значенню $\bar{\lambda}_1$, ніж блоків оригінальних A , для яких має місце (1.1).

Таким чином доведено:

Твердження 1. Для більшості блоків у симетричному вигляді (2.6), що ставляться у відповідність блокам оригінального ЦЗ, має місце співвідношення (2.17), при цьому (2.17) виконується для більшої кількості блоків, ніж співвідношення (1.1) для оригінальних блоків зображення.

Поняття «більшості блоків ЦЗ» на практиці візуалізується модою гістограми, що далі позначається Γ_λ , значень кутів $\angle(\bar{u}_1, \bar{\lambda})$ $l \times l$ -блоків цього зображення, яка для оригінального ЦЗ, дорівнює значенню кута $\angle(n^\circ, e_1)$ у відповідному просторі R^l . З доказу твердження 1 випливає, що при порушенні цілісності ЦЗ мода Γ_λ може зсуватися з положення $\angle(n^\circ, e_1)$, що буде, разом з іншими характеристиками, вказівкою на неавторизовану зміну зображення. Крім зсуву моди Γ_λ на порушення цілісності ЦЗ буде вказувати зміна характеру

гістограми: в результаті збурної дії, навіть якщо мода Γ_λ і залишиться в $\angle(n^\circ, e_1)$, значення в моді значно зменшиться, значна кількість блоків, що в оригінальному ЦЗ робили свій внесок в стовпець Γ_λ , що відповідає моді, в збуреному ЦЗ зробить внесок в інші стовпці [1].

Оскільки відповідно до загальної формули для довільної симетричної матриці M :

$$\max_j |\lambda_j(M) - \lambda_j(M + \Delta M)| \leq \|\Delta M\|_2,$$

(2.18)

де $M + \Delta M$ – збурена матриця, ΔM – матриця збурення, $\|\cdot\|_2$ – спектральна матрична норма [28], всі ВЗ симетричної $n \times n$ -матриці M є добре обумовленими, таким же буде і вектор $\bar{\lambda}$ (2.15) симетричного блоку B (2.6), що ставиться у відповідність блоку A ЦЗ. Чутливість (обумовленість) ВВ \bar{u}_1 симетричної довільної M визначається формулою:

$$\sin \theta_1 \leq \frac{2\|\Delta M\|_2}{\text{gap}_{abs}(1, M)}, \quad (2.19)$$

де θ_1 – гострий кут між ВВ, що відповідають максимальним ВЗ в матрицях M і $M + \Delta M$. Оскільки абсолютна відокремленість (2.14) максимального ВЗ для ЦЗ завжди є значною, такою, що набагато перевищує абсолютні відокремленості інших ВЗ, з (2.19) впливає нечутливість (добра обумовленість) \bar{u}_1 для блоку B (2.6). Виходячи з (2.18), (2.19), можна стверджувати, що для тих блоків, що в оригінальному ЦЗ робили свій внесок в стовпець Γ_λ , який відповідає моді $\angle(n^\circ, e_1)$, і для яких в результаті збурної дії значення кута $\angle(\bar{u}_1, \bar{\lambda})$ змінилося, ця зміна не може бути значною, залишаючи внесок таких блоків для збуреного ЦЗ в стовпці Γ_λ , що знаходяться в деякому незначному околі моди. Визначені зміни Γ_λ при змінах зображення будуть тим більше, чим більше буде величина збурної дії, якій піддалося ЦЗ.

Отримані теоретичні висновки знайшли своє підтвердження на практиці, ілюстрація чого для конкретного ЦЗ наведена на рис.2.1., де очевидним є збільшення значення гистограми Γ_λ в моді для випадку симетризованих блоків ЦЗ, яка відповідає значенню кута $\angle(n^\circ, e_1)$, що для 4×4 -блоків дорівнює 60 градусів (рис.2.1(б)), в порівнянні зі значенням в моді гистограми кутів між нормованим вектором СНЧ і лівим СНВ, що відповідає максимальному СНЧ, оригінальних блоків оригінального ЦЗ (рис.2.1(а)), а також видозміна Γ_λ при порушенні цілісності ЦЗ (рис.2.1(в,г)).

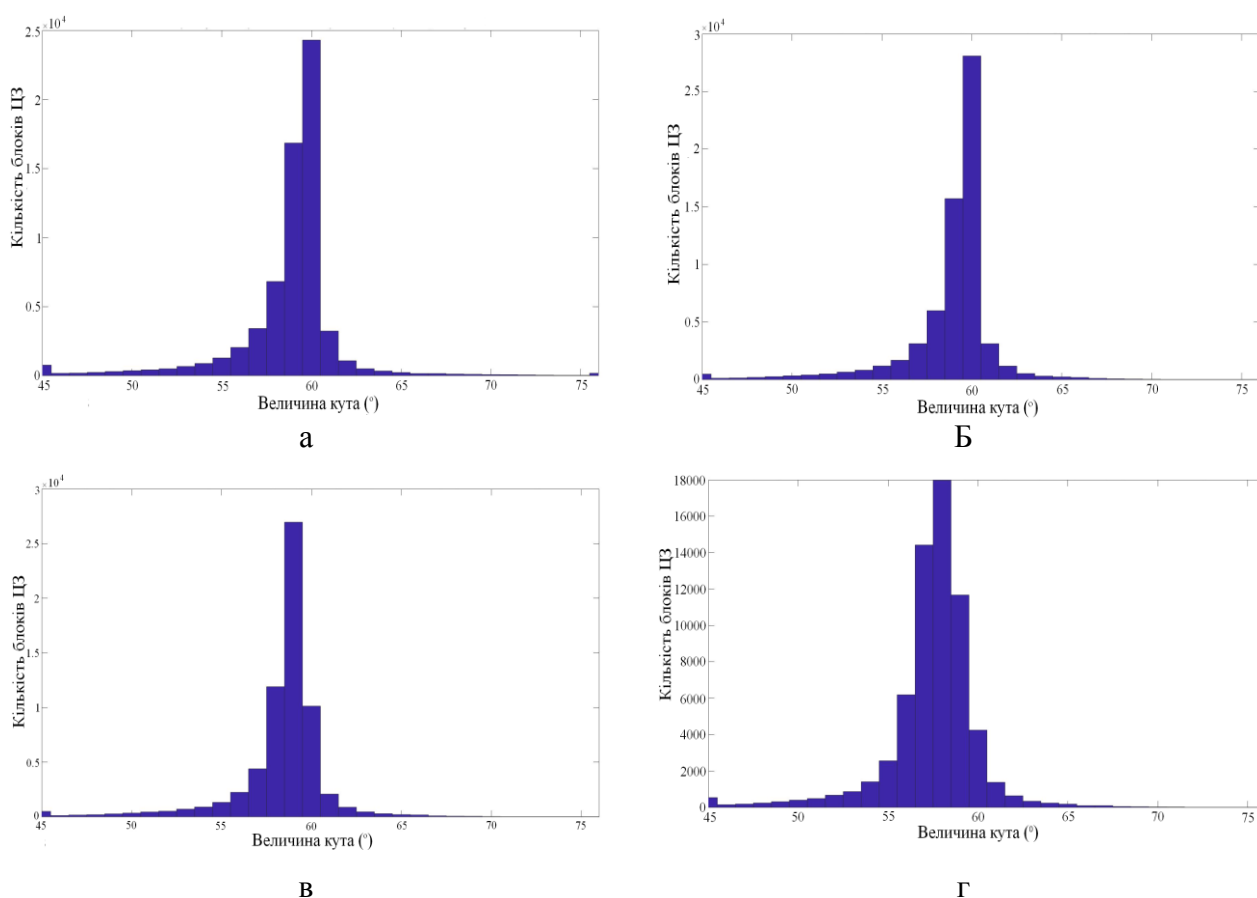


Рис.2.1. Ілюстрація відмінностей відповідних гистограм для конкретного ЦЗ при його стандартному розбитті на 4×4 -блоки: а – гистограма значень кутів між нормованим вектором СНЧ і лівим СНВ, що відповідає максимальному СНЧ, оригінальних блоків оригінального ЦЗ; б – гистограма Γ_λ для оригінального ЦЗ; в, г – Γ_λ для ЦЗ, що піддалося збурній дії (мультиплікативний шум з

$D = 0.001, 0.005$ відповідно)

2.2. Удосконалений метод експертизи цілісності цифрового зображення

Враховуючи все наведене вище, пропонується наступний удосконалений відносно [7] універсальний метод виявлення порушення цілісності ЦЗ, основні кроки якого наведені нижче.

Крок 1. Матриця F аналізованого ЦЗ розбивається стандартним чином на непересічні $l \times l$ -блоки, довільний з яких – блок A .

Крок 2. Кожному блоку A , отриманому на попередньому кроці, ставиться у відповідність симетричний блок

$$B = \frac{A + A^T}{2}.$$

Крок 3. Для аналізованого ЦЗ будується гістограма Γ_λ значень кутів $\angle(\bar{u}_1, \bar{\lambda})$ в блоках B з кроком h .

Крок 4. Для гістограми Γ_λ визначається мода A_λ , а також значення M_λ в моді.

Крок 5. Для аналізованого ЦЗ з використанням Γ_λ обчислюється кількість S_λ блоків, для яких:

$$\angle(\bar{u}_1, \bar{\lambda}) \in [\angle(n^\circ, e_1) - T, \angle(n^\circ, e_1) + T],$$

де T – параметр, що визначається експериментально, характеризує радіус окола $\angle(n^\circ, e_1)$.

Крок 6 (перевірка).

Якщо

$$A_\lambda \notin \{\angle(n^\circ, e_1) - g^\circ, \angle(n^\circ, e_1), \angle(n^\circ, e_1) + g^\circ\}$$

де g° - параметр, що визначається експериментально, залежить від кроку h гістограми Γ_λ , характеризує зсув моди гістограми при порушенні цілісності ЦЗ

то

для аналізованого ЦЗ цілісність порушена.

Якщо

$$(A_\lambda \in \{\angle(n^\circ, e_1) - g^\circ, \angle(n^\circ, e_1), \angle(n^\circ, e_1) + g^\circ\}) \& (S_\lambda / M_\lambda > P),$$

де P – порогове значення, що визначається експериментально, характеризує видозміну гістограми в результаті збурної дії,

то

для аналізованого ЦЗ цілісність порушена.

Якщо

$$(A_\lambda \in \{\angle(n^\circ, e_1) - g^\circ, \angle(n^\circ, e_1), \angle(n^\circ, e_1) + g^\circ\}) \& (S_\lambda / M_\lambda \leq P)$$

то

для аналізованого ЦЗ цілісність не порушена.

Наслідком при реалізації удосконаленого методу очікувано буде збільшення показника $TN(2.1)$ та незбільшення показника TP у порівнянні з первісним методом [7], що в результаті, виходячи з теоретичних міркувань, повинно привести до відносної порівнянності значень ACC для цих двох методів [1].

2.3. Аналіз ефективності удосконаленого універсального методу

Подальші результати обчислювального експерименту приводяться для алгоритмічної реалізації методу при наступних значеннях параметрів:

$$T = 15^\circ, P = 3.2, h = g^\circ = 1^\circ, l \in \{4, 8, 16, 32\}.$$

В обчислювальному експерименті, метою якого була оцінка, в тому числі порівняльна, ефективності алгоритмічної реалізації удосконаленого методу, було задіяно 1000 оригінальних ЦЗ розміром 1024×1024 пікселя.

Результати експерименту, який проводився з використанням двох комп'ютерів (K1, K2) різної конфігурації (табл.2.1), що стосуються оцінки часових витрат для $l \in \{4, 8, 16, 32\}$, представлено в табл.2.2. Отримані результати ілюструють значну часову перевагу запропонованого удосконалення методу для кожного розміру блоку.

Для наочності порівняння часових витрат методів визначимо:

$$Tt = \frac{t_1 - t_2}{t_1} \cdot 100\% , \quad (2.20)$$

де t_1, t_2 — часові витрати для [7] і удосконаленого методу відповідно. Оцінка (2.20) відображена в таблиці 2.2, де перевага удосконаленого методу помітна як для пристрою К1, так і для пристрою К2.

Таблиця 2.1 – Характеристики обчислювальних пристроїв К1, К2, використаних при тестуванні

| Пристрій | К1 | К2 |
|-----------------------------|-------------------------|---------------------------------|
| Ім'я пристрою | Aspire ES1-532G | LenovoIdeaPadGaming 3 15ACH6 |
| RAM | 4 ГБ | 16 ГБ |
| Відеокарта | NVIDIA GeForce 920mx | NVIDIA GeForce RTX 3050 Ti |
| Об'єм відеокарти | 2 ГБ | 4 ГБ |
| Процесор | IntelPentium N3710 | AMD Ryzen 5 5600H |
| Кількість ядер процесора | 4 | 6 |
| Кількість потоків процесора | 4 | 12 |
| Базова швидкість | 1.6 Гц | 3.3 Гц |
| Накопичувач | SSD 512 ГБ | SSD 512 ГБ |

Таблиця 2.2 – Порівняльна оцінка часових витрат експертизи ЦЗ для одного ЦЗ розміром 1024×1024 пікселя від розміру блоку l для пристроїв К1 таК2

| Розмір блоку l | К1 | | | К2 | | |
|------------------|------------------|-----------------------------------|----------|-------------------|-----------------------------------|----------|
| | Метод [11](сек.) | Удоско- налений метод(сек.) | Tt(%) | Метод [11] (сек.) | Удоско- налений метод(сек.) | Tt(%) |
| 4 | 5.392520 | 4.541609 | 15.77947 | 0,577490 | 0,465230 | 19,43934 |

| | | | | | | |
|----|----------|----------|----------|----------|----------|----------|
| 8 | 1.980348 | 1.416426 | 28.47590 | 0,250279 | 0,172278 | 31,16562 |
| 16 | 1.275664 | 0.765268 | 40.01021 | 0,230547 | 0,134410 | 41,69966 |
| 32 | 0.932924 | 0.598114 | 35.88826 | 0,168329 | 0,101768 | 39,54190 |

Треба зазначити, що наявні значні часові переваги удосконаленого методу відносно методу [7] досягаються не тільки в результаті меншої обчислювальної складності безпосередньо спектрального розкладання симетризованих матриць блоків в порівнянні з сингулярним розкладанням оригінальних блоків, які відбуваються в ході експертизи, а й завдяки тому, що у вдосконаленому методі вдвічі зменшується кількість досліджуваних при експертизі об'єктів. Дійсно, в [7] будуються і досліджуються гістограми Γ_U , Γ_V кутів $\angle(u_1, \bar{\sigma})$, $\angle(v_1, \bar{\sigma})$ для першого лівого і першого правого СНВ відповідно, тоді як в запропонованому методі лише Γ_λ , оскільки спектральне розкладання симетричної матриці зменшує кількість параметрів, що визначають відповідну матрицю, в порівнянні з сингулярним розкладанням.

Важливо зазначити, що результат будь-якої збурної дії ΔF , спрямованої на ЦЗ з матрицею F , можна представити у вигляді [29]:

$$\bar{F} = F + \Delta F,$$

(2.21)

де \bar{F} – матриця ЦЗ, цілісність якого порушена. З (2.21) випливає наявність нескінченної кількості різноманітних збурних дій, кожна з яких визначається своєю матрицею ΔF . З урахуванням практичної неможливості розгляду всієї різноманітності збурень, а також того, що результат будь-якої збурної дії загалом може розглядатися як накладання деякого шуму [31], при моделюванні збурних дій в роботі були використані різноманітні шуми з різними параметрами. Результати оцінки, зокрема порівняльної, точності виявлення порушення цілісності ACC (2.1) в таких умовах наведені в табл.2.3.

Для наочності порівняння ефективностей методів визначимо:

$$R = \frac{ACC_2 - ACC_1}{ACC_1} \cdot 100\%, \quad (2.22)$$

де ACC_1 , ACC_2 – значення параметру ACC для [7] і удосконаленого методу відповідно. Оцінка (2.22) знайшла своє відображення в таблиці 2.3, де очевидно є перевага удосконаленого методу для більшості збурних дій при різних розмірах блоків. Така перевага здебільшого пояснюється збільшенням параметру TN , що фігурує в (2.1), тобто більшою кількістю правильно виявлених оригінальних цифрових зображень [1].

Таблиця 2.3 – Значення ACC в умовах різних збурних дій та різних розмірів використовуваних при експертизі ЦЗ блоків

| Збурна дія | Розмір блоку l | ACC | | |
|----------------------------------|---------------------|---------------|------------------------|--------|
| | | (1)Метод [11] | (2)Удосконалений метод | R |
| Пуасонівський шум | 4 | 0.8975 | 0.9280 | 3.3983 |
| | 8 | 0.8665 | 0.9060 | 4.5586 |
| | 16 | 0.8150 | 0.8800 | 7.9755 |
| | 32 | 0.7710 | 0.8325 | 7.9767 |
| Мультиплікативний шум, $d=0.003$ | 4 | 0.8955 | 0.9235 | 3.1267 |
| | 8 | 0.8660 | 0.9060 | 4.6189 |
| | 16 | 0.8150 | 0.8795 | 7.9141 |
| | 32 | 0.7710 | 0.8320 | 7.9118 |
| Мультиплікативний шум, $d=0.005$ | 4 | 0.8975 | 0.9300 | 3.6212 |
| | 8 | 0.8660 | 0.9065 | 4.6767 |
| | 16 | 0.8150 | 0.8805 | 8.0368 |
| | 32 | 0.7710 | 0.8330 | 8.0415 |
| Мультиплікативний шум, $d=0.01$ | 4 | 0.8975 | 0.9315 | 3.7883 |
| | 8 | 0.8660 | 0.9065 | 4.6767 |
| | 16 | 0.8150 | 0.8805 | 8.0368 |
| | 32 | 0.7710 | 0.8330 | 8.0415 |

| | | | | |
|------------------------------|----|--------|--------|----------|
| Гаусів шум, m=0; d=0.0001 | 4 | 0.5735 | 0.5665 | -1.2206 |
| | 8 | 0.5445 | 0.5350 | -1.7447 |
| | 16 | 0.5205 | 0.5470 | 5.0913 |
| | 32 | 0.5200 | 0.5290 | 1.7308 |
| Гаусів шум, m=0; d=0.001 | 4 | 0.8705 | 0.7440 | -14.5319 |
| | 8 | 0.8640 | 0.8905 | 3.0671 |
| | 16 | 0.8130 | 0.8490 | 4.4280 |
| | 32 | 0.7695 | 0.8120 | 5.5231 |

Продовження таблиці 2.3

| Збурна дія | Розмір блоку <i>l</i> | АСС | | |
|-----------------------------|-----------------------------|------------------|---------------------------|---------|
| | | (1)Метод [11] | (2)Удосконалений метод | R |
| Гаусів шум, m=0; d=0.005 | 4 | 0.8980 | 0.9305 | 3.6192 |
| | 8 | 0.8665 | 0.9065 | 4.6163 |
| | 16 | 0.8150 | 0.8805 | 8.0368 |
| | 32 | 0.7710 | 0.8330 | 8.0415 |
| Гаусів шум, m=0; d=0.01 | 4 | 0.8980 | 0.9315 | 3.7305 |
| | 8 | 0.8665 | 0.9065 | 4.6163 |
| | 16 | 0.8150 | 0.8805 | 8.0368 |
| | 32 | 0.7710 | 0.8330 | 8.0415 |
| «Сіль &перець», d=0.01 | 4 | 0.5230 | 0.5265 | 0.6692 |
| | 8 | 0.7265 | 0.7175 | -1.2388 |
| | 16 | 0.8020 | 0.8375 | 4.4264 |
| | 32 | 0.7710 | 0.8330 | 8.0415 |
| «Сіль &перець», d=0.05 | 4 | 0.6840 | 0.6910 | 1.0234 |
| | 8 | 0.8665 | 0.9065 | 4.6163 |
| | 16 | 0.8150 | 0.8805 | 8.0368 |
| | 32 | 0.7710 | 0.8330 | 8.0415 |
| «Сіль &перець», | 4 | 0.897 | 0.9305 | 3.7347 |

| | | | | |
|---------------------------|----|--------|--------|---------|
| d=0.1 | 8 | 0.8665 | 0.9065 | 4.6163 |
| | 16 | 0.8150 | 0.8805 | 8.0368 |
| | 32 | 0.7710 | 0.8330 | 8.0415 |
| Метод Коха і Жао, P=50 | 4 | 0.6620 | 0.6665 | 0.6798 |
| | 8 | 0.6285 | 0.6290 | 0.0796 |
| | 16 | 0.5925 | 0.5860 | -1.0970 |
| | 32 | 0.5735 | 0.5590 | -2.5283 |

Продовження таблиці 2.3

| Збурна дія | Розмір блоку l | ACC | | |
|---------------------------|------------------------|------------------|---------------------------|---------|
| | | (1)Метод [11] | (2)Удосконалений метод | R |
| Метод Коха і Жао, P=60 | 4 | 0.7545 | 0.7590 | 0.5964 |
| | 8 | 0.6710 | 0.6800 | 1.3413 |
| | 16 | 0.6285 | 0.6170 | -1.8298 |
| | 32 | 0.6165 | 0.5835 | -5.3528 |
| Метод Коха і Жао, P=90 | 4 | 0.8970 | 0.9315 | 3.8462 |
| | 8 | 0.8565 | 0.8895 | 3.8529 |
| | 16 | 0.8015 | 0.7825 | -2.3706 |
| | 32 | 0.7475 | 0.6980 | -6.6221 |

І хоча для деяких збурних дій (гауссівський шум ($d=0.001, 0.0001$), «Сіль & перець» $d=0.01$) при деяких значеннях l спостерігалось зменшення ACC, а також у випадку використання в якості збурення – метод Коха і Жао – при всіх збуреннях спостерігалось підвищення ACC при $l=4, 8$ і зменшення – при $l=16, 32$. В цілому очевидним є підвищення точності виявлення порушення цілісності ЦЗ удосконаленим методом: середнє по експерименту значення R є додатним і складає більше ніж 3.5%.

Таким чином, по результатам тестування алгоритмічної реалізації удосконаленого універсального методу виявлення порушення цілісності ЦЗ

можна стверджувати, що його ефективність перевищує ефективність методу [7] як по обчислювальній складності (часу експертизи одного ЦЗ), так і по точності виявлення. Виходячи з представлених результатів найкращим є використання блока при $l=4,8$ так як АСС для всіх збурень є більшим, ніж при $l=16,32$, а виграш в часі при більших блоках не сильно відрізняється .

3. РОЗРОБКА ПРОГРАМНОЇ РЕАЛІЗАЦІЇ УДОСКОНАЛЕНОГО УНІВЕРСАЛЬНОГО МЕТОДУ ВИЯВЛЕННЯ ПОРУШЕННЯ ЦІЛІСНОСТІ ЦИФРОВОГО ЗОБРАЖЕННЯ

3.1. Вибір мови програмування

Для реалізації розробленого методу, було використано MATLAB версії R2017b. Вибір даної мови програмування був зумовлений тим, що в даній роботі виконуються різні розрахунки, а MATLAB саме на це і розрахований, проводити різні розрахунки. Дана система є найбільшою з поміж усіх систем комп'ютерної математики, орієнтація яких зосереджена на ПК. Дана система на даний момент фактично є світовим стандартом у сфері сучасного науково-технічного і математичного ПЗ[32].

Назва MATLAB походить від «MATrix» та «LABoratory», що дослівно перекладається, як матрична лабораторія. Вона була заснована у кінці 1970-х людиною, що пізніше стала в Нью-Мексико керівником університетського відділу обчислювальних наук – Клівом Молером. Головною його задумкою було розроблення програмного продукту, який би давав доступ його студентам до пакетів EISPACK та LINPACK, без потреби опановувати мову програмування Fortran. MATLAB дуже швидко став популярним і в інших університетах, а також прикладні математики зацікавились ним. Передбачаючи успіх MATLAB на ринку, Джон Літл(інженер), приєднався до Молера та Стіва Бангерта, та разом з ними переписав MATLAB на язык програмування C, після чого, у 1984-му році, ними була заснована компанія The MathWorks. Бібліотеки котрі вони переписали, стали відомі світу, як JACSPAC. Окрім викладачів числового аналізу та лінійної алгебри, MATLAB був визнаний спеціалістами, що працювали з ЦЗ.

Використовуючи MATLAB користувачеві надано великий обсяг функцій, що допомагають аналізувати дані, що стосується математики, такі функції торкаються: арифметики, рівнянь, матриць, обробки даних, многочленів, статистики, тощо.

Для реалізації удосконаленого універсального методу, окрім вбудованих функцій MATLAB, було розроблено декілька функцій, це було зумовлено наявністю стрічок коду, що повторялися. Розбиття програмного коду на функції робить читання коду більш легшим.

3.2. Особливості реалізації програмного продукту

Окрім програмного коду, що показано в Додатку А, для реалізації даного методу, потрібні були ще 2 функції, їх створення дало змогу видалити аналогічні блоки коду з основного коду програми, замінивши кожен з них на одну стрічку – назву функції з вхідними та вихідними параметрами.

Функція отримання масиву кутів між останнім ВВ та нормованим вектором ВЗ має наступний вигляд:

```
function [arU] = func_diplom_eig_degree_v3(A,N,k)
arU=zeros(1,(N/k)^2);%масив кутів
t=1;
for i=1:k:N
    for j=1:k:N
        [U, SIGMA]=eig((A(i:i+k-1,j:j+k-1)+A(i:i+k-1,j:j+k-1)')/2);
        for ii = 1:k
            if U(1,ii)<0
                U(:,ii)=-U(:,ii);
            end
        end
        a=sum(abs(SIGMA));
        b=sqrt(sum(a.^2));

arU(t)=acos(sum(U(:,end).*a')/(sqrt(sum(U(:,end).^2))*b))*180/pi;
        t=t+1;
    end
end
arU=arU(arU>1 & arU<180);
end
```

Перша стрічка – функція повертає масив кутів arU, на свій вхід вона приймає:

- А – квадратна двовимірна матриця створена взяттям однієї з трьох кольорових компонентів ЦЗ RGB, попередньо обрізана до розмірів NxN;
- N– ширина і довжина матриці А;

- k – розмір досліджуваного блоку матриці A для якого відбувається знаходження кута.

В другій стрічці створюється пустий масив arU потрібного розміру (виходячи з вхідних параметрів даної функції), який в кінці функції буде заповнений кутами між останнім VB та нормованим вектором $V3$ для кожного непересічного блоку $k \times k$ матриці A .

Третя стрічка – створюється лічильник t для масиву arU .

З четвертої по сімнадцяту стрічки – цикл в циклі, відбувається розбиття матриці A на непересічні блоки $k \times k$,

В шостій стрічці – для кожного непересічного блоку відбувається перетворення його на симетричний, елементи якого знаходяться в діапазоні $[0;255]$. Потім отриманий симетричний блок розкладають, за допомогою спектрального розкладання на матрицю VB та матрицю $V3$.

З сьомої по одинадцяту стрічку відбувається перевірка на лексикографічну додатність VB , якщо вони не є такими, то знак такого вектора змінюється на протилежний. На виході ми отримуємо матрицю лексикографічно додатніх VB , що означає що ми зробили нормальне спектральне розкладання.

Дванадцята стрічка – знаходимо: a – Вектор $V3$ взятих по модулю.

Тринадцята стрічка – знаходимо: b – норма a (отриманого на попередній стрічці).

Чотирнадцята стрічка – знаходимо t та записуємо в масив arU під індексом t – кут в градусах між останнім VB та нормованим вектором $V3$.

П'ятнадцята стрічка – збільшення t на одиницю.

Вісімнадцята стрічка – запасна перевірка, яка виключає з масиву arU кути, що не входять в діапазон $(1;180)$.

Дев'ятнадцята стрічка – кінець функції.

Функція отримання масиву кутів між останнім VB та нормованим вектором $V3$ має наступний вигляд:

```
function [Au,Mu,Su,answ] = func_diplom_eig_degree_v6_anw(U,a,Pu,T)
Hu = hist(U,a-T-1:1:a+T+1);
```

```

[Mu, Au] = max(Hu);
Au = a-T-2+Au;
Su = sum(Hu(2:end-1));
answ = 0;
if Au>=a-1 & Au<=a+1 & (Su/Mu)<=Pu
    answ=1;%цілісність ЦЗ не порушена
end
end

```

Перша стрічка – функція повертає:

- Au – моду гістограми кутів масиву U;
- Mu– значення в моді гістограми кутів масиву U;
- Su– значення в околі гістограми кутів масиву U;
- Answ– якщо 1, то зображення – оригінальне, якщо 0 – неоригінальне.

На свій вхід вона приймає:

- U– масив кутів, отриманий з першої функції;
- a– величина кута між n-оптимальним вектором та вектором стандартного базису для даного розміру блоку (k);
- Pu – величина параметру який вказує, що кількість кутів в околі кута поділена на кількість кутів в моді не повинна перевищувати заданий параметр ($(Su/Mu) \leq Pu$).
- T– величина околу кута a в градусах.

В другій стрічці за допомогою функції `hist()` записую в масив `Hu` кількість кутів для кожного кута з кроком 1 в на один більшому околі заданого `a`, береться на один більше адже коли ми використовуємо функцію `hist()`, на обмеженому проміжку всі залишки(кути що не ввійшли в заданий діапазон) записуються в околі діапазону. Збільшенням околу на 1, ми гарантуємо що в околі кута `a` дійсно залишаться лише кути, які входять до цього околу. Наприклад, якщо буде розглядатися розбивка на непересічні блоки 4x4, то `a = 60`, то при околі в 15 – у масиві `Hu` буде 33 елементи які відповідають градусним мірам кутів з діапазону [44, 45, 46, ..., 74, 75, 76]. На місці кожного елементу буде кількість таких кутів.

Третя стрічка – за допомогою функції `max()` в змінну `Mu` записується найбільше число з масиву `Hu` (іншими словами записуємо значення в

моді побудованої раніше гістограми), а в змінну A_u – індекс цього найбільшого числа в масиві H_u .

В четвертій стрічці, знаходимо градусну міру A_u , тобто моду побудованої раніше гістограми. Знаючи що зараз A_u – це індекс найбільшого числа в масиві H_u і те якому кутіві відповідає який індекс(в прикладі вище $2=45, 3=46\dots$).

В п'ятій стрічці – знаходимо суму кутів в заданому околі змінною T , та записуємо отриману суму кутів в змінну S_u .

В шостій стрічці створюємо змінну $answ$, та присвоюємо їй значення 0.

З сьомої по дев'яту стрічки йде перевірка на оригінальність. Якщо мода гістограми відхилився не більше ніж на 1 градус від заданого раніше a , та значення виразу S_u/M_u не перевищує P_u , то змінна $answ$ змінить своє значення на 1, що вказує на те, що програма вважає дане ЦЗ оригінальним.

Десята стрічка – кінець функції.

Завантаження ЦЗ відбувається при натисненні на кнопки, у випадку з першою кнопкою – це реалізовано наступним чином:

```
path=uigetdir('Title','Оберіть директорію з зображеннями для
подальшого їх аналізу');
while path==0
path=uigetdir('Title','Оберіть директорію з зображеннями для
подальшого їх аналізу');
end
myfold=dir(path);
Max=size(myfold,1);
```

З даного коду стає зрозуміло, що натиснувши на першу кнопку – користувачеві одразу відкривається вікно з заголовком «Оберіть директорію з зображеннями для подальшого їх аналізу» та пропонує вибрати директорію з зображеннями, після чого програма вже знатиме яку директорію слід сканувати. Варто звернути увагу на цикл `while` з умовою «`path==0`», даний цикл гарантує що користувач вибере директорію, а не натисне на скасувати чи закриє вікно не вибравши директорію. Тепер, знаючи шлях до зображень, які програма буде перевіряти, за допомогою функції `dir()` створюється структура за допомогою якої можна дізнатися наступне:

1. ім'я файлу;

2. абсолютний шлях до файлу;
3. дата останньої модифікації файлу;
4. величина файлу в байтах;
5. чи це є директорія (тобто піддиректорія у вибраній раніше користувачем директорії);
6. дата останньої модифікації файлу – як серійний номер дати MATLAB.

Тепер, знаючи що робить функція `dir()`, стає зрозумілим, що в останній стрічці коду програма дізнається скільки елементів (файли та піддиректорії) містяться у вибраній раніше директорії з зображеннями. Наступні стрічки коду розкажуть яким чином вибирається зображення для перевірки з вибраної раніше директорії:

```
Allext={' .bmp';'.tiff';'.tif';'.jpg';'.jpeg';'.png'};
...
iter=0;
...
for i=3:Max
    file=strcat(path, '/',myfold(i).name);
    [filepath,~,ext]=fileparts(file);
    if isfile(file) & ismember(ext,Allext)
        IMG=imread(file);

        [x,y,z]=size(IMG);
        ifz==3 &x>=N&y>=N
start_iter = tic;
iter=iter+1;
...

```

Маючи директорію і знаючи скільки в тій директорії елементів (файлів та піддиректорій) програма запускає цикл щоб пробігтися по кожному елементу, починаючи з третього. Починається саме з третього, тому що перші два елементи у кожній директорії є «.» та «..». Далі йде 5 перевірок (2 до зчитування зображення програмою та 3 вже після, перевірка чи підходить дане зображення):

1. чи файл дійсно є файлом?
2. чи розширення файлу входить в список допустимих, записаних в масиві `Allext`?
3. чи зчиталося дане зображення як 3 кольорові складові чини?
4. чи ширина зображення є неменшою ніж вибране `N`?

5. чи довжина зображення є неменшою ніж вибране N ?

Лише якщо елемент вибраної раніше директорії пройшов усі ці перевірки – змінна `iter`, яка рахує ітерації циклу, збільшується на одиницю, а після завершення перевірки всієї директорії в даній змінній буде зберігатися кількість зображень які були перевірені. Окрім цього, як тільки зображення пройшло всі перевірки вмикається таймер який вимкнеться після закінчення однієї ітерації циклу. І знаючи скільки елементів в заданій директорії залишилося та скільки часу витрачається на кожне зображення. Програма після кожної ітерації циклу оновлює інформацію про те, скільки часу ще залишилося, ось як це в кодї реалізовано:

```
end_iter(iter)=toc(start_iter);
set(handles remainder, 'String', string((Max-
i)*(sum(end_iter)/iter)/60));
pause(1);
```

Функція в останній стрічці потрібна для того, щоб інтерфейс програми оновився інакше програма не встигає оновити інтерфейс, та перевіряє наступні зображення з даної директорії.

Наступний фрагмент коду вказує яким чином вхідне зображення обрізається до розмірів $N \times N$:

```
xx=fix((x-N)/2);
yy=fix((y-N)/2);
A=double(IMG(xx+1:xx+N, yy+1:yy+N, K));
```

З зображення вибирається одна кольорова складова, вказана користувачем раніше (K), а також зображення обрізається до розміру $N \times N$ в центрі вхідного зображення. Це зроблено з таким задумом, що головне в зображенні скоріш за все розміщене в центрі зображення.

Фрагмент коду для блоку 4×4 , який реалізує перевірку отриманої матриці
Апредставлено нижче:

```
start_kyt=tic;
[U4]=func_diplom_eig_degree_v3(A,N,4);
[Au4, Mu4, Su4, answ]=func_diplom_eig_degree_v6_answ(U4, 60, Pu1, T1);
if answ
    answ4 = 'Оригінальне';
    origin_img(1,1) = origin_img(1,1) + 1;
else
    answ4 = 'Підроблене';
```

```

noise_img(1,1) = noise_img(1,1) + 1;
end
speed_sec_e(1,1) = toc(start_kyt);

```

Як бачимо з першої та останньої стрічок коду, час затрачений саме на виконання даного проміжку коду записується в окрему змінну `speed_sec_e(1,1)`.

Друга стрічка – використовуємо описану раніше функцію, щоб отримати масив кутів між останнім ВВ та нормованим вектором ВЗ для кожного непересічного блоку 4x4 та записати результат в U4.

Третя стрічка – використовуємо описану раніше функцію, щоб дізнатися моду гістограми кутів(Au4), значення в моді(Mu4) та значення в околі(Su4), а також результат перевірки(answ).

Четверта-десята стрічки – відбувається перевірка, якщо зображення оригінальне, то в масив `origin_img` з індексом одиниця збільшиться на одиницю. Взагалі даний масив має чотири елементи, кожен з яких відповідає одному розміру блоку. Кожен елемент збільшується лише у випадку коли його розмір блоку для певного зображення показав що зображення є оригінальним. Якщо ж зображення було підробленим, то аналогічний масив, ціль якого підрахувати кількість підроблених зображень `noise_img` збільшиться на одиницю. Окрім цього в змінну `answ4` записується результат перевірки, який в подальшому запишиться в файл.

Фрагмент коду, що відповідає за збереження результату програми після кожної ітерації циклу представлено нижче:

```

All_time_e = All_time_e + speed_sec_e;
xlswrite(xlsFile_e, {myfold(i).name,'4',Au4,Su4/Mu4, answ4, ''};
            '','8',Au8,Su8/Mu8, answ8, ''};
            '','16',Au16,Su16/Mu16, answ16, ''};
            '','32',Au32,Su32/Mu32, answ32, sum(speed_sec_e)}},
strcat('A',string(4*iter-2),':F',string(4*iter+1)));
...
xlswrite(xlsFile_e,
{'blok','Оригінальних','підроблених','SumImOrigin','','Time(sec)';
 '4',origin_img(1,1),noise_img(1,1),iter, '',
All_time_e(1,1)/iter;
 '8',origin_img(1,2),noise_img(1,2),iter, '',
All_time_e(1,2)/iter;

```



```
'16',origin_img(1,3),noise_img(1,3),iter, '',
All_time_e(1,3)/iter;
'32',origin_img(1,4),noise_img(1,4),iter, '',
All_time_e(1,4)/iter;},'H1:M5');
```

Перша ділянка коду, до «...» в кінці кожної ітерації циклу робить дві дії:

1. збільшує загальний час виконання по кожному блоці;
2. записує результат перевірки в .xlsxфайл, який було створено раніше.

Друга ділянка коду – це код після завершення циклу. Відкривається той самий файл .xlsx і в ньому ж справа вгорі, правіше від попередніх записів створюється підсумовуюча таблиця.

Останньою особливістю першої кнопки є перевірка, яка продемонстрована на коді нижче:

```
if iter>0
msgbox(char(strcat('Результати перевірки збережено в
','',xlsFile_e,'')), 'Інформація', 'help');
else
delete(char(xlsFile_e));
msgbox('Дана директорія не містила ні одного зображення: довжина і
ширина якого були б більші або рівні за "N", або яке б можна було
розкласти на 3 кольорові складові RGB', 'Інформація', 'help');
end
```

Дана перевірка розміщена наприкінці коду першої кнопки. Користувач:

- Отримає повідомлення, в якому йдеться де збережено файл з результатами, у випадку, якщо хоча б один елемент з вибраної користувачем директорії виявився зображенням, яке задовільнило 5 початкових умов, на початку циклу.

- Отримає повідомлення про помилку, якщо такого зображення не знайшлося. А також буде видалено пустий файл з результатами по даній директорії(адже даний файл створюється одразу коли користувач вибирає директорію).

У випадку з другою кнопкою – завантаження ЦЗ відбувається одразу з перевіркою, лише пройшовши яку – зображення буде перевірено аналогіно, як одна ітерація циклу попередньої кнопки, за виключенням елементів збереження у файл та фіксація часу роботи, так як це є недоречним у випадку 1 зображення. Код завантаження зображення продемонстровано у фрагменті нижче.

```
[filepathfilterindex]=uigetfile({'*.bmp; *.tiff; *.tif; *.jpg;
*.jpeg; *.png'}, 'Оберіть зображення (запропонованого формату) для
перевірки його оригінальності');
iffilterindex~=1
msgbox('Будь-ласка, оберіть зображення, не змінюючи фільтр файлів на
"AllFiles" та не закриваючи діалогове вікно.', 'Помилка', 'warn');
else
    IMG=imread(strcat(path, file));
    [x,y,z]=size(IMG);
    if x<N || y<N
msgbox('Будь-ласка, оберіть зображення, Ширина і Довжина якого
більші або рівні за параметр "N".', 'Помилка', 'warn');
    elseif z~=3
msgbox('Нажаль вибране вами зображення неможна розкласти на 3
кольорові складові RGB, оберіть інше зображення, будь-ласка.',
'Помилка', 'warn');
else
```

Як можна зрозуміти з даних стрічок коду, при натисненні на кнопку, перед користуваєм відкривається вікно з заголовком «Оберіть зображення (запропонованого формату) для перевірки його оригінальності» та пропонує вибрати файли з розширеннями, які є популярними для ЦЗ, а саме: «.bmp», «.tiff», «.tif», «.jpg», «.jpeg», «.png», після того, як тільки він зробить вибір, програма уже знатиме повний шлях до вибраного зображення та почне перевіряти дане зображення, щоб зрозуміти воно є «Оригінальним» чи «Підробленим». Користувач побачить повідомлення про помилку, з тексту якого зрозуміє чому сталася помилка, якщо користувач не вибере файл, або обраний файл не буде виконувати наступні вимоги:

1. користувач при виборі змінив фільтр(розширення) файлів;
2. користувач вибрав зображення ширина або довжина якого менші за параметр N;
3. вибране користувачем зображення не можна розкласти на три кольорові складові RGB (така помилка спостерігалася при скануванні деяких файлів р розширенням .tif/.tiff).

На відміну від першої кнопки, запис результату в другій кнопці відбувається в інтерфейс програми. Результатом є надписи зліва під другою кнопкою, а також гістограми для блоків 4x4,8x8,16x16,32x32. Фрагмент коду що

відповідає за це для блоку 4x4, так код який вирішує остаточно зображення «Оригінальне» чи «Підроблене» продемонстровано нижче.

```

if answ
    set(handles.text_one_result44,'String','Оригінальне');
    answ4 = 1;
else
set(handles.text_one_result44,'String','Підроблене');
end
axes(handles.axes1)
hist(U4,60-T1-1:1:60+T1+1); m=axis; axis([60-T1-1 60+T1+1 m(3)
m(4)]);
xlabel('градусна міра кутів');
ylabel('кількість кутів');
title('Гістограма величинів кутів для блоку 4x4');% між ВВ і
нормованим вектором ВЗ
...
if answ4 && answ8
set(handles.text_one_main_result,'String','Оригінальне');
set(handles.text_one_main_result,'ForegroundColor',[0 1 0]);
else
set(handles.text_one_main_result,'String','Підроблене');
set(handles.text_one_main_result,'ForegroundColor',[1 0 0]);
end

```

Змінна `answ4` була створена раніше зі значення 0, тому лише у випадку, коли ЦЗ «Оригінальне», значення цієї змінної зміниться на 1. Як можна зрозуміти з коду вище, відповідь для блоку 4x4 записується у відповідний об'єкт інтерфейсу, а потім на графіку, в програмі, відображається гістограма для даного ЦЗ при розбивці його на блок 4x4. В умові що розміщена після «...» сказано, що коли дане ЦЗ при розбивці його на блоки 8x8 та 4x4 веде себе як оригінальне, то висновок програми – ЦЗ є «Оригінальним».

Розроблений програмний продукт має декілька перевірок вхідних параметрів, які вводить користувач до натиснення на кнопки, що зменшує кількість навмисних та випадкових помилок під час роботи з даним програмним продуктом. Запрограмовані перевірки гарантують що при будь-яких даних користувача програма запрацює, змінивши його вхідні дані на найближчі можливі з доступного діапазону.

3.3.Інтерфейс програми

При відкритті розробленої програми з'являється інтерфейс як на рисунку 3.1. На представленій формі в верху присутній надпис – який слугує заголовком розробленого програмного продукту. Зліва розміщені чотири мітки з відповідними їм, чотирма полями вводу: «N», «K», «T», «Pi», перше поле вводу слугує як розмір до якого буде обрізано по центрі кожне ЦЗ перед тим як перевірити що воно «Оригінальне» чи «Підроблене», друге поле вказує на кольорову складову, яка буде використана для аналізу ЦЗ, третє поле вказує на окіл, який буде розглядатися при аналізі ЦЗ, четверте поле слугує пороговим значенням суми кутів в околі поділений на суму кутів в моді гістограми. Також під цими полями наявні 2 кнопки: «Старт програми на директорії», «Старт програми на файлі», вони відповідно відповідають за: старт програми на вибраній користувачем директорії, аналіз одно ЦЗ вибраного користувачем. Окрім цього, праворуч наявні чотири підписані області для графіків, які відповідають за відображення гістограм після аналізу ЦЗ яке аналізується після натиснення на кнопку «Старт програми на файлі», ці 4 області підписані: «Гістограма величинів кутів для блоку 4x4», «Гістограма величинів кутів для блоку 8x8», «Гістограма величинів кутів для блоку 16x16», «Гістограма величинів кутів для блоку 32x32», і відповідають за 4 гістограми, для різних блоків: 4x4, 8x8, 16x16, 32x32 відповідно.

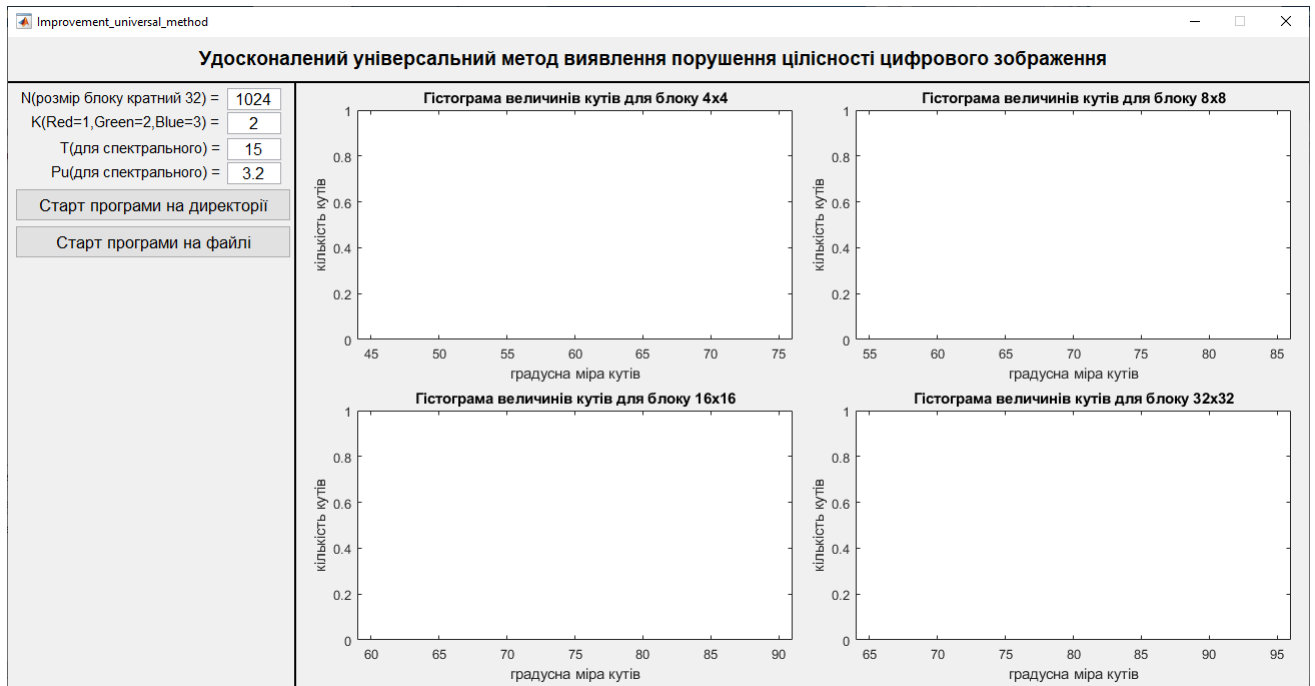


Рисунок 3.1 – Головна форма розробленого інтерфейсу програми

Після натиску на першу кнопку, користувача інформують про те, що він повинен вибрати директорію з зображеннями для подальшого їх аналізу (рис. 3.2).

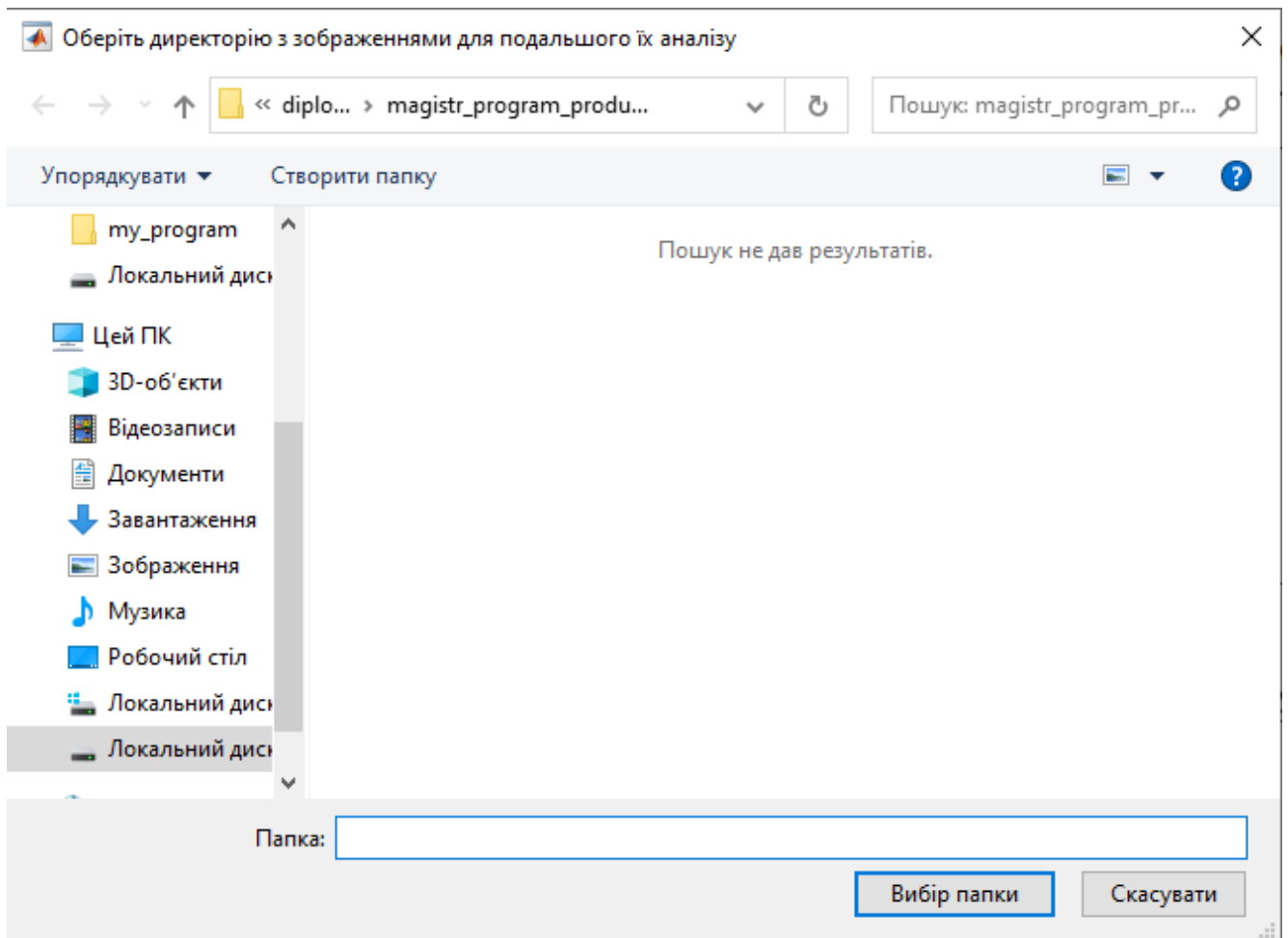


Рисунок 3.2 – Вибір директорії з зображеннями для подальшого їх аналізу

Після цього, всі необхідні дані для виконання програми на вибраній користувачем директорії присутні, поки програма працює – на головній формі оновлюється приблизний час (в хвилину) закінчення перевірки, а також відображена директорія яка сканується (рисунок 3.3). Після закінчення виконання першої кнопки, якщо в обраній директорії було хоч одне ЦЗ що виконує всі умови, то користувач може відкрити згенерований .xlsxфайл з результатом перевірки (рисунок 3.4), а також з'явиться інформація(повідомлення) про успішне виконання першої кнопки (рисунок 3.5).

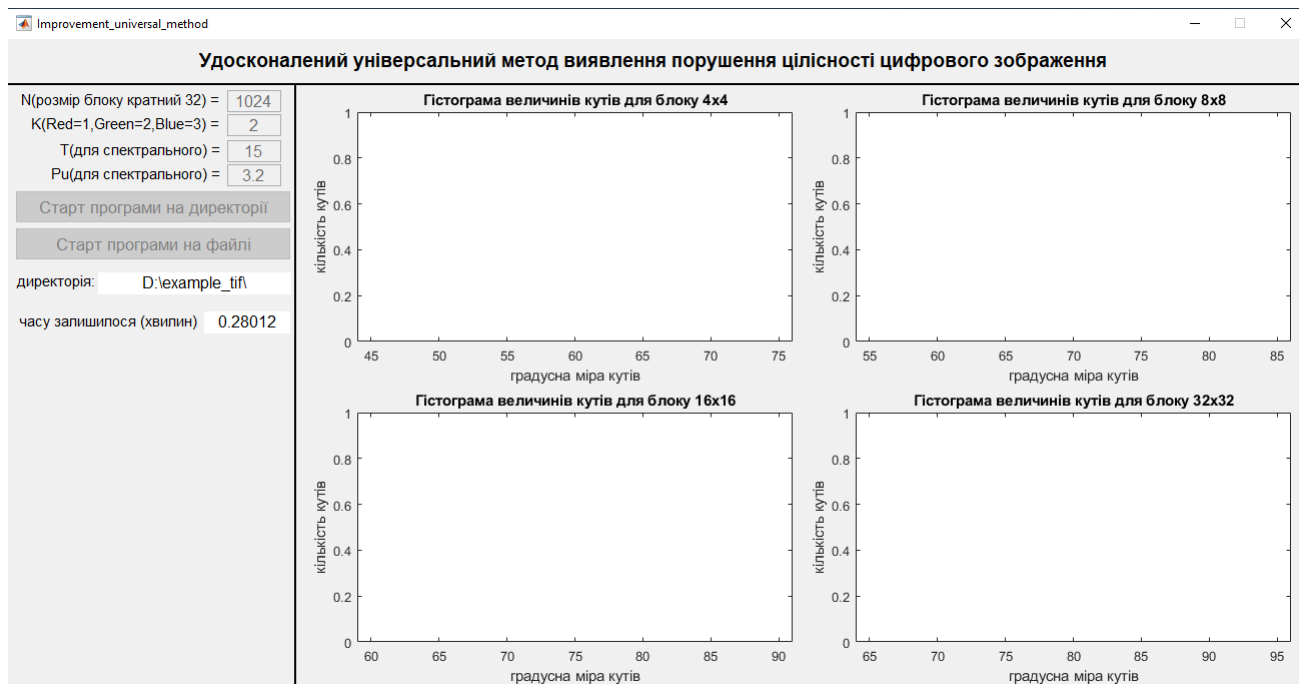


Рисунок 3.3 – Головна форма під час роботи програми після натиснення на кнопку «Старт програми на директорії»

| | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T |
|----|---|-----------------|----|-------|-------------|-------------|-----------|------|----------|----------|-------------|-----------|---|---|---|---|---|---|---|---|
| 1 | | blok | Au | Su/Mu | Result | Time | | blok | Оригінал | підробле | SumImOrigin | Time(sec) | | | | | | | | |
| 2 | | NRCSMD08014.tif | 4 | 60 | 1,400012818 | Оригінальне | | 4 | 5 | 2 | 7 | 4,865327 | | | | | | | | |
| 3 | | | 8 | 69,5 | 2,097286226 | Оригінальне | | 8 | 4 | 3 | 7 | 1,520663 | | | | | | | | |
| 4 | | | 16 | 75,6 | 1,991249392 | Оригінальне | | 16 | 4 | 3 | 7 | 0,923822 | | | | | | | | |
| 5 | | | 32 | 78,9 | 2,081300813 | Оригінальне | 7,9199951 | 32 | 3 | 4 | 7 | 0,786802 | | | | | | | | |
| 6 | | NRCSMD08063.tif | 4 | 59 | 4,010282776 | Підроблене | | | | | | | | | | | | | | |
| 7 | | | 8 | 67,5 | 3,870120653 | Підроблене | | | | | | | | | | | | | | |
| 8 | | | 16 | 73,6 | 3,300727567 | Підроблене | | | | | | | | | | | | | | |
| 9 | | | 32 | 76,9 | 3,32247557 | Підроблене | 7,7566088 | | | | | | | | | | | | | |
| 10 | | NRCSND09047.tif | 4 | 60 | 3,020456225 | Оригінальне | | | | | | | | | | | | | | |
| 11 | | | 8 | 68,5 | 3,053261073 | Оригінальне | | | | | | | | | | | | | | |
| 12 | | | 16 | 74,6 | 3,120518688 | Оригінальне | | | | | | | | | | | | | | |
| 13 | | | 32 | 78,9 | 3,706521739 | Підроблене | 7,8569197 | | | | | | | | | | | | | |
| 14 | | NRCSND09050.tif | 4 | 60 | 1,776828574 | Оригінальне | | | | | | | | | | | | | | |
| 15 | | | 8 | 69,5 | 2,56630094 | Оригінальне | | | | | | | | | | | | | | |
| 16 | | | 16 | 75,6 | 2,619961612 | Оригінальне | | | | | | | | | | | | | | |
| 17 | | | 32 | 78,9 | 2,149159664 | Оригінальне | 7,873934 | | | | | | | | | | | | | |
| 18 | | NRCSND09096.tif | 4 | 60 | 3,679427992 | Підроблене | | | | | | | | | | | | | | |
| 19 | | | 8 | 68,5 | 4,063760897 | Підроблене | | | | | | | | | | | | | | |
| 20 | | | 16 | 73,6 | 4,719399538 | Підроблене | | | | | | | | | | | | | | |
| 21 | | | 32 | 76,9 | 4,802816901 | Підроблене | 8,7870119 | | | | | | | | | | | | | |
| 22 | | NRCSVA08022.tif | 4 | 60 | 2,91635878 | Оригінальне | | | | | | | | | | | | | | |
| 23 | | | 8 | 68,5 | 3,228554526 | Підроблене | | | | | | | | | | | | | | |
| 24 | | | 16 | 74,6 | 3,490614334 | Підроблене | | | | | | | | | | | | | | |
| 25 | | | 32 | 77,9 | 3,908396947 | Підроблене | 8,4507586 | | | | | | | | | | | | | |
| 26 | | NRCSVA08126.tif | 4 | 59 | 1,929856592 | Оригінальне | | | | | | | | | | | | | | |
| 27 | | | 8 | 68,5 | 1,7061335 | Оригінальне | | | | | | | | | | | | | | |

Рисунок 3.4 – Приклад згенерованого .xlsxфайлу з результатом після перевірки директорії, тобто успішного виконання програми на першій кнопці «Старт програми на директорії»

У випадку коли у вибраній користувачем директорії не було ні одного ЦЗ що виконує всі умови, то користувачу відображається інформація(повідомлення) про помилку, і що програма не проаналізувала ні одного зображення з вибраної користувачем директорії(рисунок 3.6).

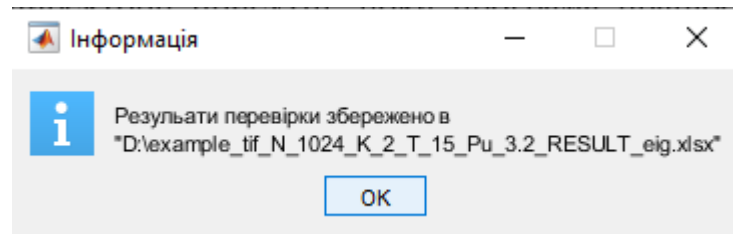


Рисунок 3.5 – Повідомлення після успішного виконання програми на першій кнопці «Старт програми на директорії»

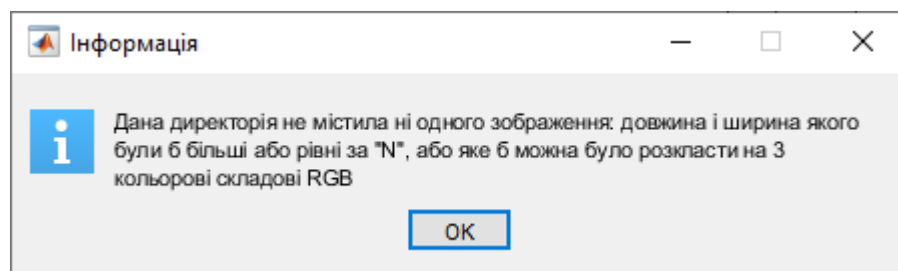


Рисунок 3.6 – Повідомлення про те, що у вибраній директорії не було ні одного ЦЗ що виконує умови, після запуску програми на першій кнопці «Старт програми на директорії»

Після натиснення на другу кнопку, мітки з інформацією про директорію та час що залишився до виконання програми – зникають, на заміну їм під кнопками з'являється 5 міток – результати по блоках 4x4, 8x8, 16x16, 32x32 та загальний в цілому висновок по ЦЗ, відкривається вікно (рисунок 3.7), в якому користувачеві пропонують вибрати зображення (запропонованого формату) для перевірки його оригінальності. Після закінчення аналізу ЦЗ, на головній формі програми відображаються 4 гістограми, результат перевірки для кожного виду блоків, а також висновок перевірки даного ЦЗ (рисунок 3.8).

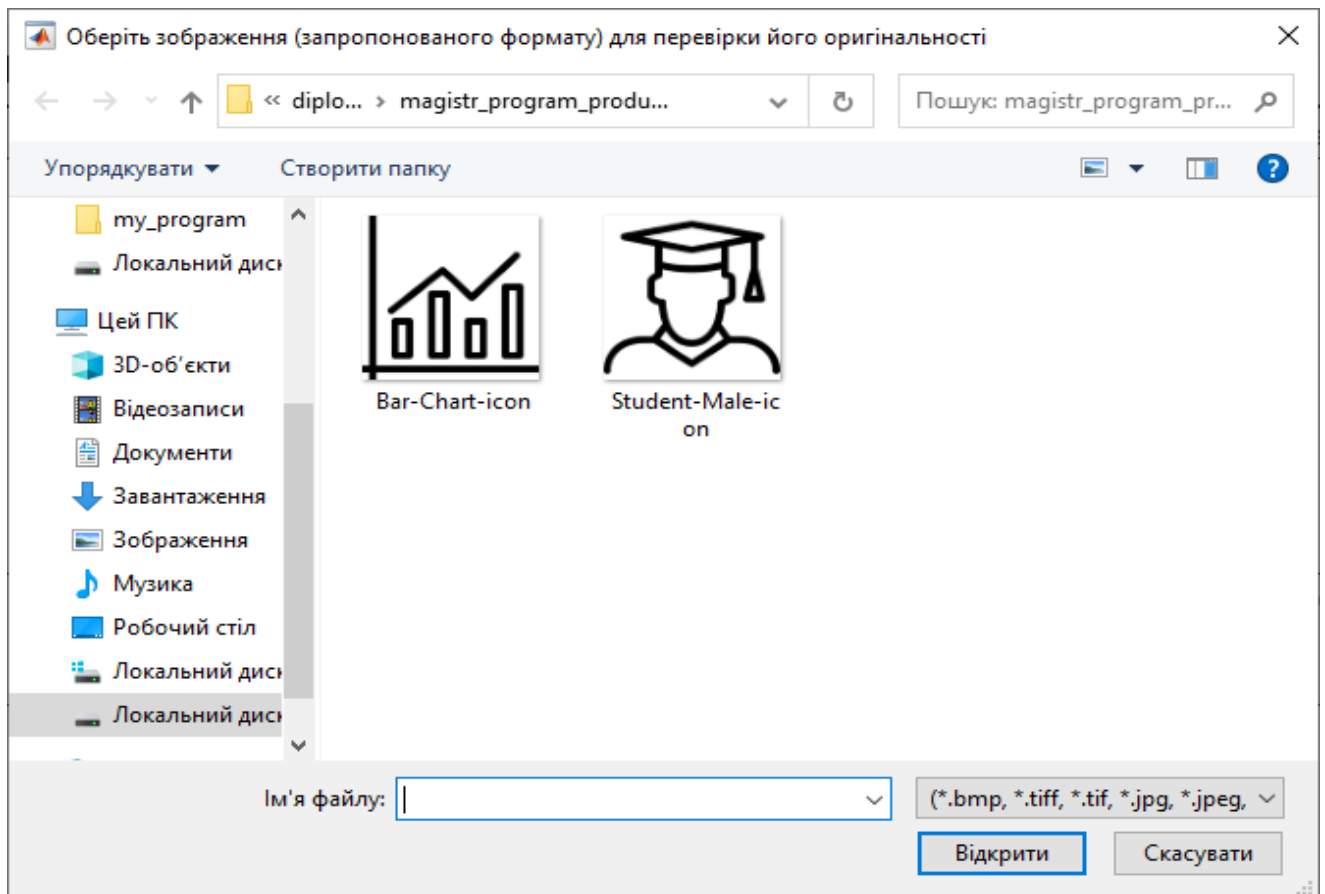


Рисунок 3.7 – Вибір ЦЗ (запропонованого формату) для подальшого його аналізу

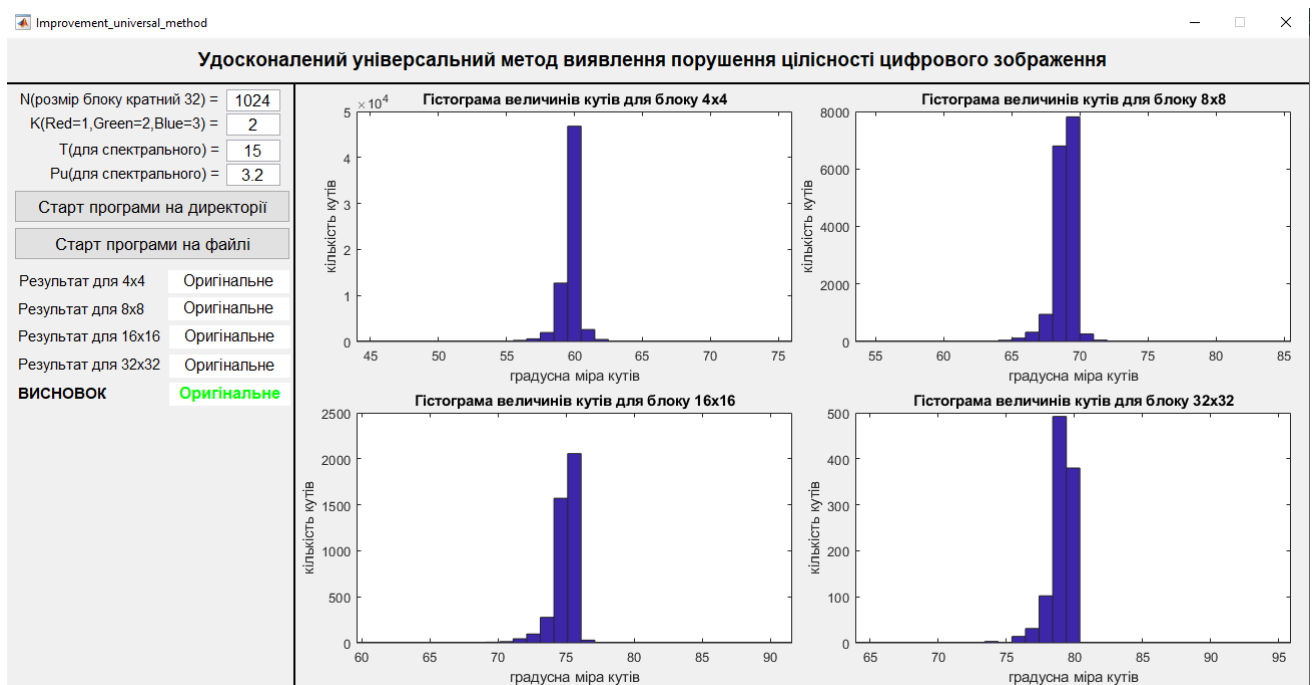


Рисунок 3.8 – Головна форма після закінчення роботи програми після натиснення на кнопку «Старт програми на файлі»

При некоректній роботі з другою кнопкою програмного продукту, користувачу може бути відображено триповідомлення. Перше повідомлення з'являється у випадку, якщо при виборі ЦЗ користувач натиснув змінити фільтр із запропонованих розширень на «AllFiles», або закрив діалогове вікно чи натиснув скасувати (що також закриває діалогове вікно) (рис.3.9). Друге – у випадку, якщо обране ЦЗ не було потрібної ширини або довжини(рис.3.10). Третє – якщо ЦЗ було потрібного розміру, але програма не змогла його розкласти на 3 кольорові складові RGB(рисунок 3.11).

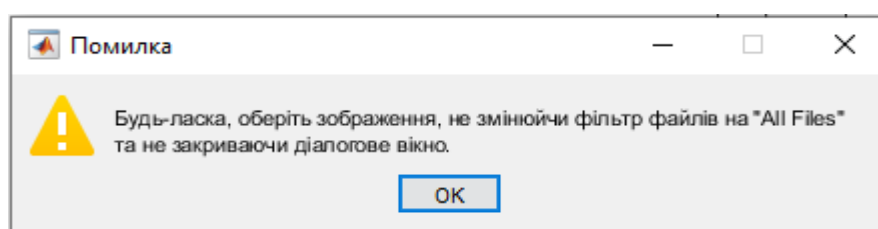


Рисунок 3.9 – Повідомлення про те, що користувач вибрав інший фільтр, або закрив діалогове вікно, після натиснення на кнопку «Старт програми на файлі»

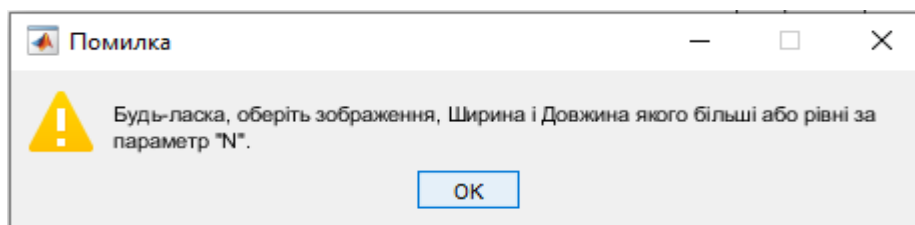


Рисунок 3.10 – Повідомлення про те, що користувач вибрав ЦЗ, ширина або довжина якого менші за «N», після натиснення на кнопку «Старт програми на файлі»

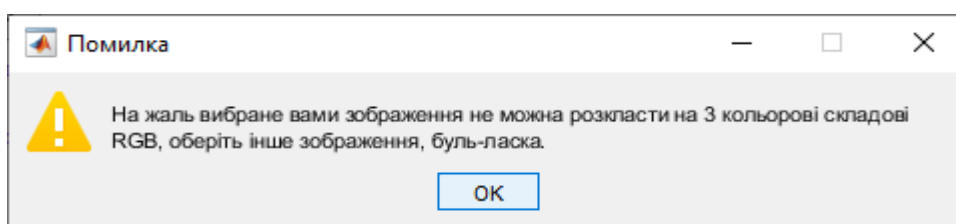


Рисунок 3.11 – Повідомлення про те, що користувач вибрав ЦЗ, яке не вдалося розкласти на 3 кольорові складові (RGB), після натиснення на кнопку «Старт програми на файлі»

Також під час роботи програми, після натиснення на будь-яку з кнопок, всі поля вводу та кнопки на головній формі стають недоступні до натиснення або змін до того часу, поки дії запрограмовані в кнопці не закінчать свою дію.

Отже, в даному розділі було створено програмний продукт з користувацьким інтерфейсом, що реалізує удосконалений універсальний метод виявлення порушення цілісності цифрового зображення. Було описано особливості написання коду в мові програмування MATLAB. Крім того наочно показано на рисунках – як працює розроблений програмний продукт, які можливі повідомлення можуть з'являтися та що вони означають. Використання даного програмного продукту є доступним та зрозумілим при його використанні рядовим користувачем.

ВИСНОВКИ

В роботі вирішено важливу та актуальну науково-практичну задачу підвищення ефективності виявлення порушень цілісності ЦЗ шляхом удосконалення універсального методу, запропонованого в [7].

Мета роботи була досягнута завдяки теоретично обґрунтованому вибору способу симетризації матриці $l \times l$ -блоку ЦЗ з наступним доведенням того, що для більшості отриманих симетричних блоків, що ставляться у відповідність блокам оригінального ЦЗ, кут між лексикографічно додатним власним вектором, що відповідає максимальному власному значенню симетризованого блока, і нормованим вектором модулів власних значень дорівнює куту між n -оптимальним та першим вектором e_1 стандартного базису відповідного простору R^l . Найбільш важливим результатом роботи є удосконалений універсальний метод виявлення порушення цілісності ЦЗ, ефективність якого перевищує ефективність методу-прототипу. Властивості параметрів що аналізуються в процесі експертизи, зменшення їх кількості в симетричних блоках дають можливість підвищити показник правильно виявлених оригінальних ЦЗ і, як наслідок, точність виявлення в середньому більше, ніж на 3.5%; зменшити обчислювальну складність і, як наслідок, часові витрати на експертизу ЦЗ в середньому більше, ніж на 15%, в порівнянні з [7].

Створено програмну реалізацію удосконаленого методу з командним інтерфейсом. Розроблений інтерфейс є зрозумілим і простим в користуванні. Програмний продукт дозволяє:

- провести аналіз цілої директорії з ЦЗ;
- перевірити цілісність одного конкретного ЦЗ.

Подальше покращення удосконаленого методу – це уточнення параметрів, що використовуються при алгоритмічній реалізації методу, для підвищення показника TP з наступним підвищенням точності виявлення порушення цілісності ЦЗ.

ПЕРЕЛІК ПОСИЛАНЬ

1. Кобозєва А.А., Маєвський Д.А., Кирилюк В.О. Метод виявлення порушення цілісності цифрового зображення, заснований на спектральному розкладанні симетризованої матриці блоку/ Збірник наукових праць Військового інституту Київського національного університету імені Тараса Шевченка. 2022. 77. С.66-81.
2. Информационное противоборство в современных условиях / Л.Г. Пирцхалава, В.А.Хорошко, Ю.Е.Хохлачева, М.Е.Шелест. К.: ЦП «Компринт», 2019. 226 с.
3. Uliyan, D.M., Jalab, H.A., Abdul Wahab, A.W., Sadeghi, S. Image region duplication forgery detection based on angular radial partitioning and Harris key-points / *Symmetry*. 2016. 8(7). 62.
4. Задірака,В.К. Сучасні методи розв'язання задач інформаційної безпеки / Вісник НАН України. 2014. 5. С. 65–69.
5. Mandal P.C., Mukherjee I., Paul G., Chatterji B.N. Digital image steganography: A literature survey / *Information Sciences*. 2022. 609. P. 1451–1488.
6. Борисенко, І.І. Виявлення цифрового фотомонтажу на основі аналізу контрастності зображення / Сучасний захист інформації. 2020. №2. С. 47–51.
7. Kobozeva,A.A., Bobok,I.I., Garbuz,A.I. General principles of integrity checking of digital images and application for steganalysis / *Transport and Telecommunication Journal*. 2016. 17(2). P. 128–137.
8. Joglekar,N.P., Chatur,P.N. A compressive survey on active and passive methods for image forgery detection /*International Journal of Engineering and Computer Science*. 2015. 4(1). P. 10187–10190.
9. Shwetha B., Sathyanarayana S.V. Digital image forgery detection techniques: a survey /*ACCENTS Transactions on Information Security*. 2017. 2(5). P. 22–31.
10. Thakur, T., Singh, K., Yadav, A. Blind approach for digital image forgery detection / *International Journal of Computer Applications*. 2018. 179(10). P. 34–42.

11. Chu X., Li, H. A Survey of Blind Forensics Techniques for JPEG Image Tampering / *Journal of Computer and Communications*. 2019. 7(10). P. 1–13.
12. Лебедева, Е.Ю., Кобозева, А.А. Основы метода выявления клонированных участков изображения, подвергнутых коррекции яркости / *Сучасна спеціальна техніка*. 2013. 3(34). С. 17–24.
13. Li, H., Luo, W., Qiu, X., Huang, J. Image forgery localization via integrating tampering possibility maps / *IEEE Transactions on Information Forensics and Security*. 2017. 12(5). P. 1240–1252.
14. Трифонова, К.О. Метод виявлення порушення цілісності цифрового зображення шумом Перліна / *Радіоелектроніка, інформатика, управління*. 2017. 2. С. 134–142.
15. Khan, S., Khan, K., Ali, F., Kwak, K.-S. Forgery detection and localization of modifications at the pixel level / *Symmetry*. 2020. 12(1). 137.
16. Al-Jarrah, M.M., Al-Taie, Z.H., Abuarqoub, A. Steganalysis using LSB-focused statistical features / *Proceedings of the International Conference on Future Networks and Distributed Systems (ICFNDS'17)*. 2017. Article 54. P.1–5.
17. Зоріло, В.В., Кіусєва, О.І., Зоріло, І.В. Модифікація алгоритму виявлення штучного підвищення різкості цифрового зображення / *Інформатика та математичні методи в моделюванні*. 2018. 8(2). С. 156–163.
18. Зоріло В.В., Карпова О.А. Алгоритм виявлення обробки зображення фільтром «Motion blur» / *Інформатика та математичні методи в моделюванні*. 2019. 9(1-2). С. 49–58.
19. Duan, X.T., Peng, T., Li, F.F., Wang, J. Blind separation of tampered images based on JPEG double compression properties / *Journal of University of Jinan (Science and Technology)*. 2017. 31. P. 87–96.
20. Берія Д.Ю., Войтовецька М.Є., Козаченко Н.Г., Зоріло В.В., Лебедева О.Ю. Виявлення порушень цілісності цифрового зображення в контексті цифрової криміналістики / *Інформатика та математичні методи в моделюванні*. 2021. 11(3). С. 181–190.

21. Бобок,І.І Розвиток загального підходу до проблеми виявлення порушень цілісності цифрових зображень / Правове, нормативне та метрологічне забезпечення системи захисту інформації в Україні. 2017. 2(34). С.78–88.
22. Lerch-Hostalot,D., Megias,D. Unsupervised steganalysis based on artificial training sets / Engineering Applications of Artificial Intelligence. 2016. 50. P.45–59.
23. Bobok,I.I. Steganalysis method for detection of the hidden communication channel with low capacity /Telecommunications and Radio Engineering. 2018. 77(18). P. 1597–1604.
24. Bobok,I.I., Kobozeva,A.A. Method for detecting of digital image integrity violations due to its block processing / Радіотехніка. 2019. 199. С. 130–141.
25. Geetha,S., Sindhu,S., Kamaraj,N. Close color pair signature ensemble adaptive threshold based steganalysis for LSB embedding in digital images /Transactions on Data Privacy. 2009. 1. P. 140–161.
26. Гонсалес,Р., Вудс,Р. Цифровая обработка изображений. М.: Техносфера, 2006. 1070 с.
27. Bergman, C., Davidson, J. Unitary embedding for data hiding with the SVD / Security, steganography and watermarking of multimedia contents VII, SPIE. 2005. 5681. P.619–630.
28. Деммель,Д. Вычислительная линейная алгебра: теория и приложения. М.: Мир, 2001. 430 с.
29. Кобозева,А.А. Основы общего подхода к разработке универсальных стеганоаналитических методов для цифровых изображений / Праці Одеського політехнічного університету. 2014. 2. С. 136–146.
30. Гантмахер,Ф.Р. Теория матриц: монография. 5-е изд. М.: Физматлит, 2004. 559 с.
31. Srinivas, R., Panda,S. Performance analysis of various filters for image noise removal in different noise environment / International Journal of Advanced Computer Research. 2013. 3. P.47–52.
32. Дьяконов В. П. MATLAB. Полный самоучитель. – М.: ДМК Пресс, 2012. – 768 с.: ил.