

Міністерство освіти і науки України
Національний університет «Одеська політехніка»
Українсько-польський інститут
Кафедра кібербезпеки та програмного забезпечення

Розумяк Богдан Олегович,
група ЛРЗ-171

КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА

Дослідження деяких розкладань і перетворень матриці цифрового зображення як теоретична основа методів виявлення порушень його цілісності (комплексна)

Дослідження деяких розкладань і перетворень матриці цифрового зображення як основа метода виявлення мультиплікативного шуму

Спеціальність:
125 Кібербезпека
Спеціалізація, освітня програма Кібербезпека

Керівник:
Зоріло Вікторія Вікторівна
к.т.н.

Одеса – 2022

Міністерство освіти і науки України
Національний університет «Одеська політехніка»
Українсько-польський інститут
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
_____ 2022р.

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

Розумяку Богдану Олеговичу

1. Тема роботи: *Дослідження деяких розкладань та перетворень матриці цифрового зображення як теоретична основа методів виявлення порушень його цілісності (комплексна). Дослідження деяких розкладань і перетворень матриці цифрового зображення як основа методу виявлення мультиплікативного шуму,*
керівник роботи *к.т.н. Зоріло Вікторія Вікторівна,*
затвержені наказом ректора від „_____” _____ 20__ р. № _____ .
2. Зміст роботи: *аналіз методів виявлення обробки зображень, розробка методу виявлення мультиплікативного шуму як порушення цілісності цифрового зображення, програмна реалізація розробленого методу.*
3. Перелік ілюстративного матеріалу: *рисунок клонування, рисунок фотомонтажу, рисунок узагальненої схеми виявлення сліпої підробки зображень, рисунок блок-схеми виявлення*

підробки зображень, рисунки спектрального розкладання, рисунки дискретно-косинусного перетворення, рисунки інтерфейсу програмного продукту, 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	Аналіз літератури за темою випускної кваліфікаційної роботи	29.08.2022	виконано
2	Обґрунтування вибору рішення та збір даних	15.09.2022	виконано
3	Розробка програмного забезпечення	01.10.2022	виконано
4	Тестування та перевірка програмного продукту	15.10.2022	виконано
5	Підготовка тексту роботи	01.11.2022	виконано
7	Підготовка презентації та доповіді	12.11.2022	виконано
8	Попередній захист	02.12.2022	виконано
9	Нормоконтроль, рецензування	15.12.2022	виконано
10	Занесення роботи в електронний архів	18.12.2022	виконано
11	Допуск до захисту	19.12.2022	виконано

Здобувач вищої освіти _____

Розумяк Б.О.

Керівник роботи _____

Зоріло В.В.

АНОТАЦІЯ

Кваліфікаційна робота на тему «Дослідження деяких розкладань та перетворень матриці цифрового зображення як теоретична основа методів виявлення порушень його цілісності (комплексна). Дослідження деяких розкладань і перетворень матриці цифрового зображення як основа метода виявлення мультиплікативного шуму» на здобуття другого (магістерського) рівня вищої освіти за спеціальністю 125 Кібербезпека, спеціалізація, освітня програма: Кібербезпека, містить 51 рисуноків, 1 таблицю, 1 додаток, 33 літературних джерел за переліком посилань. Робота виконана на 59 сторінках загального тексту і 47 сторінках основного тексту.

Мета – виявлення мультиплікативного шуму як порушення цілісності цифрового зображення шляхом розробки метода, заснованого на аналізі частотного спектру його матриці.

Проведено обчислювальний експеримент з використанням 200 цифрових зображень. На основі отриманих результатів розроблено та програмно реалізовано метод виявлення мультиплікативного шуму, ефективність якого в термінах помилок першого та другого роду складає 7% і 17% відповідно.

ЦИФРОВЕ ЗОБРАЖЕННЯ, ФАЛЬСИФІКАЦІЯ ЗОБРАЖЕНЬ, МЕТОДИ ОБРОБКИ ЗОБРАЖЕНЬ, МЕТОД ВИЯВЛЕННЯ ПІДРОБКИ ЦИФРОВИХ ЗОБРАЖЕНЬ.

ANNOTATION

The qualification work on the topic "Study of some decompositions and matrix transformations of a digital image as the basis of a method for detecting multiplicative noise" for obtaining the second (master's) level of higher education in the specialty 125 Cybersecurity, specialization, educational program: Cybersecurity, contains 51 figures, 1 table, 1 appendix, 33 literary sources according to the list of references. The work was completed on 59 pages of the general text and 47 pages of the main text.

The goal is to detect multiplicative noise as a violation of the integrity of a digital image by developing a method based on the analysis of the frequency spectrum of its matrix.

A computational experiment was conducted using 200 digital images. On the basis of the obtained results, a method of detecting multiplicative noise was developed and software implemented, the efficiency of which in terms of errors of the first and second kind is 7% and 17%, respectively.

DIGITAL IMAGE, IMAGE FORGERY, IMAGE PROCESSING METHODS,
DIGITAL IMAGE DETECTION METHOD.

ЗМІСТ

ВСТУП.....	7
1 АНАЛІЗ МЕТОДІВ ВИЯВЛЕННЯ ОБРОБКИ ЗОБРАЖЕНЬ.....	9
1.1 Методи виявлення підроблених зображень	9
1.2 Методи активної ідентифікації підробок.....	9
1.2.1. Цифровий підпис	9
1.2.2. Цифровий водяний знак	10
1.3. Пасивні методи виявлення підробок.....	10
1.3.1. Виявлення підробки копіювання та переміщення.....	11
1.3.2. Виявлення підробки з'єднання зображень	12
1.4 Узагальнена схема сліпого виявлення клонування зображень	13
1.5 Блок-схема виявлення підробки клонування зображень	14
2. РОЗРОБКА МЕТОДУ ВИЯВЛЕННЯ МУЛЬТИПЛІКАТИВНОГО ШУМУ ЯК ПОРУШЕННЯ ЦІЛІСНОСТІ ЦИФРОВОГО ЗОБРАЖЕННЯ	18
2.1 Спектральне розкладання.....	18
2.2 Дискретне косинусне перетворення.....	23
2.3 Розробка методу знаходження збурених зображень	29
3. ПРОГРАМНА РЕАЛІЗАЦІЯ РОЗРОБЛЕНОГО МЕТОДУ	35
3.1 Вибір мови програмування	35
3.2 Інтерфейс та робота програми	36
ВИСНОВОК.....	43
ПЕРЕЛІК ПОСЛАНЬ	44
Додаток А. Код програмного продукту	Ошибка! Закладка не определена.

ВСТУП

У сучасному світі технологій важливою складовою є цифрове зображення, яке використовується в багатьох сферах. Треба розуміти, що існує багато методів підробки або обробки цифрових зображень, які негативно впливають на сферу інформаційної безпеки, а саме приховують важливу, а, можливо, конфіденційну або таємну інформацію.

Із швидким розповсюдженням цифрової інформації, такої як цифрові відео, аудіо, фото, текстові документи тощо, у всіх сферах життя постало важливе питання про безпеку вмісту цієї інформації, зокрема, цифрових зображень, над чим працюють фахівці та вчені у всьому світі.

Маніпуляції із зображеннями можуть здійснюватись з різними цілями: отримання ліпшого зображення, приховання дефектів зйомки, вилучення з зображення якихось корисних даних, фальсифікація зображення з метою компрометування інших осіб, атака на зображення як на стеганографічний контейнер тощо [1]. Після виконання фальсифікації цифрове зображення часто піддають обробці різноманітними фільтрами для приховання слідів фальсифікації, тобто виконують постобробку.

Найвідоміші види підробки та постобробки цифрових зображень:

- накладання шуму (Гаусів, мультиплікативний, сіль-перець);
- розмиття (рівномірне (за Гаусом), «розумне» (тільки фонові частини), динамічне тощо);
- підвищення різкості (рівномірне або «розумне» (тільки контури) тощо);
- клонування (копіювання-переміщення в межах одного зображення);
- фотомонтаж (комбінація частин різних зображень).

Розвиток методів виявлення порушень цілісності цифрових зображень призводить до підвищення системи захисту інформації, тому тема роботи актуальна.

Мета – виявлення мультиплікативного шуму як порушення цілісності цифрового зображення шляхом розробки метода, заснованого на аналізі частотного спектру його матриці.

Для досягнення поставленої мети потрібно вирішити наступні задачі.

1. Ознайомлення з відомими методами фальсифікації зображень та огляд методів виявлення цих фальсифікацій.

2. Вибір параметрів цифрового зображення для аналізу.

3. Розробка методу виявлення мультиплікативного шуму як порушення цілісності цифрового зображення.

4. Реалізація програмного продукту.

Об'єкт дослідження – процес виявлення фальсифікації зображень.

Предмет дослідження – методи обробки цифрових зображень.

Методи дослідження:

- пошук та аналіз відкритих джерел інформації (наукові статті, публікації, електронні ресурси);
- порівняльний аналіз для розробленого методу;
- узагальнення для підведення підсумків дослідження та написання висновків.

Наукова новизна – вперше розроблено метод виявлення мультиплікативного шуму шляхом аналізу високочастотних коефіцієнтів ДКП.

Розроблений метод можна використовувати як складову комплексної системи захисту інформації на об'єктах інформаційної діяльності, як метод експертизи в лабораторіях з цифрової криміналістики, або як окремий додаток перевірки цілісності цифрових зображень.

Апробація дослідження. Результати представлено у публікації у фаховому журналі Інформатика та математичні методи в моделюванні [1].

1 АНАЛІЗ МЕТОДІВ ВИЯВЛЕННЯ ОБРОБКИ ЗОБРАЖЕНЬ

1.1 Методи виявлення підроблених зображень

Постійний розвиток методів виявлення підробок цифрових зображень – це те, що ми можемо спостерігати разом з розвитком інформаційних технологій [2].

Методи виявлення підробки зображень застосовуються для знаходження клонування і фотомонтажу. Для клонованих зображень, класифікація залежить від різновиду процесів вкладених малюнків з або без змін (трансформації) перед виділенням ознак зображення.

Для фотомонтажу узагальнено групи методів виявлення на основі характеристик зображення або характеристик фотокамери [3]. Ідентифікація підробки вказує на неможливість використання цифрового зображення в офіційних джерелах як свідоцтво якогось факту чи події.

Методи виявлення підробки зображень поділяються на дві групи: методи активної ідентифікації підробок; пасивні методи виявлення підробок.

1.2 Методи активної ідентифікації підробок

Для активного методу виявлення підробок потрібні попередньо витягнуті (вилучені) або вбудовані дані. Умовно методи, які використовуються для перевірки медіа, можна класифікувати на два види: на основі цифрового підпису і на основі водяних знаків [4-5].

1.2.1. Цифровий підпис

Цифровий підпис є одним із активних методів, які використовуються для виявлення підробки або фальсифікації зображення. Відображення автентичності цифрового документа за допомогою свого роду математичного формату називається цифровим підписом.

У цифровому підписі надійні біти витягуються з оригінального зображення. Воно розбивається на блоки розміром 16×16 пікселів. Секретний ключ k використовується для отримання N випадкових матриць із записами, рівномірно розподіленими в інтервалі $[0, 1]$.

Фільтр низьких частот часто застосовується до кожної випадкової матриці, щоб отримати N випадкової гладкої моделі [6]. Система виробляє цифровий підпис шляхом застосування процесу підпису до цифрового зображення.

1.2.2. Цифровий водяний знак

Водяні знаки також використовуються для виявлення підроблених зображень. У схемі контрольної суми можна додавати дані в останній старший біт пікселів. Послідовність лінійного регістра зсуву максимальної довжини до піксельних даних і таким чином визначається водяний знак шляхом обчислення функції просторової крос-кореляції послідовності, а також зображення з водяним знаком.

Ці водяні знаки розроблено таким чином, щоб їх неможливо було виявити або злити з природним шумом камери чи сканера. Також існують видимі водяні знаки. На додаток до цього також доступна візуально невизначена схема водяних знаків, яка може розпізнавати модифікацію в окремих пікселях і знаходити всюди, де відбувається модифікація [3].

Активні методи мають деякі недоліки, оскільки вимагають участі людини або спеціально обладнаних камер. Щоб подолати цей недолік, була запропонована пасивна автентифікація.

1.3. Пасивні методи виявлення підробок

Сліпі методи, відомі також як пасивні методи, використовують тільки, безпосередньо, зображення для його перевірки дійсності і цілісності [7]. Метод може «здогадуватись», хоч і видимих ознак фальсифікації зображення може не бути, фальсифікація може порушити базову статистичну властивість через неузгодженість

шуму, розмиття зображення, підвищення різкості зображення [8], підробку через клонування [9] та зображення, створені в графічних редакторах [10].

Існують техніки, що пропонуються для визначення лише певного виду підробок, наприклад фотомонтаж [11]. Є порушення цілісності цифрових зображень, які змінюють сцену оригіналу, тобто додають на фотографію об'єкти, яких там не було, або, навпаки, забирають їх. Інші види порушення цілісності не змінюють сцену, однак все одно вносять зміни до оригіналу, наприклад, такі порушення цілісності як розмиття, накладання шуму, зміна яскравості.

Дані види обробки цифрового зображення можуть застосовуватись як для приховання слідів клонування чи фотомонтажу, так і для здійснення стеганографічної атаки у випадку, якщо зображення використовують як контейнер для передачі стеганоповідомлення. У кожному випадку виявлення таких порушень цілісності свідчить про неможливість використання зображення як автентичного або про розкриття каналу передачі стеганоповідомлення з подальшою необхідністю вжити застережні заходи.

Пасивні методи виявлення підробки: виявлення підробки копіювання та переміщення (клонування); виявлення підробки з'єднання зображень (фотомонтаж).

1.3.1. Виявлення підробки копіювання та переміщення

Копіювання-переміщення (клонування) – це метод створення складеного зображення шляхом вирізання деякого об'єкта із зображення та додавання його до того самого зображення, див. на рис.1.1.

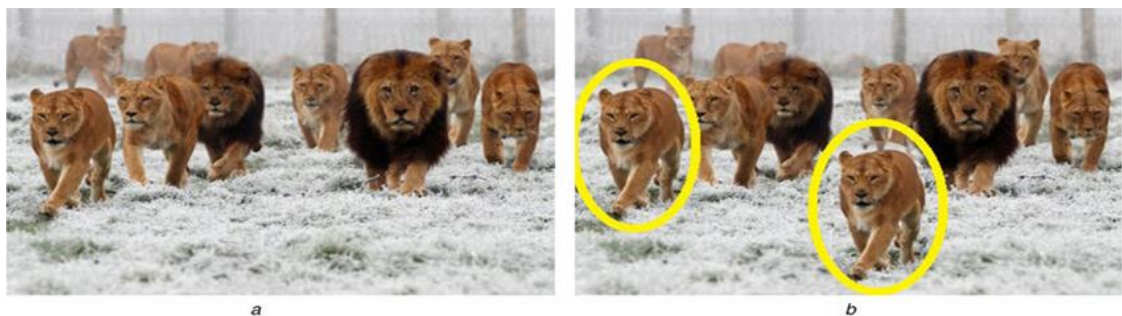


Рисунок 1.1 – Клонування

Методи виявлення підробки копіювання є трьох типів:

- метод грубої сили;
- блоковий метод;
- метод на основі ключових точок.

Метод грубої сили базується на методі безмежного пошуку та послідовної кореляції. У безмежному пошуку дане зображення застосовується для перевірки деякого сегмента з циклічно зміщеними версіями.

Даний метод використовує велику кількість порівнянь, тому, очевидно, що його обчислювальна непередбачуваність являється великою.

Підхід на основі блоків використовує такі алгоритми, як дискретне вейвлет-перетворення (DWT), аналіз основних компонентів (PCA), сингулярне розкладання (SVD) і дискретне косинусне перетворення (DCT) [12].

Метод на основі ключових точок використовує детектор і дескриптор інваріантних ознак масштабу та обертання, які є прискореними надійними функціями (SURF) і перетворенням інваріантних функцій масштабу (SIFT).

1.3.2. Виявлення підробки з'єднання зображень

Фотомонтаж – копіювання частини зображення та об'єднання його з іншим зображенням (з'єднанням зображень, рис.1.2).



Рисунок 1.2 – Фотомонтаж

Об'єднання зображень передбачає об'єднання хоча б декількох зображень для отримання підробленого зображення. Якщо комбінувати малюнки з контрастною

основою, то зробити межі і границі незв'язними, виявляється непростю задачею [13].

Цифровий фотомонтаж – це з'єднання зображень, при використанні різного програмного забезпечення, а саме графічних редакторів, таких як Adobe Photoshop.

1.4 Узагальнена схема сліпого виявлення клонування зображень

Основна ціль техніки сліпого виявлення підробки клонування – це класифікація підробки або оригінальності даного зображення. Схема процедури ідентифікації підробки зображення наведена на рис.1.3.

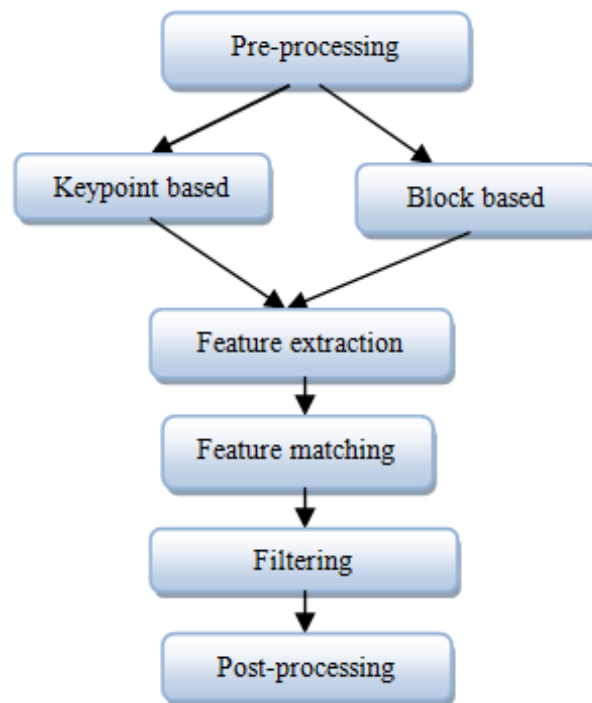


Рисунок 1.3 – Узагальнена схема сліпого виявлення підробки клонування зображень

Попередня обробка зображення: попередня обробка зображення – це перший пункт. Деякі попередні обробки виконуються на зображенні, яке розглядається, як-от фільтрація зображення, обрізка, зміна коефіцієнтів ДКП, перетворення RGB у градації сірого перед обробкою зображення для процедур виділення функцій [14].

Вилучення функцій: вибір функцій для кожного класу відокремлює набір зображень від різних класів, однак тим часом залишається постійним, призначеним для конкретного вибраного класу.

Привабливий елемент вибраного набору функцій полягає в тому, що він має крихітне вимірювання, щоб можна було зменшити обчислювальну складність і мати широку відмінність від інших класів [15].

Вибір класифікатора: залежно від набору функцій, який витягується під час вилучення ознак, вибирається або складається відповідний класифікатор. Великі навчальні набори дадуть покращену продуктивність класифікатора [16-17].

Класифікація: єдиною причиною класифікації є визначення оригінальності зображення. Для цього використовуються нейронні системи [18], LDA [19] і SVM [20].

Постобробка: деякі підробки, ймовірно, потребуватимуть постобробки, яка включає такі маніпуляції, як локалізація підроблених областей [21-22].

1.5 Блок-схема виявлення підробки клонування зображень

Блок-схема виявлення підробки клонування зображень наведена на рис.1.4. Вхідне зображення розділяється на різні блоки різної форми, що перекриваються, а потім відбувається вилучення ознак з кожного блоку.

Сортування виконується на основі ознак, щоб легко знайти область із такими ж ознаками, а решта вважається підробленими точками. І деяка морфологічна операція, щоб виявити підроблену область [23].

Дискретне вейвлет-перетворення (DWT), дискретне косинусне перетворення (DCT), принципівий компонентний аналіз (PCA), сингулярна декомпозиція (SVD), масштабно-інваріантне перетворення ознак (SIFT) і локально лінійне вбудовування (LLE) – це різні методи, які допомагають виявити підробку.

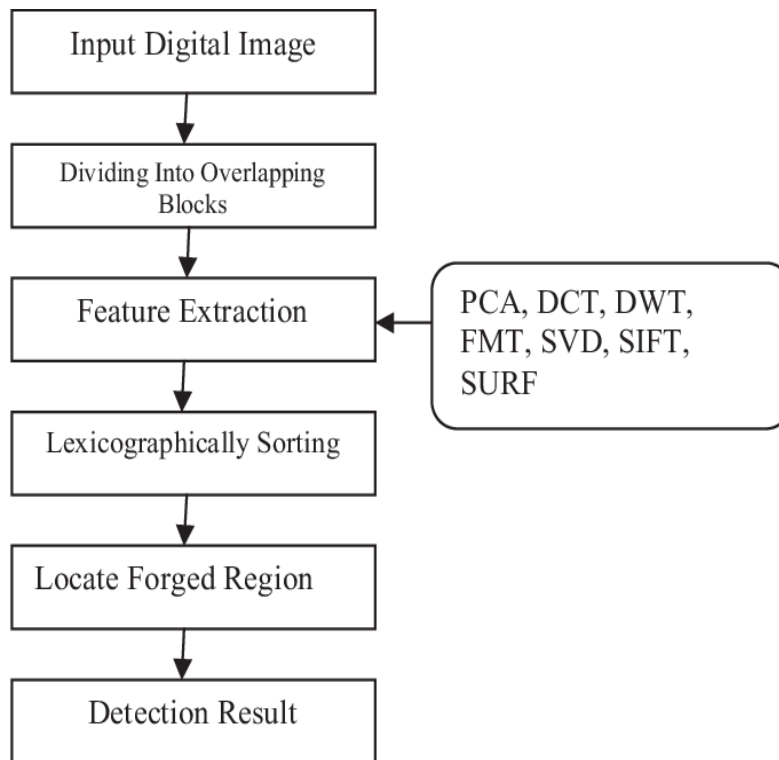


Рисунок 1.4 – Блок-схема виявлення підробки клонування зображень

Схема виявлення підробок клонування з використанням адаптивної сегментації та зіставлення ознак працює наступним чином. Зображення ділиться на блоки, які не перекриваються, за допомогою методу DWT.

Характеристики виділяються з блоків за допомогою методу SIFT, а для належного виявлення підробленої області застосовується морфологічна операція з низькими обчислювальними витратами та високою точністю та швидкістю відгуку. Також виявляються операції зменшення дискретизації, масштабування, обертання та стиснення JPEG. За допомогою цього підходу можна виявити фотомонтаж.

Дискретне косинусне перетворення є різновидом перетворення Фур'є і теж має зворотне перетворення. Графічне зображення можна розглядати як сукупність просторових хвиль, причому осі X і Y збігаються з шириною і висотою картинки, а осі Z відкладається значення кольору відповідного пікселя зображення.

Дискретне косинусне перетворення дозволяє переходити від просторового представлення картинки до її спектрального представлення, та у зворотньому

напрямку. Впливаючи на спектральне представлення картинки, що складаються з "гармонік", тобто, відкидаючи найменш значущі з них, можна балансувати між якістю відтворення і ступенем стиснення.

Дискретне косинусне перетворення перетворює матрицю пікселів розміром $N \times N$ на матрицю частотних коефіцієнтів відповідного розміру.

Однією з особливостей є те, що практично неможливо виконати дискретне косинусне перетворення для зображення відразу. Як розв'язання цієї проблеми група розробників JPEG запропонувала розбивати зображення на блоки розміром 8×8 пікселів.

За визначенням дискретного косинусного перетворення його реалізації потрібно два вкладених циклу, і тіло циклів буде виконуватися $N \times N$ раз кожного елемента матриці дискретного косинусного перетворення.

Значно ефективніший варіант обчислення коефіцієнтів дискретного косинусного перетворення реалізований через перемноження матриць.

При такому підході формула дискретного косинусного перетворення може бути записана так:

$$\text{ДКП} = \text{КП} \times \text{Р} \times \text{КПт.}$$

де:

- ДКП – дискретне косинусне перетворення;
- КП – матриця косинусного перетворення розміром $N \times N$;
- Р – матриця розміром $N \times N$, що складається зі значень яскравості пікселів зображення;
- КПт – транспонована матриця КП [24].

У 2004 році Д. Лоу, Університет Британської Колумбії, запропонував новий алгоритм Scale Invariant Feature Transform (SIFT) у своїй статті «Відмітні ознаки зображення з масштабно-інваріантних ключових точок», який витягує ключові точки та обчислює їх дескриптори. SIFT – це незмінність до масштабу та обертання зображення.

Алгоритм SIFT складається з п'яти кроків:

1. Вибір масштабно-просторового піку: потенційне розташування для пошуку функцій.
2. Локалізація ключових точок: точне визначення місцезнаходження ключових точок функції.
3. Призначення орієнтації: призначення орієнтації ключовим точкам.
4. Дескриптор ключових точок: Опис ключових точок як багатовимірний вектор.
5. Зіставлення ключових точок.

Основними перевагами SIFT є наступні.

Локальність: функції локальні, настільки стійкі до оклюзії та безладу (без попередньої сегментації).

Відмінність: окремі ознаки можна зіставити з великою базою даних об'єктів.

Кількість: навіть для невеликих об'єктів можна створити багато функцій.

Ефективність: близька до продуктивності в реальному часі.

Розширюваність: можна легко розширити до широкого діапазону різних типів функцій, кожна з яких додає надійність [25].

В даному розділі зроблено огляд сучасного стану виявлення різних видів порушення цілісності цифрових зображень. Встановлено, що виявленню пост обробки зображення у відкритому друці приділяється мало уваги. Зокрема, виявлення мультиплікативного шуму як порушення цілісності цифрового зображення – питання досі не вирішене до кінця. Розглянемо в наступному розділі вплив мультиплікативного шуму на зображення.

2. РОЗРОБКА МЕТОДУ ВИЯВЛЕННЯ МУЛЬТИПЛІКАТИВНОГО ШУМУ ЯК ПОРУШЕННЯ ЦІЛІСНОСТІ ЦИФРОВОГО ЗОБРАЖЕННЯ

2.1 Спектральне розкладання

Нехай A – квадратна підматриця матриці яскравості пікселів цифрового зображення. Надзвичайно важливим для виявлення порушень цілісності цифрових зображень є матричне розкладання квадратної матриці A на так звані власні значення та власні вектори. Це розкладання зазвичай називається «діагоналізація матриці». Однак ця назва не є оптимальною, оскільки процес, який описується, насправді є розкладанням матриці на добуток трьох інших матриць, лише одна з яких є діагональною, а також тому, що всі інші стандартні типи розкладу матриці використовують термін «розкладання» в їхніх назвах, наприклад, розкладання Холеського, розкладання Гессенберга, і так далі. Розкладання матриці на матриці, що складаються з її власних векторів і власних значень, у цій роботі називається спектральним розкладанням (власним розкладом).

Припустімо, матриця A розміром $k \times k$ має невироджені власні значення $\lambda_1, \lambda_2, \dots, \lambda_k$ та відповідні лінійно незалежні власні вектори X_1, X_2, \dots, X_k , які можна позначити так (див. рис.2.1).

$$\begin{bmatrix} x_{11} \\ x_{12} \\ \vdots \\ x_{1k} \end{bmatrix}, \begin{bmatrix} x_{21} \\ x_{22} \\ \vdots \\ x_{2k} \end{bmatrix}, \dots, \begin{bmatrix} x_{k1} \\ x_{k2} \\ \vdots \\ x_{kk} \end{bmatrix}.$$

Рисунок 2.1 – Лінійно незалежні власні вектори

Визначення матриці, що складаються з власних векторів показано на рис.2.2 та рис.2.3.

$$[X_1 \ X_2 \ \dots \ X_k]$$

Рисунок 2.2 – Матриця власних векторів

$$\begin{bmatrix} x_{11} & x_{21} & \cdots & x_{k1} \\ x_{12} & x_{22} & \cdots & x_{k2} \\ \vdots & \vdots & \ddots & \vdots \\ x_{1k} & x_{2k} & \cdots & x_{kk} \end{bmatrix}$$

Рисунок 2.3 - Матриця власних векторів

Визначення матриці власних значень, де D – діагональна матриця, відображена на рисунку 2.4.

$$D \equiv \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_k \end{bmatrix},$$

Рисунок 2.4 – Матриця власних значень

Потім робимо розкладання A на перетворення подібності за участі P (власні вектори) і D , яке зображено на рисунку 2.5 і отримуємо спектральне розкладання (власний розклад), який показано на рисунку 2.6

$$\begin{aligned} &= A[\mathbf{X}_1 \ \mathbf{X}_2 \ \cdots \ \mathbf{X}_k] \\ &= [A\mathbf{X}_1 \ A\mathbf{X}_2 \ \cdots \ A\mathbf{X}_k] \\ &= [\lambda_1 \mathbf{X}_1 \ \lambda_2 \mathbf{X}_2 \ \cdots \ \lambda_k \mathbf{X}_k] \\ &= \begin{bmatrix} \lambda_1 x_{11} & \lambda_2 x_{21} & \cdots & \lambda_k x_{k1} \\ \lambda_1 x_{12} & \lambda_2 x_{22} & \cdots & \lambda_k x_{k2} \\ \vdots & \vdots & \ddots & \vdots \\ \lambda_1 x_{1k} & \lambda_2 x_{2k} & \cdots & \lambda_k x_{kk} \end{bmatrix} \\ &= \begin{bmatrix} x_{11} & x_{21} & \cdots & x_{k1} \\ x_{12} & x_{22} & \cdots & x_{k2} \\ \vdots & \vdots & \ddots & \vdots \\ x_{1k} & x_{2k} & \cdots & x_{kk} \end{bmatrix} \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_k \end{bmatrix} \\ &= PD. \end{aligned}$$

Рисунок 2.5 – Розкладання A на перетворення подібності за участі P і D

$$A = PDP^{-1}.$$

Рисунок 2.6 – Власний розклад

На рисунку 2.6 використано такі позначення: P – матриця власних векторів заданої квадратної матриці A , D – діагональна матриця з відповідними власними значеннями на діагоналі.

Тоді, доки P є квадратною матрицею A , її можна записати як власне розкладання, в якому D – діагональна матриця з власними значеннями на головній діагоналі. Крім того, якщо A є симетричною, то стовпці P є ортогональними векторами.

Якщо P не є квадратною матрицею (наприклад, простір власних векторів $[1 \ 1; 0 \ 1]$ є одновимірним), то P не може мати оберненої матриці та A не може мати розкладання на власні ланки [26]. Той факт, що це розкладання завжди можливе для квадратної матриці A , якщо P є квадратною матрицею, відомий як теорема про власний розклад (див. рис.2.6). Спектральне розкладання для загальних додатних цілих степенів зображений на рисунку 2.7.

$$A^n = P D^n P^{-1}.$$

Рисунок 2.7 – Спектральне розкладання для додатних цілих степенів
Обернене розкладання до матриці A є (див. рис.2.8).

$$\begin{aligned} &= (P D P^{-1})^{-1} \\ &= P D^{-1} P^{-1}, \end{aligned}$$

Рисунок 2.8 – Обернене розкладання до матриці A
Обернена матриця D визначається як показано на рисунку 2.9.

$$D^{-1} = \begin{bmatrix} \lambda_1^{-1} & 0 & \dots & 0 \\ 0 & \lambda_2^{-1} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \lambda_k^{-1} \end{bmatrix}.$$

Рисунок 2.9 – Обернена матриця D

Отже, рівняння виконується як для негативних, так і для позитивних значень [27]. Для знаходження власних значень та власних векторів у мові програмування Matlab використовується функція `eig`. Її синтаксис виглядає так:

```

e = eig(A)
[V, D] = eig(A)
[V, D, W] = eig(A)
e = eig (A, B)
[V, D] = eig (A, B)
[V, D, W] = eig (A, B)
[___] = eig (A, balanceOption)
[___] = eig (A, B, algorithm)
[___] = eig (___, outputForm)

```

Для більш детального розуміння ознайомимось з описом та основними прикладами роботи даної функції:

`e = eig (A)` повертає вектор-стовпець, що містить власні значення квадратної матриці A . Приклад роботи даної функції зображений на наступних рисунках (див. рис. 2.10-2.12)

```

A = gallery('lehmer',4)

A = 4x4

    1.0000    0.5000    0.3333    0.2500
    0.5000    1.0000    0.6667    0.5000
    0.3333    0.6667    1.0000    0.7500
    0.2500    0.5000    0.7500    1.0000

```

Рисунок 2.10 – Симетрично-позитивна визначена матриця

Використовується функція `gallery` для створення симетричної позитивно визначеної матриці. Обчислення власних значень матриці A показано на рисунку 2.11. Результатом є вектор-стовпець.

```

e = eig (A)

e = 4x1

    0,2078
    0,4078
    0,8482
    2,5362

```

Рисунок 2.11 – Власні значення матриці A

Як альтернативу, можна використати `outputForm` для повернення власних значень у діагональній матриці, як зображено на рисунку 2.12.

```
D = eig(A, 'matrix')
D = 4x4
    0.2078      0      0      0
         0    0.4078      0      0
         0      0    0.8482      0
         0      0      0    2.5362
```

Рисунок 2.12 – Відображення власних значень у діагональній матриці

$[V, D] = \text{eig}(A)$ повертає діагональну матрицю D власних значень і матрицю V , стовпці якої є відповідними правими власними векторами, так що $A \times V = V \times D$. Обчислення власних значень та правих власних векторів матриці A відображено на рисунку 2.13.

```
[V,D] = eig(A)
V = 3x3 complex
-0.5774 + 0.0000i    0.2887 - 0.5000i    0.2887 + 0.5000i
-0.5774 + 0.0000i   -0.5774 + 0.0000i   -0.5774 + 0.0000i
-0.5774 + 0.0000i    0.2887 + 0.5000i    0.2887 - 0.5000i

D = 3x3 complex
 6.0000 + 0.0000i    0.0000 + 0.0000i    0.0000 + 0.0000i
 0.0000 + 0.0000i   -1.5000 + 0.8660i    0.0000 + 0.0000i
 0.0000 + 0.0000i    0.0000 + 0.0000i   -1.5000 - 0.8660i
```

Рисунок 2.13 – Власні значення та праві власні вектори матриці A

$[V, D, W] = \text{eig}(A)$ також повертає повну матрицю W , стовпці якої є відповідними лівими власними векторами, так що $W' \times A = D \times W'$.

Проблема власних значень полягає у визначенні розв'язку рівняння $A v = \lambda v$, де A – $n \times n$ матриця, v – вектор-стовпець довжини n , а λ – скаляр. Значення λ , які задовольняють рівняння, є власними значеннями.

Відповідні значення v , які задовольняють рівняння, є правими власними векторами. Ліві власні вектори w задовольняють рівняння $w' A = \lambda w'$.

Обчислення правих власних векторів V , власних значень, D і лівих власних векторів W зображено на рисунку 2.14 [28].

```
[V,D,W] = eig (A)

V = 3x3

-0,2610 -0,9734 0,1891
-0,5870 0,2281 -0,5816
-0,7663 -0,0198 0,7912

D = 3x3

25,5548 0 0
0 -0,5789 0
0 0 -7,9759

W = 3x3

-0,1791 -0,9587 -0,1881
-0,8127 0,0649
-0,7477 -0,5545 0,2768 0,6368
```

Рисунок 2.14 – Обчислення правих власних векторів, власних значень і лівих власних векторів

2.2 Дискретне косинусне перетворення

Дискретне косинусне перетворення (ДКП) представляє зображення як суму синусоїд різної величини та частоти. Ця `dct2` функція в Matlab обчислює двовимірне дискретне косинусне перетворення (ДКП) зображення. ДКП має властивість, що для типового зображення більша частина візуально значущої інформації про зображення зосереджена лише в кількох коефіцієнтах ДКП.

З цієї причини ДКП часто використовується в програмах стиснення зображень. Наприклад, ДКП лежить в основі міжнародного стандартного алгоритму стиснення зображень із втратами даних, відомого як JPEG (назва походить від робочої групи, яка розробила стандарт: Joint Photographic Experts Group).

Обчислення двовимірного ДКП $M \times N$ матриці A обчислюється наступним чином і зображена на рисунку 2.15.

$$B_{pq} = \alpha_p \alpha_q \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} A_{mn} \cos \frac{\pi(2m+1)p}{2M} \cos \frac{\pi(2n+1)q}{2N}, \quad \begin{matrix} 0 \leq p \leq M-1 \\ 0 \leq q \leq N-1 \end{matrix}$$

$$\alpha_p = \begin{cases} 1/\sqrt{M}, & p=0 \\ \sqrt{2/M}, & 1 \leq p \leq M-1 \end{cases} \quad \alpha_q = \begin{cases} 1/\sqrt{N}, & q=0 \\ \sqrt{2/N}, & 1 \leq q \leq N-1 \end{cases}$$

Рисунок 2.15 – Формула двовимірного ДКП матриці A із розміром $M \times N$

Значення B_{pq} називаються ДКТ коефіцієнтами A (індекси матриці в Matlab завжди починаються з 1, а не з 0; отже, елементи матриці Matlab $A(1,1)$ і $B(1,1)$ відповідають математичним величинам A_{00} і B_{00} відповідно).

ДКП є оборотним перетворенням, його знаходження показано на рисунку 2.16.

$$A_{mn} = \sum_{p=0}^{M-1} \sum_{q=0}^{N-1} \alpha_p \alpha_q B_{pq} \cos \frac{\pi(2m+1)p}{2M} \cos \frac{\pi(2n+1)q}{2N}, \quad \begin{matrix} 0 \leq m \leq M-1 \\ 0 \leq n \leq N-1 \end{matrix}$$

$$\alpha_p = \begin{cases} 1/\sqrt{M}, & p=0 \\ \sqrt{2/M}, & 1 \leq p \leq M-1 \end{cases} \quad \alpha_q = \begin{cases} 1/\sqrt{N}, & q=0 \\ \sqrt{2/N}, & 1 \leq q \leq N-1 \end{cases}$$

Рисунок 2.16 – Обернене перетворення ДКП

Обернене рівняння ДКП можна інтерпретувати так, що будь-яка матриця A розміру M на N може бути записана як сума функцій MN , дане рівняння зображено на рисунку 2.17.

$$\alpha_{стор} \alpha_q \cos \frac{\pi(2m+1)стор}{2M} \cos \frac{\pi(2n+1)q}{2n}, \quad \begin{matrix} 0 \leq p \leq M-1 \\ 0 \leq q \leq N-1 \end{matrix}$$

Рисунок 2.17 – Інтерпретоване обернене рівняння ДКП

Всі вище перераховані функції називаються базисними функціями ДКП. Таким чином, ДКП коефіцієнти B_{pq} можна розглядати як «вага», застосовані до кожної базисної функції. Для матриць 8×8 на рисунку 2.18, показано 64 базисні функції.

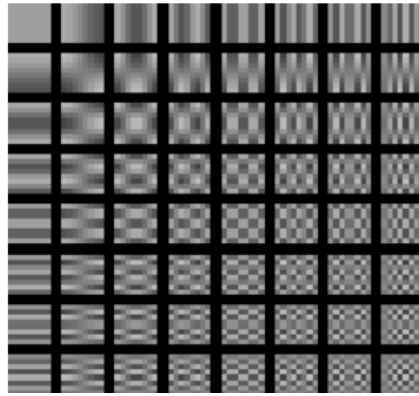


Рисунок 2.18 – 64 базисні функції матриці 8×8

Горизонтальні частоти збільшуються зліва направо, а вертикальні – зверху вниз. Базисна функція зі постійним значенням у верхньому лівому куті часто називають базисною функцією ДК, а відповідний коефіцієнт ДКП B_{00} в основному – коефіцієнтом ДК. Існує два способи обчислення ДКП за допомогою програмного забезпечення Image Processing Toolbox. Перший спосіб полягає у використанні `dct2` функції. `dct2` використовує алгоритм на основі ШПФ (швидкого перетворення Фурє) для швидкого обчислення з великими вхідними даними.

Другий метод полягає у використанні матриці перетворення ДКП, яка повертається функцією `dctmtx` та може бути більш ефективною для малих квадратних вхідних даних, таких як 8×8 або 16×16 . Матриця перетворення T розміром $M \times M$ визначається як показано на рисунку 2.19.

$$T_{pq} = \begin{cases} \frac{1}{\sqrt{M}} & p = 0, & 0 \leq q \leq M - 1 \\ \sqrt{\frac{2}{M}} \cos \frac{\pi(2q + 1)ctop}{2M} & 1 \leq p \leq M - 1, & 0 \leq q \leq M - 1 \end{cases}$$

Рисунок 2.19 – Матриця перетворення $M \times M$

Для матриці A , $T \times A$ розміру $M \times M$ – це матриця $M \times M$, стовпці якої містять одновимірні ДКП стовпців матриці A . Двовимірний ДКП матриці A можна обчислити як $B = T \times A \times T'$. Оскільки матриця T це дійсна ортонормована матриця, її інверсія така сама, як і її транспонування. Отже, обернена двовимірна ДКП матриці B задається як $T' \times B \times T$. За допомогою дискретного косинусного перетворення, як

уже стало відомим, можна стиснути зображення. У наступному прикладі, яке зображено на рисунку 2.20-2.23 обчислюється двовимірний ДКП блоків 8×8 у вхідному зображенні, відкидаються (робляться нулі) усі, крім 10 із 64 коефіцієнтів ДКП у кожному блоці, а потім реконструюється зображення за допомогою двовимірного зворотного ДКП кожного блоку. Для початку необхідно прочитати зображення, записати в робочу область (зміну) і перетворити його на клас `double`, як зображено на рисунку 2.20.

```
I = imread( 'cameraman.tif' );
I = im2double(I);
```

Рисунок 2.20 – Зчитування зображення і перетворення у клас `double`

Далі, треба обчислити двовимірний ДКП блоків 8×8 у зображенні. Функція `dctmtx` повертає матрицю перетворення ДКП $N \times N$.

```
T = dctmtx(8);
dct = @(block_struct) T * block_struct.data * T';
B = blockproc(I,[8 8],dct);
```

Рисунок 2.21 – Обчислення ДКК блоків 8×8 у зображенні

Наступне – це відкидання всіх коефіцієнтів ДКП у кожному блоці крім 10 із 64, як зображено на рисунку 2.22.

```
mask = [1  1  1  1  0  0  0  0
        1  1  1  0  0  0  0  0
        1  1  0  0  0  0  0  0
        1  0  0  0  0  0  0  0
        0  0  0  0  0  0  0  0
        0  0  0  0  0  0  0  0
        0  0  0  0  0  0  0  0
        0  0  0  0  0  0  0  0];
B2 = blockproc(B,[8 8],@(block_struct) mask .* block_struct.data);
```

Рисунок 2.22 – Відкидання непотрібних коефіцієнтів ДКП у кожному блоці

Реконструюємо зображення за допомогою двовимірного зворотного ДКП кожного блоку, як зображено на рисунку 2.23.

```
invdct = @(block_struct) T' * block_struct.data * T;
I2 = blockproc(B2,[8 8],invdct);
```

Рисунок 2.23 – Реконструкція зображення

Хоча на реконструйованому зображенні є деяка втрата якості, вона чітко помітна, навіть незважаючи на те, що майже 85% коефіцієнтів ДКП було відкинуто. Оригінальне зображення показано на рисунку 2.24, а реконструйоване зображення представлено на рисунку 2.25 [29].



Рисунок 2.24 – Оригінальне зображення



Рисунок 2.25 – Реконструйоване зображення

Для виконання ДКП у Matlab, використовується функція `dct`. Її синтаксис виглядає так:

```
y = dct (x)
y = dct (x, n)
y = dct (x, n, dim)
y = dct (___, 'Type', dcttype)
```

Для кращого розуміння необхідно ознайомитись з описом і основними прикладами цієї функції: `y = dct (x)` повертає унітарне дискретне косинусне перетворення вхідного масиву `x`. Вихід `y` має такий самий розмір, як `x`. Якщо `x` має більше одного виміру, то `dct` працює вздовж першого виміру масиву з розміром

більше 1. Наприклад, необхідно знайти скільки коефіцієнтів ДКП представляють 99% енергії в послідовності, зміні та умова зображена на рисунку 2.26.

```
x = (1:100) + 50*cos((1:100)*2*pi/40);
X = dct(x);
[XX,ind] = sort(abs(X),'descend');
i = 1;
while norm(X(ind(1:i)))/norm(X) < 0.99
    i = i + 1;
end
needed = i;
```

Рисунок 2.26 – Зміні та цикл з умовою

Далі, необхідно відновити сигнал і порівняти його з вихідним сигналом, як показано на рисунку 2.27.

```
X(ind(needed+1:end)) = 0;
xx = idct(X);

plot([x;xx])
legend('Original', ['Reconstructed, N = ' int2str(needed)], ...
      'Location','SouthEast')
```

Рисунок 2.27 – Відновлення сигналу і зрівняння його з вихідним

В результаті отримаємо графік, який зображено на рисунку 2.28.

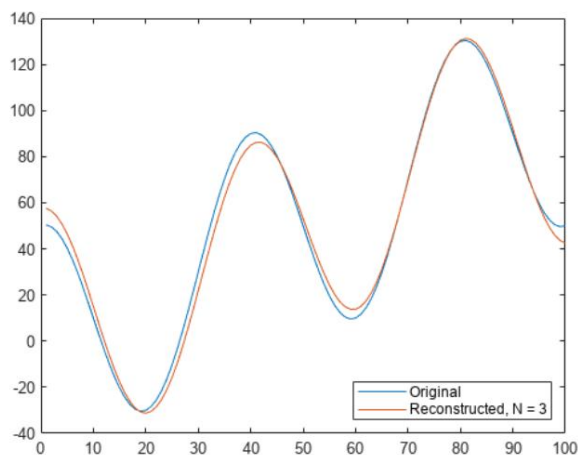


Рисунок 2.28 – Коефіцієнти ДКП, представляють 99% енергії в послідовності $y = \text{dct}(x, n, \text{dim})$ обчислює перетворення вздовж розміру dim . Щоб ввести розмір і використати значення за замовчуванням n , необхідно вказати другий аргумент порожнім [30].

2.3 Розробка методу знаходження збурених зображень

Зображення має різні частоти такі як низькі (фон), середні, високі (контур).

Проаналізувавши та розібравшись з роботою спектрального розкладання та ДКП, було вирішено проаналізувати збурення власних значень, після мультиплікативного шуму при спектральному розкладанні, а також подивитися на високочастотні коефіцієнти ДКП, так як мультиплікативний шум впливає на них. Високі частоти знаходяться в нижньому правому куті матриці. Також, необхідно експериментально виділити порогові значення для знаходження підробки, цим саме покращити роботу з дискретно-косинусним перетворенням. Для експерименту зі спектральним розкладанням було взято 100 зображень у форматі без втрат. Кожному зображенню поставимо у відповідність середні арифметичні значення власних значень з 1 по 7 по всім блокам матриці (див. рис. 2.29), отримані 7 чисел з'єднаємо лінією червоного кольору для оригінальних зображень, та синього для оброблених мультиплікативним шумом з дисперсією 0.1. З рисунка легко бачити, що відбувається суттєве збурення власних значень: менші стають ще меншими, більші – більшими. Проте є велика кількість зображень, які можна віднести до невизначеної групи, тобто важко знайти порогове значення, яке б дозволило чітко відділити оброблені зображення від необроблених.

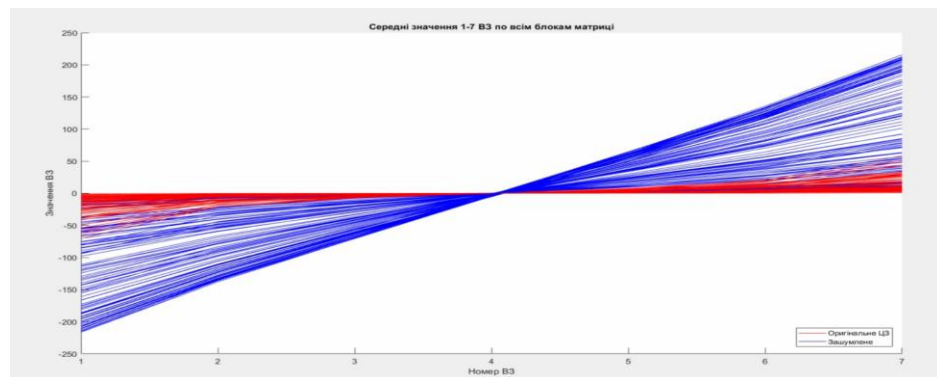


Рисунок 2.29 – Середні значення 1-7 ВЗ по всіх блоках матриці оригінальних (червоні лінії) та збурених зображень (сині лінії)

Подивимось, як впливає мультиплікативний шум на коефіцієнти ДКП. Для експерименту візьмемо 100 зображень формату tif, результат показаний на рисунку 2.30. Розіб'ємо зображення на блоки 8×8 , знайдемо для кожного блоку коефіцієнти ДКП, виділимо лише 9 коефіцієнтів, що відповідають високим частотам, знайдемо середні арифметичні значення виділених коефіцієнтів та поєднаємо їх інтерполяційним сплайном червоного кольору для оригінальних зображень і синього кольору для зображень, оброблених мультиплікативним шумом.

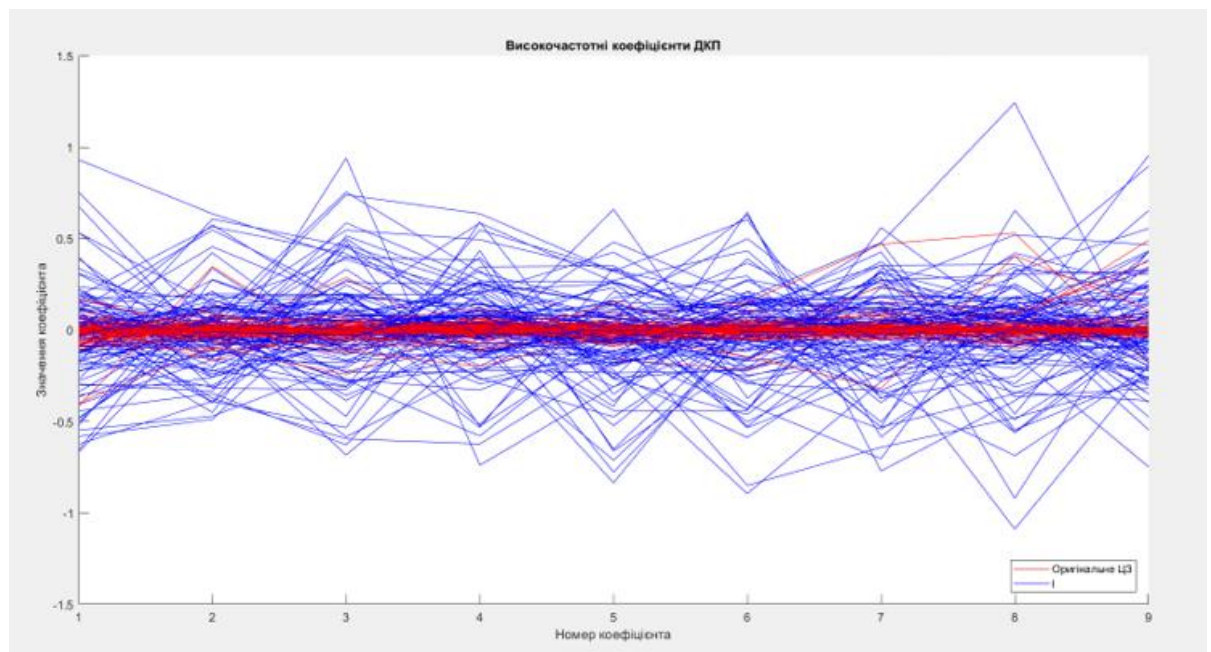


Рисунок 2.30 – Середні значення високочастотних коефіцієнтів ДКП оригінальних (червоні лінії) та збурених зображень (сині лінії)

Експеримент показав, що при застосуванні мультиплікативного шуму аналізовані коефіцієнти ДКП змінюються настільки, що стає можливим відділити оброблені зображення від необроблених. Знайдемо порогове значення для відокремлення оброблених зображень від необроблених на основі отриманих під час експерименту результатів. На рисунку 2.31 зображено гістограму амплітуд середніх значень високих частот блоку 8×8 для оригінальних зображень, на рисунку 2.32 — зображено гістограму амплітуд середніх значень високих частот для блоку 8×8 зашумлених зображень.

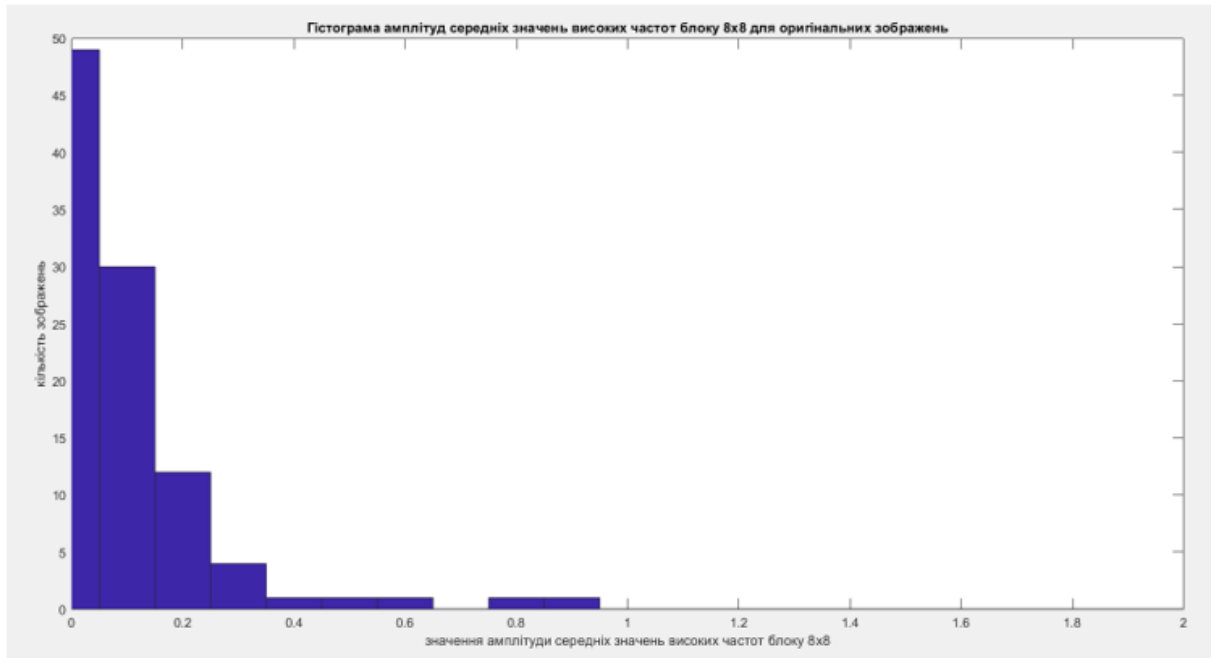


Рисунок 2.31 – Гістограма амплітуд середніх значень високих частот для блоку 8×8 оригінальних зображень.

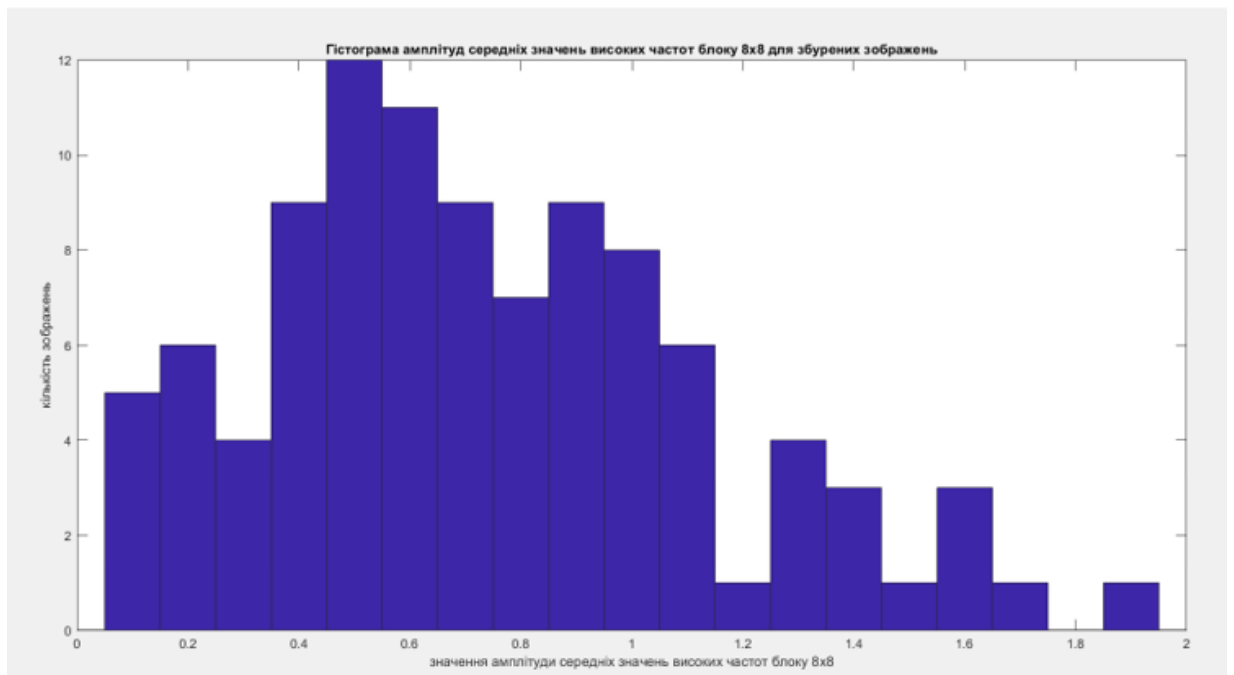


Рисунок 2.32 – Гістограма амплітуд середніх значень високих частот для блоку 8×8 зашумлених зображень.

Далі необхідно знайти середні значення амплітуд високих частот в кожному блоці 8×8 для оригінальних та збурених зображень, які показані на рисунку 2.33 та рисунку 2.34.

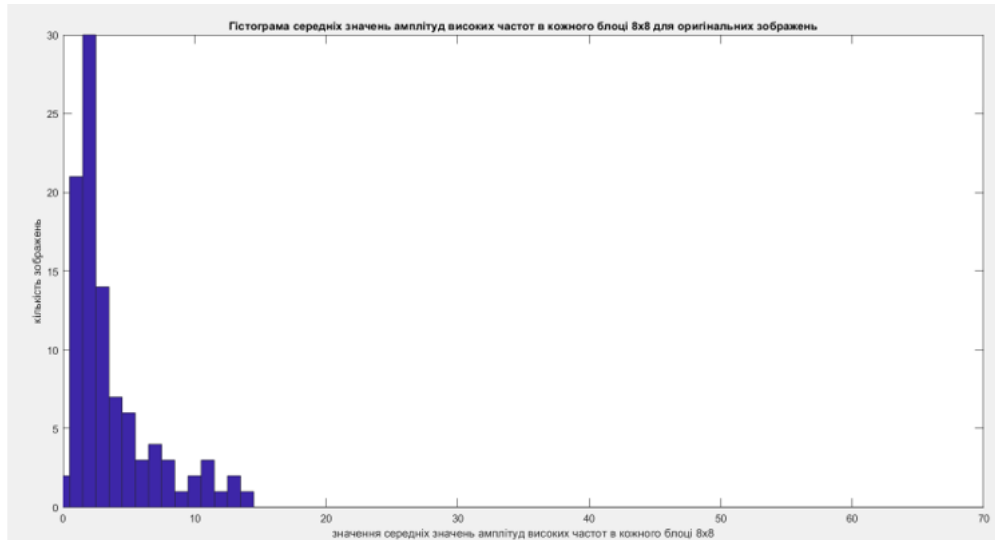


Рисунок 2.33 – Гістограма середніх значень амплітуд високих частот в кожному блоці 8×8 для оригінальних зображень

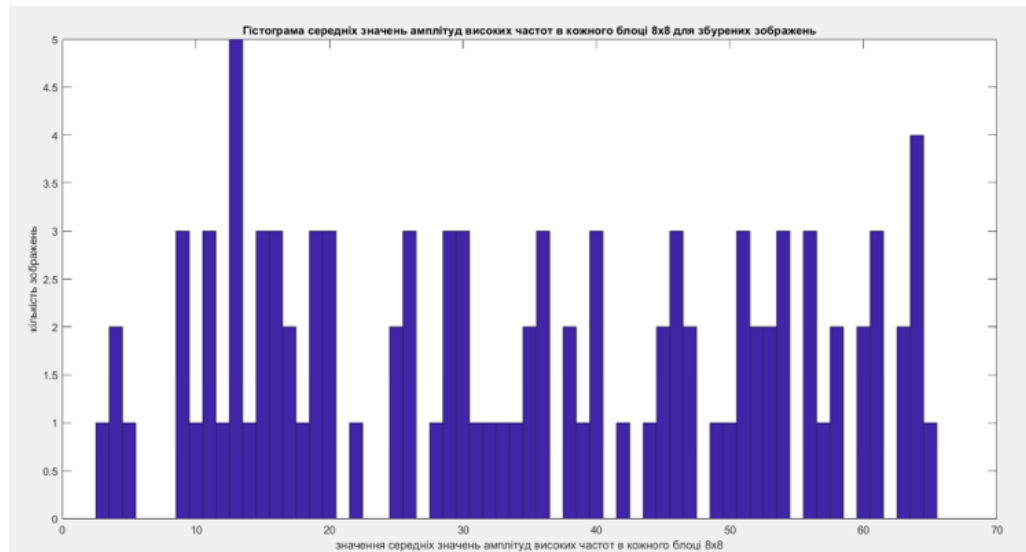


Рисунок 2.34 – Гістограма середніх значень амплітуд високих частот в кожному блоці 8×8 для зашумлених зображень

Такі ж характеристики було отримано для зображень, оброблених мультиплікативним шумом з дисперсією 0.001 та 0.005. Візуально ми бачимо, що для оброблених зображень коефіцієнти ДКП значно більші, ніж для необроблених,

тому задача полягає в тому, щоб визначити порогове значення для виявлення мультиплікативного шуму. Для цього потрібно отримати усереднені показники, які можна поставити у відповідність зображенню і використати в якості порогового значення.

Розглянемо коефіцієнти ДКП, вони можуть бути додатні і від’ємні. Знайдемо середні арифметичні значення для високочастотних коефіцієнтів ДКП і знайдемо модуль різниці між максимальним та мінімальним значенням – показник 1. Розглянемо також як альтернативу показник 2: спочатку знайдемо модуль різниці максимального та мінімального значення високочастотних коефіцієнтів в блоках матриці цифрового зображення, після чого знайдемо середнє арифметичне отриманих різниць. Будемо використовувати дані показники в якості порогового значення. Експериментально встановлено, що певні значення даних показників дійсно дозволяють розрізняти оброблені зображення від необроблених. В таблиці 2.1 показано результати їх використання.

Таблиця 2.1 – Результати розробленого методу

Збурення (величин а дисперсії)	АСС		Помилки I роду		Помилки II роду		Порогове значення	
	Показни к №1	Показни к №2	Показни к №1	Показни к №2	Показни к №1	Показни к №2	Показни к №1	Показни к №2
0.001	0.89	0.855	5%	14%	17%	15%	0.1457	6.3424
0.005	0.88	0.935	7%	4%	17%	9%	0.1457	9.1088
0.01	0.955	0.965	4%	4%	5%	3%	0.2378	11.8751

Як можемо бачити, дані показники дають різні результати за різних параметрах мультиплікативного шуму. Однак за характеристикою АСС Показник 1 кращий. Характеристика АСС (Accuracy of CC) – кількісний показник ефективності. Він визначає точність виявлення порушення цілісності. Формула розрахунку АСС $АСС = (TP + TN)/(TP + FN + TN + FP)$, де TP (True Positive) – число правильно виявлених ЦЗ, цілісність яких була порушена (істинопозитивний результат);

TN (True Negative) – число правильно виявлених оригінальних ЦЗ (істинонегативний результат);

FP (False Positive) – число оригінальних ЦЗ, помилково прийнятих за такі, цілісність яких була порушена (хибнопозитивний результат (хибна тривога) або помилка II роду);

FN (False Negative) – число ЦЗ, цілісність яких була порушена, помилково визнаних оригінальними (хибнонегативний результат або помилка I роду).

Ефективність розробленого методу по виявленню мультиплікативного шуму була перевірена по даному показнику та показана у таблиці 2.1 [31]. Крім того на даному етапі досліджень, коли експеримент було проведено для мультиплікативного шуму x дисперсією 0,001, 0,005 та 0,01 і для кожної дисперсії отримано різні значення Показника 1 (далі П1), рекомендується включити в алгоритм всі три випадки. Розробка універсального порогового значення є предметом подальших досліджень. На основі отриманих результатів розроблено метод виявлення мультиплікативного шуму, основні кроки алгоритму якого наступні.

Нехай F – $m \times n$ матриця яскравості пікселів цифрового зображення.

Крок 1. Розбити матрицю F на непересічні блоки 8×8 .

Крок 2. Знайти для кожного блоку коефіцієнти ДКП.

Крок 3. Виділити з кожного блоку по 9 коефіцієнтів ДКП, що відповідають високим частотам.

Крок 4. Знайти середні арифметичні значення відповідних коефіцієнтів ДКП по всім блокам цифрового зображення.

Крок 5. Знаходимо П1 як різницю максимального та мінімального середніх значень високочастотних коефіцієнтів:

якщо $П1 > 0,1457$, *то* вважати цифрове зображення обробленим;

інакше вважати цифрове зображення необробленим.

В наступному розділі буде показано результати практичної реалізації розробленого метода виявлення фальсифікації зображень.

3. ПРОГРАМНА РЕАЛІЗАЦІЯ РОЗРОБЛЕНОГО МЕТОДУ

3.1 Вибір мови програмування

MATLAB – це платформа програмування, розроблена спеціально для інженерів і науковців для аналізу та проектування систем і продуктів, які змінюють наш світ. Основою MATLAB є мова MATLAB, матрична мова, яка дозволяє найбільш природне вираження обчислювальної математики.

Matlab розшифровується як Matrix Laboratory і є потужною технічною мовою для математичного програмування. Це математичний і графічний програмний пакет із багатьма вбудованими інструментами для розв'язування задач і розробки графічних ілюстрацій.

Його розробив Клів Молер наприкінці 1970-х років. Він починався як програмне забезпечення інтерфейсу для легкого доступу до бібліотек Fortran для чисельних обчислень без написання Fortran. Професійна версія Matlab із графічними функціями та функціями GUI була розроблена в 1983 році спільно Джоном Літлом, Клівом Молером та Стівом Бангертом.

Потім вони почали переписувати Matlab на C, що призвело до створення MathWorks у 1984 році. Відтоді Matlab став стандартом де-факто для аналізу даних, графічної візуалізації, числового аналізу та кількох інших програм.

MATLAB надає користувачеві велику кількість функцій для аналізу даних, які покривають майже всі області математики, зокрема:

- Матриці та лінійна алгебра – алгебра матриць, лінійні рівняння, власні значення і вектори, сингулярності, факторизація матриць та інше;
- Многочлени та інтерполяція – корені многочленів, операції над многочленами та їх диференціювання, інтерполяція та екстраполяція кривих;
- Математична статистика та аналіз даних – статистичні функції, статистична регресія, цифрова фільтрація, швидке перетворення Фур'є та інші;
- Обробка даних – набір спеціальних функцій, включаючи побудову графіків, оптимізацію, пошук нулів, чисельне інтегрування та інше;

- Диференційні рівняння – вирішення диференціальних і диференційно-алгебраїчних рівнянь, диференціальних рівнянь із запізнюванням, рівнянь з обмеженнями, рівнянь в часткових похідних та інше;
- Розріджені матриці – спеціальний клас даних пакету MATLAB, що використовується у спеціалізованих додатках;
- Цілочисельна арифметика – виконання операцій цілочисельної арифметики в середовищі MATLAB.

Мільйони інженерів і вчених у всьому світі використовують MATLAB для цілого ряду додатків у промисловості та наукових колах, включаючи глибоке й машинне навчання, обробку сигналів і зв'язок, обробку зображень і відео, системи керування, тестування й вимірювання, обчислювальні фінанси й обчислювальну біологію [32].

3.2 Інтерфейс та робота програми

Для створення програми було використано мову програмування Matlab. Для початку роботу необхідно запустити програму, на екрані з'явиться вікно, яке відображено на рисунку 3.1.

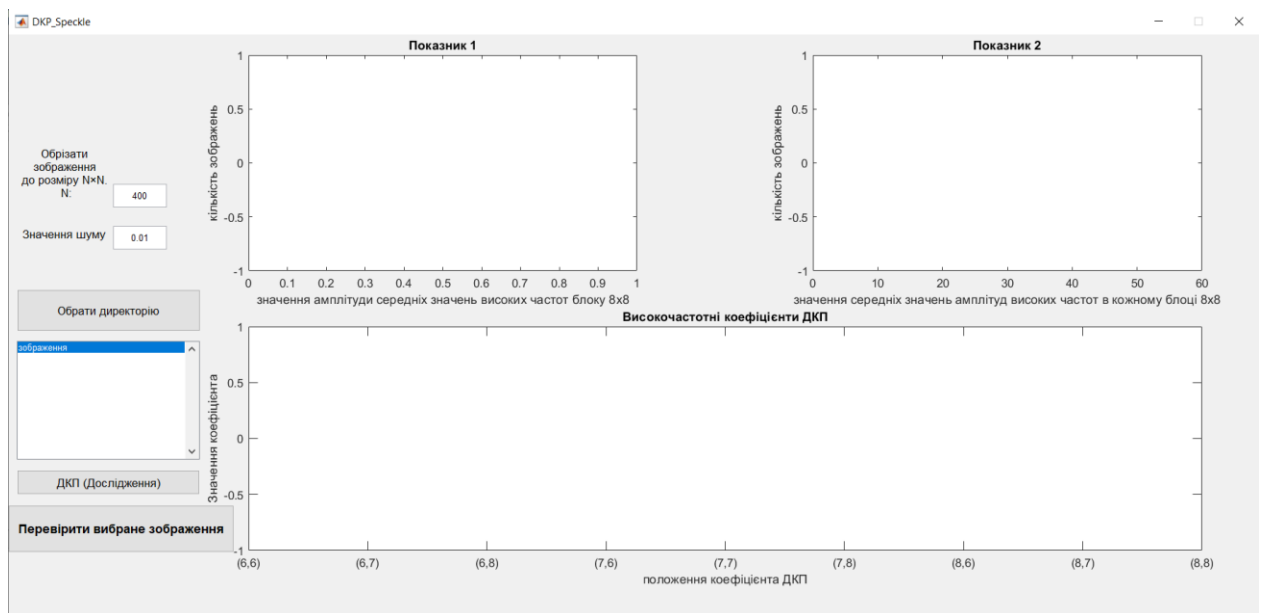


Рисунок 3.1 – Головне вікно програми

Біля надпису «Обрізати зображення до розміру N×N», можна писати розмір обрізання зображення, наприклад 400 на 400, як показано на рисунку 3.2.

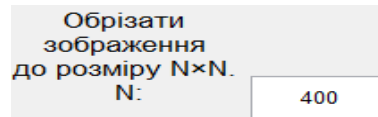


Рисунок 3.2 – Обрізання зображення

Далі, можна вписувати своє значення дисперсії мультиплікативного шуму, в даному випадку це 0.01, як зображено на рисунку 3.3.

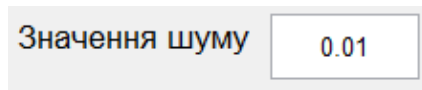


Рисунок 3.3 – Значення шуму

При натисканні на кнопку «Обрати директорію», з'являється вікно в якому необхідно обрати папку із зображеннями формату без втрат для подальшої роботи з ними, а також нижче даної кнопки виводиться список зображень, які знаходяться у вибраній директорії, як показано на рисунку 3.4. та рисунку 3.5.

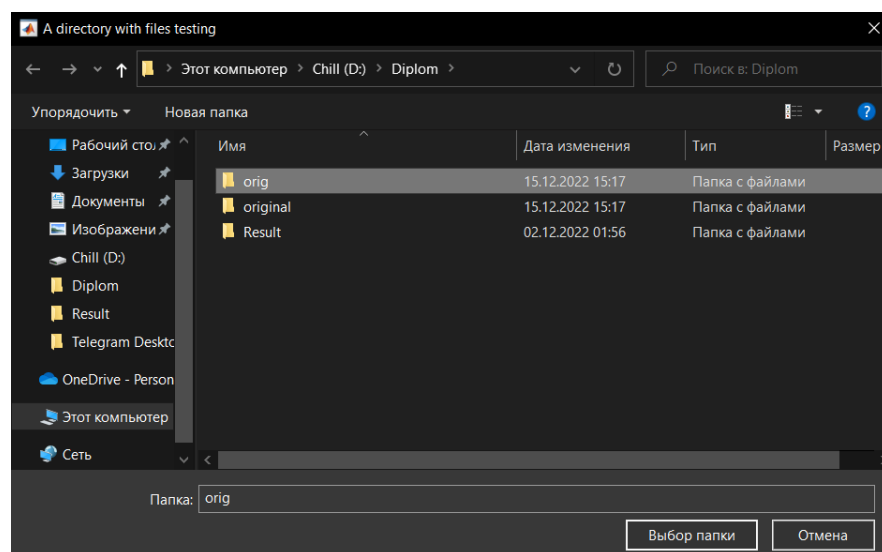


Рисунок 3.4 – Вікно для вибору директорії із зображеннями

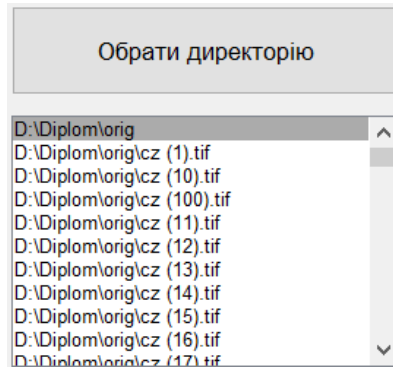


Рисунок 3.5 – Директорія та її вміст

Після обрання директорії із цифровими зображеннями, натискаємо на кнопку «ДКП (Дослідження)» і будуть розраховуватися наступне:

1. Високочастотні коефіцієнти ДКП для кожного зображення (оригінального і збуреного);
2. Амплітуди середніх значень високих частот блоку 8×8 ;
3. Середні значення амплітуд високих частот блоку 8×8 .

Також, будуть побудовані ці амплітуди та графік, як показано на рисунку 3.6, рисунку 3.7 і рисунку 3.8.

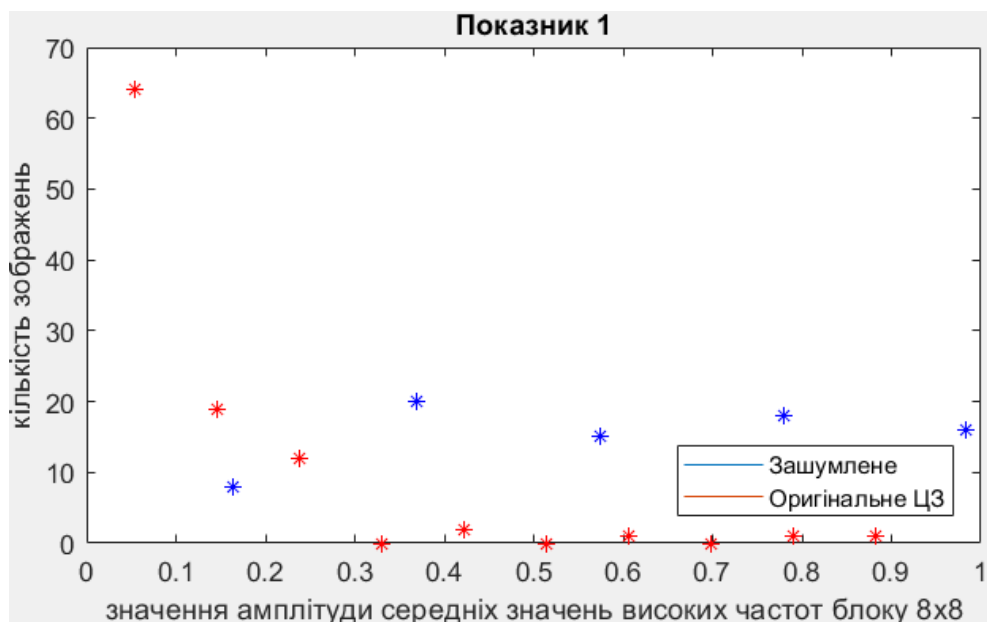


Рисунок 3.6 – Графік амплітуд середніх значень високих частот блоку 8×8

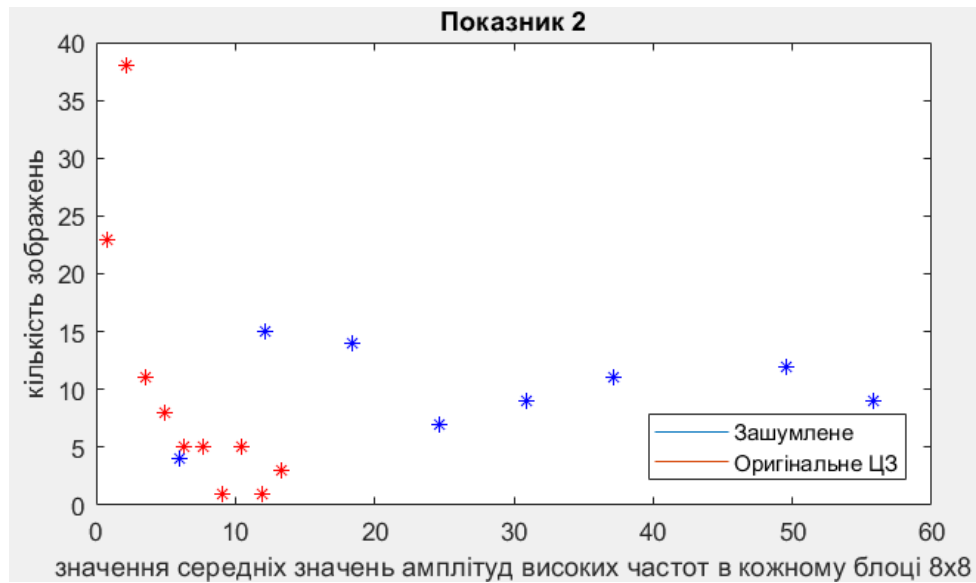


Рисунок 3.7 – Графік середніх значень амплітуд високих частот в кожному блоці 8×8

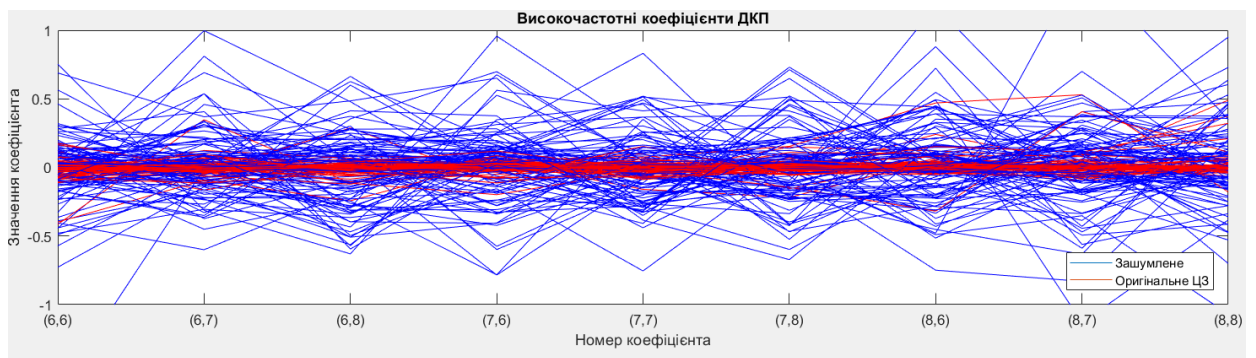


Рисунок 3.8 – Високочастотні коефіцієнти ДКП

Після завершення роботи даних розрахунків, можна зробити перевірку зображень на їх наявність фальсифікації. Для цього потрібно натиснути на директорію (див. рис.3.5) і натиснути на кнопку «Перевірити вибране зображення». Після завершення роботи даної функції нижче буде виведено надпис про завершення перевірки, як показано на рисунку 3.9.

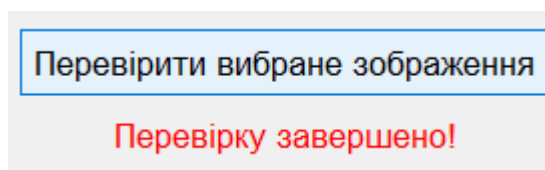


Рисунок 3.9 – Повідомлення про завершення перевірки

Результати про збурення зображень буде автоматично збережено у даний каталог в якому знаходиться користувач, у форматі Excel, як показано на рисунку 3.10.

У файл буде виведено такі результати:

- Перший стовбець це назва каталогу і зображення
- Другий стовбець це амплітуди середніх значень високих частот блоку 8×8
- Третій стовбець це виведений результат перевірки на фальсифікацію (оброблене чи необроблене зображення)

	A	B	C
1	NameImg	Amplitude	Result
2	D:\Diplom\orig\cz (1).tif	0.12209	Необроблене
3	D:\Diplom\orig\cz (10).tif	0.07729	Необроблене
4	D:\Diplom\orig\cz (100).tif	0.14368	Необроблене
5	D:\Diplom\orig\cz (11).tif		Оброблене
6	D:\Diplom\orig\cz (12).tif	0.13778	Необроблене
7	D:\Diplom\orig\cz (13).tif		Оброблене
8	D:\Diplom\orig\cz (14).tif	0.11903	Необроблене
9	D:\Diplom\orig\cz (15).tif	0.14166	Необроблене
10	D:\Diplom\orig\cz (16).tif		Оброблене
11	D:\Diplom\orig\cz (17).tif		Оброблене
12	D:\Diplom\orig\cz (18).tif	0.026944	Необроблене
13	D:\Diplom\orig\cz (19).tif		Оброблене
14	D:\Diplom\orig\cz (2).tif		Оброблене
15	D:\Diplom\orig\cz (20).tif		Оброблене
16	D:\Diplom\orig\cz (21).tif	0.047303	Необроблене
17	D:\Diplom\orig\cz (22).tif	0.022073	Необроблене
18	D:\Diplom\orig\cz (23).tif	0.051944	Необроблене
19	D:\Diplom\orig\cz (24).tif	0.020651	Необроблене
20	D:\Diplom\orig\cz (25).tif	0.038577	Необроблене
21	D:\Diplom\orig\cz (26).tif	0.070647	Необроблене
22	D:\Diplom\orig\cz (27).tif	0.024554	Необроблене
23	D:\Diplom\orig\cz (28).tif	0.039431	Необроблене
24	D:\Diplom\orig\cz (29).tif	0.040318	Необроблене
25	D:\Diplom\orig\cz (3).tif	0.030895	Необроблене
26	D:\Diplom\orig\cz (30).tif	0.10205	Необроблене
27	D:\Diplom\orig\cz (31).tif	0.049545	Необроблене
28	D:\Diplom\orig\cz (32).tif	0.043999	Необроблене
29	D:\Diplom\orig\cz (33).tif	0.024456	Необроблене
30	D:\Diplom\orig\cz (34).tif	0.06677	Необроблене

Рисунок 3.10 – Результат перевірки записаного у файл

Також, у користувача є можливість перевірки якого завгодно, обраного зображення, якщо він не хоче перевіряти їх усі.

Для цього обираємо зображення зі списку (див. рис.5), дали натискаємо на кнопку «Перевірити вибране зображення» і буде виведено текст, який з'явиться нижче графіка, з результат обробки.

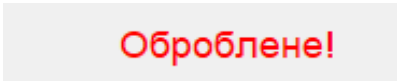
Якщо зображення необроблене буде виведений результуючий текст, як показано на рисунку 3.11.



Необроблене!

Рисунок 3.11 – Текст про перевірку вибраного зображення

Якщо цифрове зображення буде оброблене, то буде такий результат, який зображено на рисунку 3.12.



Оброблене!

Рисунок 3.12 – Текст обробленого зображення

Даний інтерфейс демонструє програмний продукт, розроблений для проведення експериментів та перевірки отриманих в даній роботі висновків, однак для користувача, який не є спеціалістом з інформаційної безпеки, даний функціонал є надлишковим, тому розроблено інтерфейс для потреб користувача (рис. 3.13).

Дана програма виконує основні функції, які потрібні для звичайного користувача, а саме:

- Вибір директорії (папки) із зображеннями для перевірки на підробку
- Перевірка всіх зображень у даній директорії
- Перевірка одного вибраного зображення

Результати перевірки будуть виглядати так само, як у програмі для експерта (див. рис. 3.9 – 3.12).

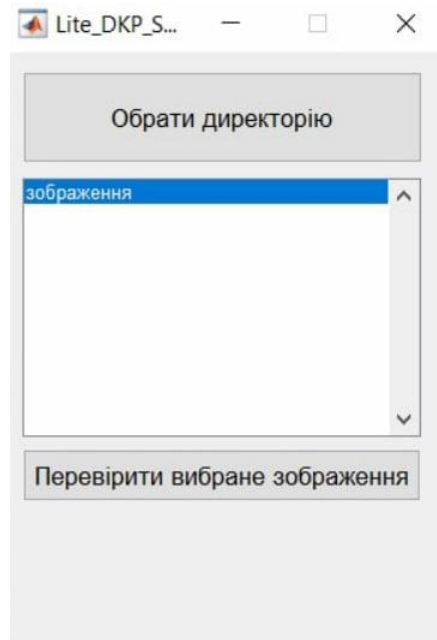


Рисунок 3.13 – Інтерфейс звичайного користувача

Таким чином в даному розділі показано результат програмної реалізації розробленого в кваліфікаційній роботі метода виявлення мультиплікативного шуму. Алгоритм реалізовано в двох варіантах: для користувача-експерта – з розширеним функціоналом і для користувача, який не є експертом з інформаційної безпеки.

ВИСНОВКИ

В даній кваліфікаційній роботі магістра проведено аналіз відомих методів виявлення фальсифікацій цифрових зображень. Встановлено, що проблема виявлення постобробки цифрових зображень після фальсифікації не є вирішеною до кінця, тому виявлення мультиплікативного шуму як порушення цілісності цифрового зображення є важливим і актуальним з точки зору інформаційної безпеки.

В якості параметрів для аналізу було розглянуто власні значення та коефіцієнти ДКП блоків матриці цифрового зображення. Встановлено, що використання власних значень для виявлення мультиплікативного шуму не є ефективним, в той час як використання коефіцієнтів ДКП дало задовільні результати.

Встановлено, що аналіз високочастотних коефіцієнтів цифрового зображення дозволяє відділити оброблені зображення від необроблених.

На основі отриманих результатів розроблено метод виявлення мультиплікативного шуму як порушення цілісності цифрового зображення. Ефективність даного методу в термінах помилок 1 і 2 роду складає 7% і 17% відповідно. Метод має потенціал для подальшого вдосконалення.

Розроблений метод реалізовано у вигляді програмного продукту з простим користувацьким інтерфейсом.

Всі задачі роботи вирішено, мету досягнуто.

ПЕРЕЛІК ПОСЛАНЬ

1. В. В. Зоріло, Є. О. Корольова, Б. О. Розум'як. Метод виявлення розмиття та мультиплікативного шуму як обробки цифрового зображення на основі аналізу коефіцієнтів ДКП. *Інформатика та математичні методи в моделюванні*. №3-4. 2022. С.23-36.
2. Mahdian B., Saic S. Blind methods for detecting image fakery. *IEEE Aerospace Electron System Management*, 2010. P.18-24. URL: <https://ieeexplore.ieee.org/document/5467652/authors#authors>
3. Shwetha B., Sathyanarayana S.V. Digital image forgery detection techniques: a survey. *ACCENTS Transactions on Information Security*, 2017. Vol.2(5). URL: https://www.accentjournals.org/paperInfo_online.php?journalPaperId=926&countPaper=251
4. Zang H.B., Yang C., Quan X.M. Image authentication based on digital signature and semi-fragile watermarking. *Journal of Computer and Technology*. 2004. URL: <https://link.springer.com/article/10.1007/BF02973435>
5. Lu C.S., Liao H.Y.M. Structural digital signature for image authentication: An incidental Distortion Resistant Scheme. *IEEE Transactions on multimedia*. 2003. URL: <https://www.iis.sinica.edu.tw/papers/lcs/1745-F.pdf>
6. Fridrich J, Robust bit extraction from images. *IEEE international conference on in multimedia computing and systems*. 1999. P. 536-540. URL: <https://ieeexplore.ieee.org/document/778542>
7. Farid H. A survey of image forgery detection. *IEEE Signal Processing Magazine*. 2006. P.6- 25. URL: <http://ceng2.ktu.edu.tr/~gulutas/dif/1.pdf>
8. Gang Z.Y., Gang N.R.C., Detection of image sharpening based on histogram aberration and ringing artifacts. *IEEE ICME*. 2009. P.1026-1029. URL: <https://ieeexplore.ieee.org/abstract/document/5202672>

9. Peng F., Nie Y., Long M. () “A complete passive blind image copy-move forensics scheme based on compound statistics features. *International journal of Forensic science*. 2011. P.21-25. URL: <https://www.ncbi.nlm.nih.gov/pubmed/21726968>
10. Zhao Y.Q., Liao M., Shih F.Y., Shi Y.Q. Tampered region detection of inpainting JPEG images. *International Journal on light electron optics*. 2013. P.2487-2492. URL: https://www.researchgate.net/publication/256821420_Tampered_region_detection_of_inpainting_JPEG_images
11. Christlein V., Riess C., Jordan J., Angelopoulou E. An evaluation of popular copymove forgery detection Approaches. *IEEE Transactions on information forensics and security*. 2012. P.1841-1854. URL: <https://arxiv.org/pdf/1208.3665.pdf>
12. Soloria B., Nandi A.K. Automated detection and localization of duplicated regions affected by reflection, rotation and scaling in image forensics. *International Journal of signal Processing*. 2011, P.1759-1770. URL: <https://dl.acm.org/citation.cfm?id=1975324>
13. Ibrahim R.W., Moghaddasi Z., Jalab H.A., Noor R.M., () Fractional differential texture descriptors base on the mach ado entropy for image splicing detection. *International Journal of Computer Science*, 2015. P.4775-4786. URL: <https://pdfs.semanticscholar.org/49d5/b0b62fca0b20680697aa9238467f3f33f516.pdf>
14. Makandar A., Halalli B. A review on preprocessing techniques for digital mammography images. *International Journal of computer applications*. 2015. P.0975-887. URL: <https://pdfs.semanticscholar.org/82f6/787a9b2e703feb3abb3b3e728dc8702fb637.pdf>
15. Phkan A., Borah M. A survey paper on the feature extraction module of offline handwriting character recognition. *International Journal of Computer*. 2019. Vol.33, No 1. P. 84-91 URL: <https://pdfs.semanticscholar.org/c9d6/e01e189d81f3ad2f08cabbcbd12cdf92db90.pdf>
16. Munirah M.Y., Nawi N.M., Wahid N., Shukra M. A comparative analysis of feature selection techniques for classification problems. *ARNP Journal of Engineering and*

Applied sciences. 2016. P.13176-13187. URL: http://www.arpnjournals.org/jeas/research_papers/rp_2016/jeas_1116_5365.pdf

17. Sutthiwan P., Q.Shi Y., Wei S., Tian N. Rake transform and edge statistics for image forgery detection. *IEEE international conference on multimedia*. 2010. P.1463-1468. URL: <https://ieeexplore.ieee.org/document/5583264>

18. Lu W., Sun W., W.Huang J. Digital image forensics using statistical features and neural network classifiers. *International conference on machine learning and cybernetics*. 2008. P.12-15. URL: <https://ieeexplore.ieee.org/document/4620890>

19. Fang Z., Wang S., Zhang X. Image splicing detection using camera inconsistency. *International Conference on multimedia information networking and security*. 2009. P.20-4. URL: <https://ieeexplore.ieee.org/abstract/document/5368456>

20. Luo W., Huang J., Qiu G. Robust detection of region-duplication forgery in digital images. *International conference on Pattern recognition*. 2006. P.746-749. URL: <https://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=11159>

21. Ghorbani M., Firouzmand M., Faraahi A. DWT-DCT(QCD) based copy-move image forgery detection. *International conference on systems, signals and image processing*, 2011. P.1-4. URL: <https://ieeexplore.ieee.org/abstract/document/5977368>

22. Christlein V., Riess C., Jordan J., E.Angelopoulou. An evaluation of popular copymove forgery detection Approaches. *IEEE Transactions on information forensics and security*. 2012. P.1841-1854. URL: <https://arxiv.org/pdf/1208.3665.pdf>

23. Bharti C.N., Tandel P. A Survey of Image Forgery Detection Techniques. *IEEE WiSPNET*. 2016. URL: <https://ieeexplore.ieee.org/document/7566257>

24. Discrete Cosine Transform URL: <https://www.mathworks.com/help/images/discrete-cosine-transform.html>

25. Introduction to SIFT (Scale-Invariant Feature Transform) URL: https://docs.opencv.org/4.x/da/df5/tutorial_py_sift_intro.html

26. Eigen Decomposition Theorem URL: <https://mathworld.wolfram.com/EigenDecompositionTheorem.html>

27. Eigen Decomposition URL: <https://mathworld.wolfram.com/EigenDecomposition.html>
28. Eigenvalues and eigenvectors URL: https://www.mathworks.com/help/matlab/ref/eig.html?searchHighlight=eig&s_tid=srchtitle_eig_1#btwn8p3-1
29. Discrete Cosine Transform URL: <https://www.mathworks.com/help/images/discrete-cosine-transform.html>
30. Dct, Discrete cosine transform URL: https://www.mathworks.com/help/signal/ref/dct.html?s_tid=doc_ta#bvldci7
31. Кобозева А.А. Алгоритмічна реалізації універсального методу виявлення порушень цілісності цифрового зображення та дослідження її ефективності. Одеса: НУОП, 2021. 9 с.
32. What is Matlab? URL: <https://www.mathworks.com/discovery/what-is-matlab.html>