Міністерство освіти і науки України

Харківський національний університет радіоелектроніки

Факультет комп’ютерної інженерії та управління
(повна назва)

Кафедра електронних обчислювальних машин
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА

Пояснювальна записка

Рівень вищої освіти другий (магістерський)

Пакетна стеганографія та об'єднаний стеганоаналіз в JPEG-зображеннях

Виконав:

студент II курсу, групи КСМм-23-1

Нарватов О. П.
(прізвище, ініціали)

Спеціальність

123 «Комп’ютерна інженерія»
(код і повна назва спеціальності)

Тип програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Освітня програма

Комп’ютерні системи та мережі
(повна назва освітньої програми)

Керівник: доц. Бологова Н.М.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри ЕОМ _____ Коваленко А.А.
(підпис) (прізвище, ініціали)

2025 р.

Харківський національний університет радіоелектроніки

Факультет комп’ютерної інженерії та управління

Кафедра електронних обчислювальних машин

Рівень вищої освіти другий (магістерський)

Спеціальність 123 «Комп’ютерна інженерія»
(код і повна назва)

Тип програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Освітня програма Комп’ютерні системи та мережі
(повна назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)
“____” _____ 20___ р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

студенту Нарватову Олександру Петровичу
(прізвище, ім’я, по батькові)

1. Тема роботи Пакетна стеганографія та об'єднаний стеганоаналіз в JPEG-зображеннях

затверджена наказом по університету від “ 22 ” листопада 2024 р. № 1237 Ст

2. Термін подання студентом роботи до екзаменаційної комісії 20 січня 2025 р.

3. Вхідні дані до роботи набір стего зображень

4. Перелік питань, що потрібно опрацювати у роботі

1) аналіз існуючих підходів;
2) розробка стратегій розподілу;
3) об'єднаний стегоаналіз;
4) моделювання стеганографічних схем;
5) емпіричні дослідження.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) 14 слайдів

6. Консультанти розділів роботи (заповнюється за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Огляд сучасних стеганографічних методів	26.11.24-30.11.24	
	та стегоаналізу в JPEG-зображеннях		
2	Аналіз існуючих підходів до виявлення	02.12.24-05.12.24	
	прихованої інформації		
3	Вибір стратегії розподілу даних для тестування	06.12.24-10.12.24	
4	Розробка дискримінативної функції об'єднання	11.12.24-21.12.24	
	для стегоаналізу		
5	Проведення експериментів	23.12.24-03.01.25	
6	Оформлення матеріалів кваліфікаційної роботи	04.01.25-07.01.25	
7	Подання кваліфікаційної роботи керівникові	08.01.25-11.01.25	
8	Подання кваліфікаційної роботи на рецензу-	13.01.25-17.01.25	
	вання		

Дата видачі завдання 25 листопада 2024 р.

Студент _____
(підпис)

Керівник роботи _____ доц. Бологова Н.М.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 54 с., 17 рис., 1 дод., 15 джерел.

ПАКЕТНА СТЕГАНОГРАФІЯ, ОБ'ЄДНАНИЙ СТЕГОАНАЛІЗ, JPEG, DCT-КОЕФІЦІЄНТИ, АДАПТИВНЕ ВБУДОВУВАННЯ, СТРАТЕГІЇ РОЗПОДІЛУ ДАНИХ, ІНФОРМАЦІЙНА БЕЗПЕКА, ПРИХОВУВАННЯ ДАНИХ, ДИСКРИМІНАТИВНА ФУНКЦІЯ ОБ'ЄДНАННЯ.

Метою кваліфікаційної роботи є розробка та аналіз методів пакетної стеганографії та об'єднаного стегоаналізу в JPEG-зображеннях, спрямованих на підвищення ефективності приховування інформації та мінімізацію ризику її виявлення.

У ході виконання кваліфікаційної роботи було досліджено сучасні стеганографічні підходи, зокрема пакетні стратегії розподілу даних (Greedy, Linear, Uses-β, IMS, DeLS, DiLS). Проведено аналіз особливостей JPEG-формату, зокрема використання DCT-коефіцієнтів, для підвищення ефективності адаптивних схем вбудовування. Також розглянуто методи об'єднаного стегоаналізу, включаючи розробку дискримінативної функції об'єднання, яка дозволяє виявляти приховані дані навіть за відсутності інформації про стратегію розподілу. Емпіричні дослідження підтвердили ефективність адаптивних стратегій, таких як DeLS і IMS, у мінімізації статистичної виявлюваності.

Результати роботи можуть бути використані у сфері інформаційної безпеки, для захисту конфіденційної інформації та створення безпечних каналів зв’язку.

ABSTRACT

Master’s thesis: 54 pages, 17 figures, 1 appendices, 15 sources.

BATCH STEGANOGRAPHY, POOLED STEGANALYSIS, JPEG, DCT COEFFICIENTS, ADAPTIVE EMBEDDING, DATA DISTRIBUTION STRATEGIES, INFORMATION SECURITY, DATA HIDING, DISCRIMINATIVE POOLING FUNCTION.

The major goal of this thesis is to develop and analyze methods of batch steganography and pooled steganalysis in JPEG images aimed at improving the efficiency of information hiding and minimizing the risk of its detection.

During the qualification work, modern steganographic approaches were studied, including batch data distribution strategies (Greedy, Linear, Uses-β, IMS, DeLS, DiLS). An analysis of the features of the JPEG format, particularly the use of DCT coefficients, was conducted to enhance the efficiency of adaptive embedding schemes. Additionally, pooled steganalysis methods were examined, including the development of a discriminative pooling function capable of detecting hidden data even in the absence of knowledge about the distribution strategy. Empirical studies confirmed the effectiveness of adaptive strategies, such as DeLS and IMS, in minimizing statistical detectability.

The results of this work can be applied in the field of information security to protect confidential information and establish secure communication channels.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

АРМ – автоматизоване робоче місце

ІАЦ – інформаційно-аналітичний центр

WDS – бездротова розподільча система (англ., Wireless Distribution System)

ВСТУП

З давніх часів люди прагнули спілкуватися один з одним таємно, тому що предмет спілкування є делікатним і повинен зберігатися в таємниці від третіх осіб (супротивників).

Двадцять перше століття пережило значний технічний прогрес з точки зору використання цифрових пристроїв. Зокрема, ці пристрої використовуються для зберігання та спільного використання мультимедійних файлів, таких як текст, аудіо, відео та файли зображень.

З появою Інтернету та соціальних медіа, які є широко використовуваними каналами зв’язку, це призвело до того, що обмін цифровими файлами став щоденним стандартом для багатьох людей у всіх секторах. Потреба в конфіденційних і безпечних комунікаціях стала критично важливим аспектом. Дослідження, що стосуються безпеки, конфіденційності та цілісності інформації та комунікацій, є важливою частиною досліджень під заголовками приховування інформації та безпеки комунікацій.

Однією з критичних вимог у спілкуванні є конфіденційність інформації. Цю конфіденційність можна досягти за допомогою криптографії, яка кодує повідомлення таким чином, що його не можуть зрозуміти сторонні особи без знання пароля. Але в криптографічній програмі можемо відразу побачити, що повідомлення присутнє, навіть якщо не можемо його прочитати.

Інша ідея полягає в тому, щоб приховати повідомлення непомітним способом. Саме в цьому полягає роль стеганографії, яка пропонує методи приховування повідомлення у вмісті файлу, щоб спілкування виглядало нормальним і не привертало уваги третіх осіб, навіть якщо повідомлення не зашифроване. Лише уповноважені особи мають як деякі підказки щодо наявності повідомлення, так і його характеристики (як, наприклад, його

розмір), вони можуть використовувати правильний метод і, зрештою, ключ (так званий стего-ключ), щоб знайти та витягти повідомлення.

Сьогодні людство не може відмовитися від цифрових даних, тому захист інформації стає критично важливим. Один із методів захисту цифрового контенту є стеганографія, яка дозволяє сховати дані всередині цифрового контенту, наприклад, JPEG-файлів. Але з розвитком методів стеганографії розвиваються і технології їх виявлення, які відносяться до стегоаналізу.

Таким чином, вдосконалення методів приховування та виявлення інформації в цифрових файлах має велике значення для інформаційної безпеки в різних сферах, від персонального користування до захисту особливо секретних даних.

1 ОПИС JPEG СТАНДАРТУ

1.1 Стиснення JPEG

Процес стиснення JPEG для кольорового зображення RGB складається з п’яти кроків:

- перетворення Y CbCr;
- підвибірка та поділ блоків;
- дискретне косинусне перетворення (DCT);
- квантування;
- стиснення без втрат.

Колір перетворюється з RGB (червоний, зелений, синій) на Y C_b C_r за допомогою наступного рівняння:

$$\begin{cases} Y = 16 + \frac{65.738R}{256} + \frac{129.057G}{256} + \frac{25.064B}{256}, \\ C_b = 128 - \frac{65.738R}{256} + \frac{129.057G}{256} + \frac{25.064B}{256}, \\ C_r = 128 + \frac{65.738R}{256} + \frac{129.057G}{256} + \frac{25.064B}{256} \end{cases} \quad (1.1)$$

де Y представляє канал яскравості, C_b – канал кольоровості синього кольору (насиченість відтінку зображення), а C_r – канал кольоровості червоного кольору (насиченість кольору зображення). Важливість цього перетворення полягає в тому, що воно обмежує інформацію про яскравість одним каналом Y, який більше відповідає сприйняттю кольору зоровою системою людини [1].

Після перетворення Y C_bC_r канали кольоровості та яскравості розділяються. Як згадувалося раніше, людське око краще сприймає яскравість.

Таким чином, можна видалити деяку інформацію про колір із каналів C_b і C_r, не надто втрачаючи якість; це мета субдискретизації кольоровості (також називається зменшення дискретизації).

Більш детально, припустімо, що у нас є блок пікселів 4 × 2 для каналів кольоровості (Cr, Cb), і нам потрібно зробити підвибірку каналів кольоровості вихідних пікселів, для цього є три можливі співвідношення, як показано на рисунку 1.1:

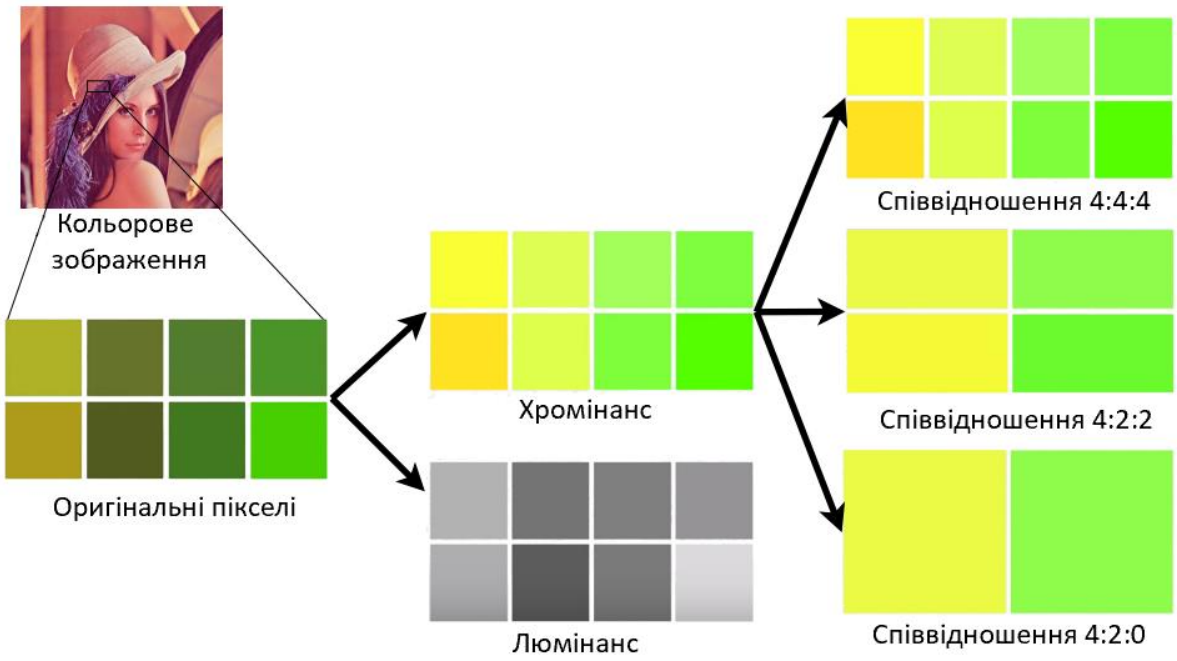

Рисунок 1.1 – Типи піддискретизації кольоровості

Співвідношення 4:4:4 означає, що субдискретизація не виконується, це еквівалентно оригінальному зображенню, тобто кожен піксель зберігає інформацію про колір. Перше значення (4) означає ширину блоку 4×2, друге значення (4) означає, що 4 пікселі з першого рядка зберігають свої значення, третє значення (4) означає, що 4 пікселі з другого рядка зберігають свої значення. їх значення, як показано на рисунку 1.2.

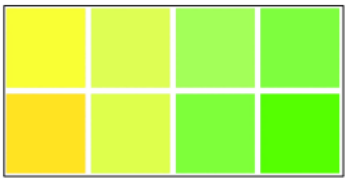

Рисунок 1.2 – Співвідношення 4:4:4

Співвідношення 4:2:2 : перше значення (4) означає ширину блоку 4×2, друге значення (2) означає, що 2 пікселі з першого рядка зберігають свої значення, третє значення (2) означає, що 2 пікселі з другого рядка зберігають свої значення, як показано на рисунку 1.3.

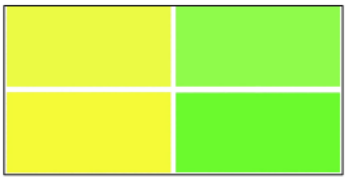

Рисунок 1.3 – Співвідношення 4:2:2

Співвідношення 4:2:0 : перше значення (4) означає ширину блоку 4 × 2, друге значення (2) означає, що 2 пікселі з першого рядка зберігають свої значення, третє значення (0) означає, що не пікселів у другому рядку зберігають свої значення, що означає, що другий рядок має ті самі кольори, що й перший, як показано на рисунку 1.4.

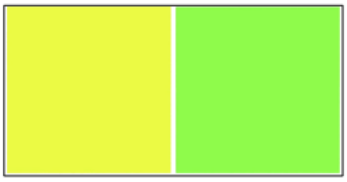

Рисунок 1.4 – Співвідношення 4:2:0

Нарешті, кожен канал ділиться на блоки 8×8 пікселів. Очевидно, що для зображень у відтінках сірого немає потреби у субдискретизації кольоровості, лише канал Y зберігається для наступного кроку, також дискретизується блоками 8 × 8.

Метою DCT-перетворення є перетворення піксельних значень Y C_bC_r каналів на частотні значення. Для цього використовується дискретне косинусне перетворення (DCT). DCT є синусоїдальною функцією, вона приймає як вхід значення пікселя P[i, j], $0 \le i \le 7$, $0 \le j \le 7$ блоку 8×8 пікселів P, і перетворює його на значення частоти F[k, l], $0 \le k \le 7$, $0 \le l \le 7$ із 8 × 8 блок частот Ф.

Перед застосуванням функції DCT значення пікселів у P зсуваються з діапазону [0, 255] до [−128, 127] шляхом простого віднімання значень на 128. У фазі стиснення функція DCT називається прямим DCT (FDCT). [2] і визначається як:

$$F[k,\ l]\ =\ \frac{1}{4}\ \gamma(k)\gamma(l)\ \sum_{i=0}^{7}\sum_{j=0}^{7}P[i,\ j]\ cos(\frac{(2i\ +\ 1)k\pi}{16})cos(\frac{(2j\ +\ 1)l\pi}{16})\ ,\quad (1.2)$$

де

$$\gamma(k),\ \gamma(l)\ =\ \begin{cases}\frac{1}{\sqrt{2}},\ якщо\ f\ k,\ l\ =\ 0\\ 1,\ всі\ інші\end{cases}.\qquad (1.3)$$

Інформація кожного пікселя P[i, j] ∈ P потім поширюється на весь блок DCT 8 × 8, оскільки кожне значення F[k, l] залежить від усіх значень у блоці пікселів, як ми бачимо в рівнянні (1.2). Перший коефіцієнт блоку DCT F[0, 0] називається коефіцієнтом постійного струму, який є коефіцієнтом із нульовою частотою в обох вимірах. Решта 63 коефіцієнта називаються коефіцієнтами AC, це коефіцієнти з ненульовими частотами.

Метою етапу квантування є зменшення кількості бітів, які використовуються для зберігання зображення, шляхом зменшення кількості інформації у високочастотних компонентах (AC). Цей метод пов’язаний з тим, що неозброєним оком неможливо розрізнити точну інтенсивність зміни яскравості на високій частоті [3].

Технічно квантування здійснюється шляхом окремого ділення його коефіцієнтів DCT на значення з матриці квантування Q, а потім округлення їх до найближчого цілого числа, як показано в рівнянні (1.4)

$$T[k,\ l]\ =\ round\ \left(\frac{F[k,\ l]}{Q[k,\ l]}\right). \tag{1.4}$$

Отримана матриця T, яка називається квантованою матрицею DCT, використовується на наступному кроці. Значення в Q залежать від коефіцієнта якості стиснутого зображення. показує таблицю квантування для коефіцієнта якості 75.

$$\begin{bmatrix} 8 & 6 & 5 & 8 & 12 & 20 & 26 & 31 \\ 6 & 6 & 7 & 10 & 13 & 29 & 30 & 28 \\ 7 & 7 & 8 & 12 & 20 & 29 & 35 & 28 \\ 7 & 9 & 11 & 15 & 26 & 44 & 40 & 31 \\ 9 & 11 & 19 & 28 & 34 & 55 & 52 & 39 \\ 12 & 18 & 28 & 32 & 41 & 52 & 57 & 46 \\ 25 & 32 & 39 & 44 & 52 & 61 & 60 & 51 \\ 36 & 46 & 48 & 49 & 56 & 50 & 52 & 50 \end{bmatrix}. \tag{1.5}$$

Останнім етапом створення стисненого файлу JPEG є стиснення без втрат, рис. 1.5. Цей крок починається з зигзагоподібного сканування, способу перегрупування коефіцієнтів DCT, щоб кластеризувати разом коефіцієнти з подібними частотами. Далі алгоритм кодування довжини циклу (RLE) стискає коефіцієнти змінного струму, а алгоритм диференційної імпульсно-

кодової модуляції (DPCM) стискає коефіцієнт постійного струму. Нарешті, для створення остаточного файлу JPEG застосовується алгоритм Хаффмана або арифметичний алгоритм кодування.

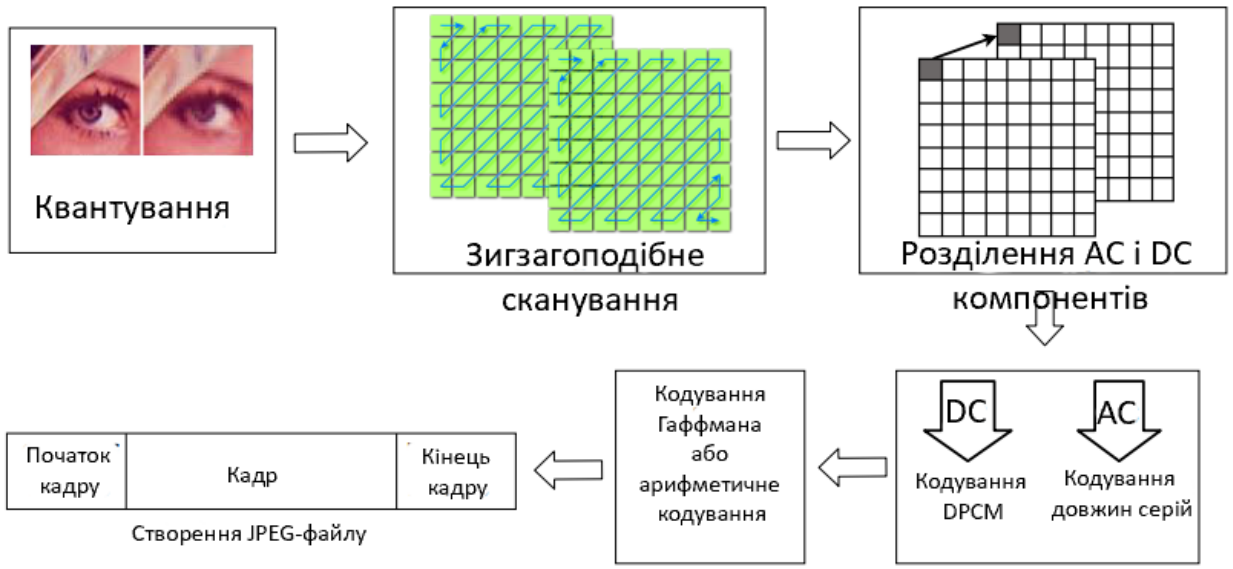

Рисунок 1.5 – Стиснення без втрат відразу після кроку квантування

1.2 Декомпресія JPEG

Процес декомпресії використовується для отримання вихідного зображення зі стисненого. Застосовуються ті самі кроки, що описані вище, але навпаки, як показано на рисунку 1.1, зокрема, DCT стає зворотним DCT (IDCT)

$$F[k,\ l]\ =\ \frac{1}{4}\ \ \sum_{i=0}^{7}\sum_{j=0}^{7}\gamma(k)\gamma(l)F[k,\ l]\ cos(\frac{(2i\ +\ 1)k\pi}{16})cos(\frac{(2j\ +\ 1)l\pi}{16})\ ,\ \ (1.6)$$

2 СТЕГАНОГРАФІЯ У JPEG

2.1 Огляд стеганографії в JPEG

Стеганографічний канал – це середовище, необхідне для здійснення стеганографічного зв'язку. У цьому середовищі повідомлення приховані в звичайних на вигляд зображеннях, які потім передаються прозоро.

Алісі, стеганографу, потрібне зображення обкладинки, щоб вставити своє повідомлення за допомогою алгоритму вбудовування, і стегоключ. Крім того, перед надсиланням стего-зображень Аліса ділиться з одержувачем, Бобом, деякою інформацією про стеганографічну операцію: алгоритм вилучення та стего-ключ перед передачею стего-зображення. Спільна інформація дозволить Бобу витягнути повідомлення із зображення обкладинки. Усі елементи представлені на рисунку 2.1.

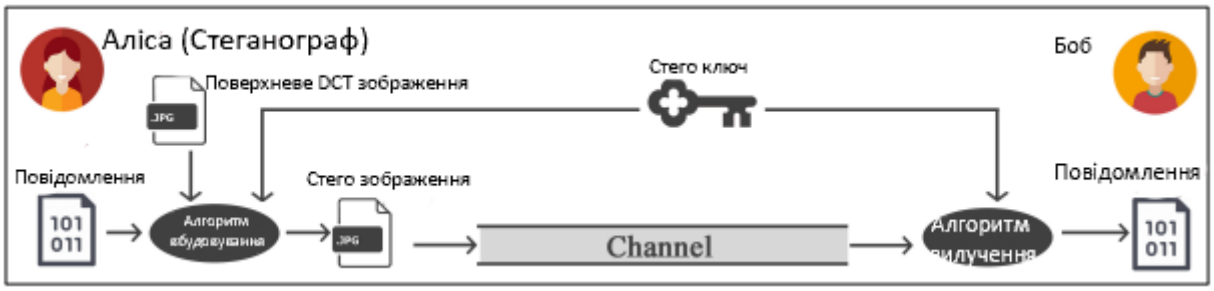

Рисунок 2.1 – Стеганоканал і основні елементи

Операції в стеганографічному каналі теоретично формулюються як функції, які приймають зображення як вхідні дані, маніпулюють ними (вбудовують, витягують) і видають результати як вихідні дані. Нехай дано набір зображень обкладинки X , набір ключів K, пов’язаних із зображеннями обкладинки, і набір повідомлень M, які Аліса може вбудовувати чи вилучати і це можна виразити наступним чином:

$$\begin{matrix} Emb : X \times K \times M \rightarrow X \\ Ext : X \times K \rightarrow M \end{matrix} . \quad (2.1)$$

Заміна LSB є однією з перших технік вбудовування. Підхід LSB був представлений у 1993 році [4], де було представлено дві методики для приховування даних у зображеннях просторової області. Ці методи базувалися на модифікації найменшого значущого біта (LSB) пікселя за допомогою алгоритму, запропонованого [5]. Цей алгоритм був відомий як метод зі зниженням якості зображення [6].

Далі розглянемо, як працює техніка заміни LSB у домені JPEG. Операція вбудовування (рівняння (2.1)), яка описує заміну LSB, працює наступним чином: спочатку повідомлення, яке вбудовується, перетворюється на бітовий потік. Потім біти повідомлення перезаписують молодші біти в квантованих значеннях DCT зображення. Повідомлення можна вбудовувати послідовно або псевдовипадково за допомогою стегоключа.

Операція вилучення (рівняння (2.1)) працює шляхом простого вилучення LSB зі стего-зображення та його перебудови за допомогою стего-ключа, який описує порядок бітів повідомлення.

На рисунку 2.2 можемо побачити, як LSB, виділені жирним зеленим шрифтом значення, модифіковані для обробки всіх 8-бітних $(0, 1, 0, 0, 1, 1, 1, 1)^T$ двійкових значень (виділено жирним червоним шрифтом) .

$$\begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 1 & \mathbf{0} \\ 1 & 1 & 1 & 1 & 0 & 0 & 1 & \mathbf{1} \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & \mathbf{0} \\ 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & \mathbf{1} \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}$$

Рисунок 2.2 – Приклад заміни LSB

LSB-matching майже ідентичний техніці LSB-replacement; різниця полягає в тому, що алгоритм робить після перетворення квантизованих значень DCT зображення та даних повідомлення у бітовий потік. Метод вбудовування за допомогою LSB-matching було вперше запропоновано [7]. Варто зазначити, що всі сучасні та безпечні стеганографічні схеми базуються на вбудовуванні за допомогою LSB-matching.

Операція вбудовування (Рівняння (2.1)), що описує роботу LSB-matching, виконується наступним чином: по-перше, повідомлення, яке потрібно приховати, перетворюється у бітовий потік. Далі, для кожного значення зображення та відповідного біта повідомлення, якщо LSB (молодший значущий біт) і відповідний біт повідомлення відрізняються, виконується випадкова двійкова операція (додавання або віднімання); в іншому випадку нічого не виконується. При виборі випадкової двійкової операції дотримується умова уникнення виходу за межі допустимого діапазону значень у квантизованих DCT зображення, який має належати діапазону {-1023, ..., 1024}.

Операція вилучення (Рівняння (2.1)) має таку ж складність, як і LSB-replacement. Вона працює шляхом вилучення LSB (молодших значущих бітів) зі стего-зображення та відновлення повідомлення за допомогою стего-ключа, який визначає порядок бітів повідомлення.

На рисунку 2.3 ми застосовуємо LSB-matching вбудовування на прикладі з рисунку 2.2. Жирні зелені значення змінені через випадкові двійкові операції з 8-бітовими двійковими значеннями $(0, 1, 0, 0, 1, 1, 1, 1)^T$ (червоними). На цьому рисунку видно, що алгоритм виконує двійкові операції лише тоді, коли значення повідомлення і LSB відрізняються.

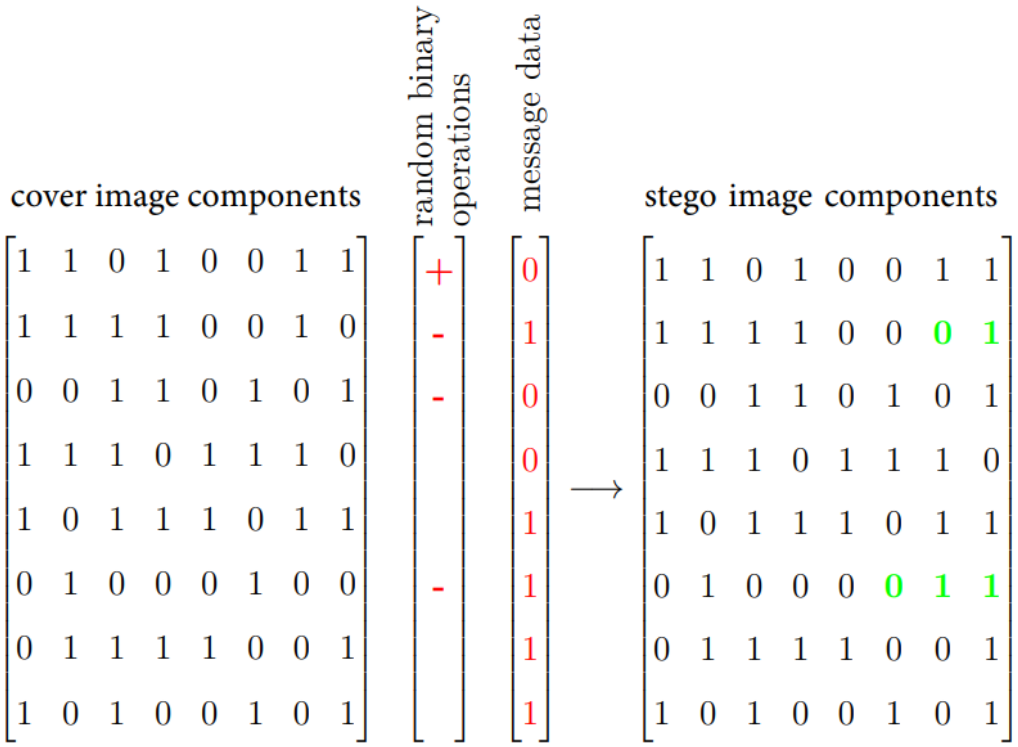

Рисунок 2.3 – Приклад, який ілюструє техніку LSB-replacement

Ідея використання кодів Хеммінга в стеганографії була вперше запропонована [8] як метод матричного кодування для підвищення ефективності вбудовування для стеганографічної схеми. Мета полягає в тому, щоб мати якомога менше модифікацій, одночасно вставляючи якомога більше бітів. Фактично, ця ідея була досліджена того ж року в раніше не опублікованій статті [8].

Код Хеммінга – це лінійний код з виправленням помилок, який може виявляти та виправляти однобітові помилки. (n, n − k) код Хеммінга використовує n бітів для передачі n − k бітів повідомлення, решта k бітів, які використовуються для виправлення помилок, називаються бітами перевірки парності, де $n = 2^k - 1$ у двійковому полі, яке містить двійкові числа.

Нехай дано біт повідомлення m розміром n − k. Щоб передати m до приймача через шумовий канал зв’язку, m перетворюється на кодове слово у розміром n бітів за допомогою (n − k) × n матриці генератора коду G наступним чином:

$$y = \left(m^T \times G^T\right) . \quad (2.2)$$

Нехай x розміром n буде кодовим словом, отриманим відправником y. Матриця перевірки парності H розміру k×n використовується для обчислення вектора синдрому z розміром k для перевірки помилки наступним чином:

$$z = H \times x . \quad (2.3)$$

Якщо вектор синдрому $z = 0_k$, приймач може зробити висновок, що помилки не сталося. В іншому випадку, для позиції вектора $i \in \{1, \ldots, n - k\}$, якщо генераторна матриця G і перевірочна матриця правильно організовані, а z, інтерпретований як число, вказує на позицію, тоді помилка виявляється в i-й позиції x . Таким чином, x коригується шляхом зміни значення на i-й позиції.

Технічно зміна може бути виконана за допомогою операції $x = x \oplus e_i$, де $\oplus$ – це операція XOR, а e_i – це вектор помилки, тобто одиничний вектор довжини n .

Проблема вбудовування в матрицю формалізується наступним чином: принцип полягає в тому, щоб модифікувати початковий носій x у y, враховуючи m , таким чином, щоб:

$$Hy = m, \quad (2.4)$$

з H матрицею перевірки парності коду. Перетворення вектора x в y виконується шляхом пошуку вектора модифікації e:

$$y = x + e. \quad (2.5)$$

З рівняння (2.5) і рівняння (2.4) ми отримуємо таку формулу:

$$H(x + e) = m \Longleftrightarrow He = Hx - m. \quad (2.6)$$

Проблема полягає в тому, щоб знайти вектор ее, який є кодовим словом і має синдром Hx−m, оскільки метою є зробити мінімальні зміни в векторі x. Після обчислення Hx−m Аліса перетворює цей вектор у число і у десятковій системі числення. Це число відповідає і-му біту в x, який необхідно змінити.

Розглянемо простий приклад з кодом Хеммінга, щоб пояснити концепцію. Припустимо, що повідомлення $m=(1,0,1,1)^T$ передається приймачу. Код Хеммінга (7, 4) використовується для кодування повідомлення на стороні приймача та, за потреби, виправлення помилок.

Спершу кодове слово y для повідомлення m обчислюється за допомогою рівняння (2.2):

$$y = \left(\begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \end{pmatrix} \times \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} \right)^T = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \end{pmatrix} \quad (2.7)$$

Приймач отримує кодове слово x. Припустимо, що є помилка, така що отримане слово $x= (1, 0, 1, 1, 1, 0, 1)^T$. Тепер рівняння (2.4) використовується для обчислення синдрому z, щоб перевірити, чи є якась помилка в x:

$$z = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 \end{bmatrix} \times \begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \quad (2.8)$$

z =(1, 0, 0)^T , z інтерпретується як 5, який ідентичний 5-му стовпцю в H. Виправлення (1, 0, 1, 1, 1, 0, 1)^T $\oplus$ (0, 0 , 0, 0, 1, 0, 0)^T = (1, 0, 1, 1, 0, 0, 1)^T = y.

У стеганографії, з підходом вбудовування матриці, повідомлення, яке буде вбудовано в набір квантованих значень DCT, пов'язане із синдромом. Отже, для повідомлення розміром k нам потрібен (2^k , 2^k − k) код Хеммінга. Залишаючись у тій самій концепції Хеммінга, Аліса використовує код Хеммінга, щоб знайти позицію, необхідну для зміни, щоб імітувати ту саму зміну (помилку) у повідомленні призначення.

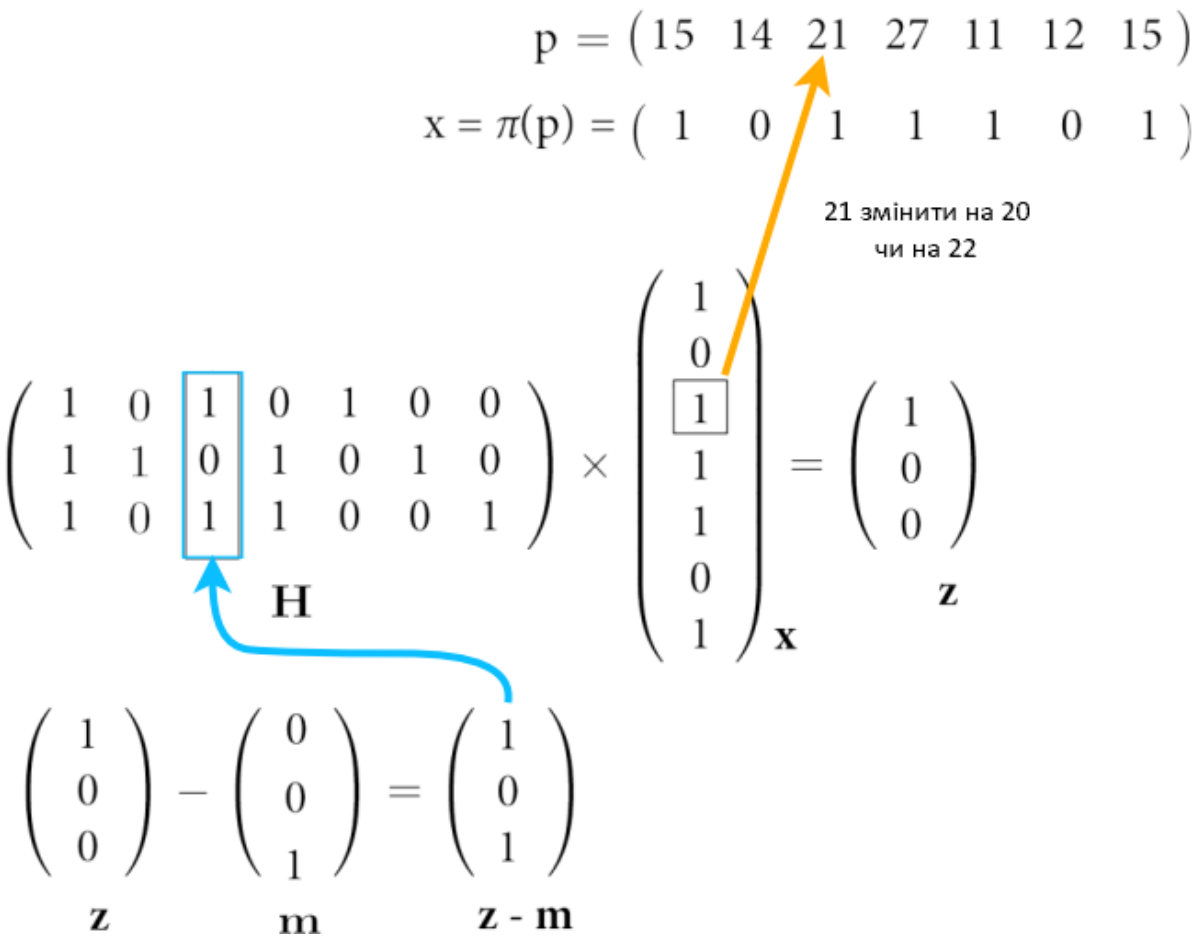

Рисунок 2.4 – Вбудовування за допомогою бінарного Хеммінга

Наведемо приклад вбудовування матриці для набору квантованих значень DCT p = (15, 14, 21, 27, 11, 12, 15) і повідомлення m =(0, 0, 1)^T .

Приклад показано на рисунку 2.4. Аліса спочатку перетворює квантовані значення DCT в біти, $x = p \bmod 2 = (1, 0, 1, 1, 1, 0, 1)^T$; значення x, таким чином, є LSB від p. Далі Аліса обчислює синдром x, який дає $z = (1, 0, 0)^T$, використовуючи матрицю H із попереднього прикладу. Потім вона обчислює вектор $z-m = (1, 0, 1)^T$ і намагається знайти його між стовпцями в H. Це третій стовпець. Таким чином, усе, що їй потрібно, щоб вбудувати m у блок, це перевернути LSB третього квантованого значення DCT. Наприклад, вона може змінити p[3] = 21 на 20 або з 21 на 22 вона, таким чином, отримає однаковий результат для обох варіантів.

2.2 Огляд застарілих алгоритмів у JPEG

У цьому розділі розглядаємо найвідоміші алгоритми стеганографії, спеціально розроблені для JPEG-зображень, починаючи з моменту, коли стеганографія стала окремою науковою дисципліною. До перших і найвивченіших алгоритмів належать Jsteg [9], JPHide&Seek 1998 року від Аллана Латема, Outguess [10] та F5 [11], які використовуються для вбудовування даних у файли JPEG. Вбудовування здійснюється у квантизовані DCT-коефіцієнти.

У 1997 році було створено перший стеганографічний алгоритм для JPEG, Jsteg. У своїй першій версії алгоритм використовував техніку LSB-replacement для вбудовування повідомлення у послідовному порядку, тобто без застосування стего-ключа. Пізніше Jsteg зазнав покращень завдяки розробці алгоритму JPEG-Jsteg, у якому повідомлення спочатку шифрувалося, а потім послідовно вбудовувалося у стандартному порядку зигзагового сканування за допомогою техніки LSB-replacement. Алгоритм вбудовував повідомлення в молодші значущі біти (LSB) квантизованих DCT-коефіцієнтів, значення яких не належали до {0, 1, -1}.

[12] покращили алгоритм JPEG-Jsteg, збільшивши розмір прихованого повідомлення, зберігаючи той самий рівень безпеки. Метод полягав у

вбудовуванні секретного повідомлення в середньочастотну частину квантизованих DCT-коефіцієнтів замість використання стандартного порядку зигзагового сканування рисунок 2.5. Ще одне покращення алгоритму JPEG-Jsteg було зроблено у версії JPEG-Jsteg-v4.

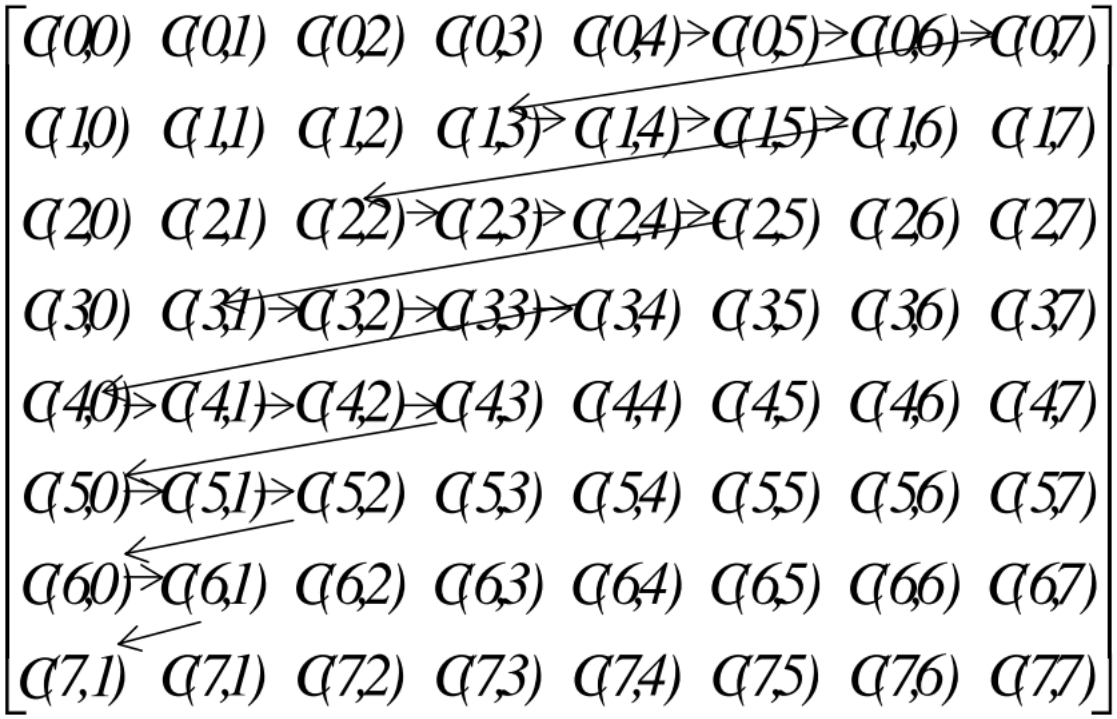

Рисунок 2.5 – Послідовність вбудовування з [12]

Загалом удосконалення стеганографічних схем полягало у наближенні стегозображення до вихідного зображення. Іншими словами, ідеальна стеганографічна схема повинна зберігати всі статистичні властивості зображення, але складно отримати відповідні статистичні моделі зображень.

2.3 Адаптивна стеганографія в JPEG

В адаптивній стеганографії Аліса адаптує процес вбудовування, щоб приховати секретне повідомлення у "безпечних" областях, тобто там, де зміни важче виявити на основі статистики квантизованих DCT-коефіцієнтів

зображення. Ідея адаптивної стеганографії бере початок з перших днів цифрової стеганографії.

Зазвичай, для зображень у градаціях сірого, Аліса виконує процес вбудовування, починаючи з вибору квантизованих значень DCT зображення $x = (x_1, ..., x_n) \in \{-1023, ..., 1024\}^n$, щоб вбудувати секретне повідомлення $m=(m_1,...,m_m)\in\{0,1\}^{|m|}$, з метою створення стего-зображення з квантизованими значеннями DCT $y=(y_1,...,y_n)\in\{-1023,...,1024\}$.

Потім Аліса вибирає найбезпечніші області, створюючи карту вартості ρ, яка моделює вплив вбудовування з точки зору безпеки для кожного квантизованого DCT-коефіцієнта. Карта ρ допомагає Алісі вбудувати повідомлення у специфічні позиції, де статистичні зміни виявити важче. Вона обирає ті квантизовані DCT-коефіцієнти, які мають найслабші значення ρ, і які дозволяють закодувати та вбудувати повідомлення.

Вплив вбудовування може бути змодельований функцією, яка вимірює відстань між прихованим зображенням x та відповідним стего-зображенням і може бути формально представлене таким чином:

$$\begin{gathered} D : \{-1023, ..., 1024\}^n \times \{-1023, ..., 1024\}^n \to R \\ D(x,y) = \|f(x) - f(y)\|, \end{gathered} \quad (2.9)$$

Значення f є функцією, яка повертає вектор дескрипторів, що описують зображення. Ці дескриптори кодують цінну інформацію про зображення та діють як своєрідний цифровий "відбиток", який може використовуватися для розрізнення однієї характеристики від іншої.

Це математичне формулювання функції спотворення (2.9) створює суттєву проблему для стеганографа, оскільки вона є неадитивною та нелокальною. Щоб подолати та спростити цю проблему, одним із підходів є припущення, що зміна квантизованого значення DCT не впливає на виявленість сусідніх квантизованих значень DCT. Це дозволяє модифікувати функцію спотворення в адитивну версію. У цьому контексті пропонують

адитивну версію функції спотворення, яка використовує карту вартості.

$$\rho = \{\rho_i \in [0, \infty]\}_{i=1}^{n} . \quad (2.10)$$

Принцип полягає в тому, щоб призначити для кожного квантизованого значення DCT зображення x_i вартість виявлення $\rho_i \in [0, \infty]$. Це значення моделює вплив зміни і-го квантизованого значення DCT прихованого зображення на загальну безпеку. Ці вартості формують так звану карту вартості. Розглядаючи бінарне вбудовування ($|x_i - y_i| \leq 1 |x_i - y_i|$), (2.9) можна переписати наступним чином:

$$D(x, y) = \sum_{i=0}^{n} \rho i \, |x_i - y_i|, \quad (2.11)$$

де D(x,y) вимірює вплив вбудовування, викликаний процесом вбудовування. Мінімальне очікуване спотворення для заданого повідомлення m представлено у такій формі:

$$min\, D(x, y) = \sum_{i=0}^{n} \rho_i \, \psi_i , \quad (2.12)$$

де ψ_i– це ймовірність модифікації і-го квантизованого значення DCT, яка безпосередньо пов’язана зі значенням виявлюваності ρ_i квантизованого DCT-значення на позиції і, оскільки ψ_i визначається наступним чином:

$$\psi_i = \frac{e^{-\lambda \rho i}}{1 + e^{-\lambda \rho i}}, \quad (2.13)$$

де $\lambda > 0$ — це константа, що визначається обмеженням на розмір повідомлення:

$$-\sum_{i=0}^{n}\left(\psi_i \log_2 \psi_i + \left(1 - \psi_i\right) \log_2 \left(1 - \psi_i\right)\right) = |m|. \qquad (2.14)$$

Карта ймовірностей модифікації може бути визначена наступним чином: $p=\{\psi_i \in R_+\}_i$, де квантизовані DCT-значення $\{x_i\}^n_{i=1}$ із вищою ймовірністю модифікації $\{\psi_i\}^n_{i=1}$ мають більшу ймовірність бути зміненими.

На даний момент можна швидко зробити висновок, що значною складністю цього підходу є обчислення карти вартості ρ $\rho_i \in [0, \infty]$. Обчислення значень ρ_i, які відображають вплив процесу вбудовування на безпеку, залишається відкритим питанням.

3 ПАКЕТНА СТЕГАНОГРАФІЯ

3.1 Перспективи пакетної стеганографії

Стеганографія традиційно зосереджувалася на вбудовуванні повідомлення в одне приховане зображення за раз. Однак набагато реалістичніше для Аліси приховати повідомлення, розподіливши його між кількома зображеннями, тобто послідовністю зображень. Цей процес розподілу називається пакетною стеганографією і протиставляється об'єднаному стегоаналізу, де аналізується набір зображень, щоб зібрати низку підказок і зробити висновок про наявність або відсутність прихованого повідомлення у наборі зображень. Теми пакетної стеганографії та об'єднаного стегоаналізу були вперше запропоновані у [13] і стали однією з найбільш складних відкритих проблем у цій галузі за останні роки .

Тема пакетної стеганографії проілюстрована на рисунку 3.1. У першій статті [13] автор наводить приклад такого сценарію: припустимо, злочинець хоче приховати інформацію на своєму комп’ютері, використовуючи стеганографію, що є безсумнівним фактом. У нього вже є велика кількість прихованих зображень на жорсткому диску. Щоб переконатися, що він добре приховує свою секретну інформацію, він може розділити її на кілька невеликих частин і сховати трохи в кожному з обраних зображень (пакетна стеганографія), вважаючи, що це безпечніше, ніж альтернатива, коли велика кількість даних заповнює меншу кількість зображень.

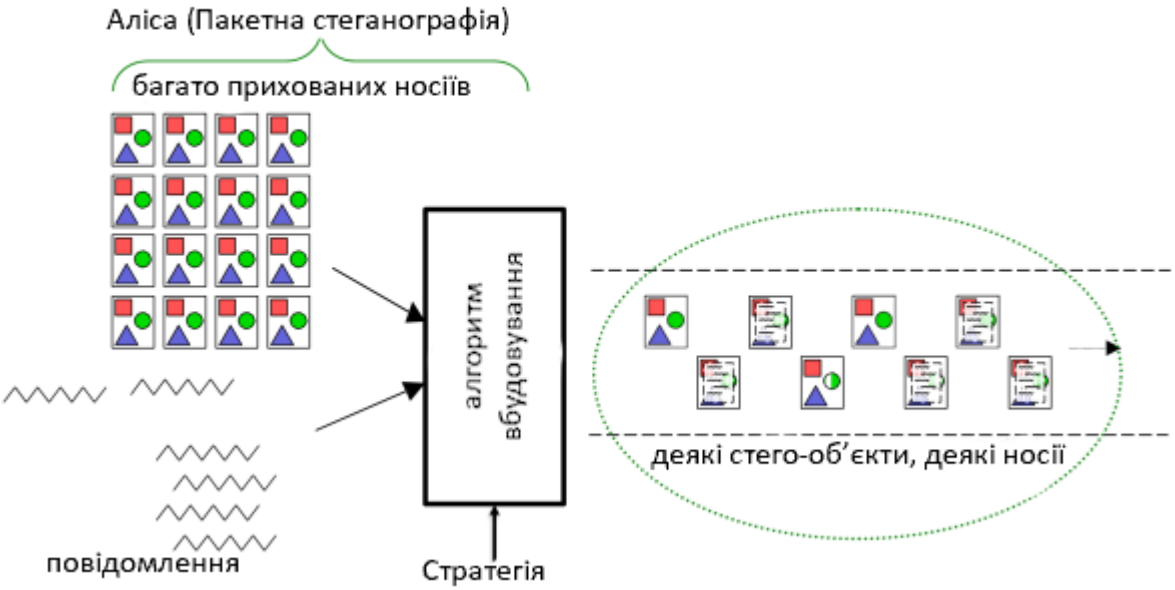

Рисунок 3.1 – Схема, що ілюструє процес пакетної стеганографії

У [13] Кер запропонував теоретичну базу для раніше не дослідженої теми пакетної стеганографії. Для створення цієї бази Кер висунув кілька гіпотез, зокрема такі: передбачається, що Єва знає розмір набору зображень (bag), який є фіксованим, а також повідомлення має фіксовану довжину. Крім того, Кер припустив, що всі приховані зображення мають однакову вбудовувальну здатність, тобто в усі зображення можна вбудувати повідомлення з фіксованою максимальною довжиною.

Ендрю Кер продовжив теоретичні дослідження з цієї теми та порівняв ефективність різних стратегій розподілу на основі неадаптивних схем вбудовування, коли детектор стегоаналізу створено для атаки конкретної схеми вбудовування. У цьому випадку Кер показав, що оптимальна стратегія розподілу полягає або в концентрації повідомлення в якомога меншій кількості прихованих зображень (жадібна стратегія), або, навпаки, у максимальному розподілі повідомлення між якомога більшою кількістю зображень (лінійна стратегія).

3.2 Теоретичні визначення

Згадуючи визначення стеганографічного каналу, де Аліса використовує приховане зображення для вбудовування свого повідомлення за допомогою схеми вбудовування та стего-ключа. Узагальнюючи це визначення, у пакетній стеганографії Алісі потрібен набір прихованих JPEG-зображень для вбудовування свого повідомлення із застосуванням схеми вбудовування, стего-ключа та стратегії розподілу. Усі ці елементи представлені на рисунку 3.2.

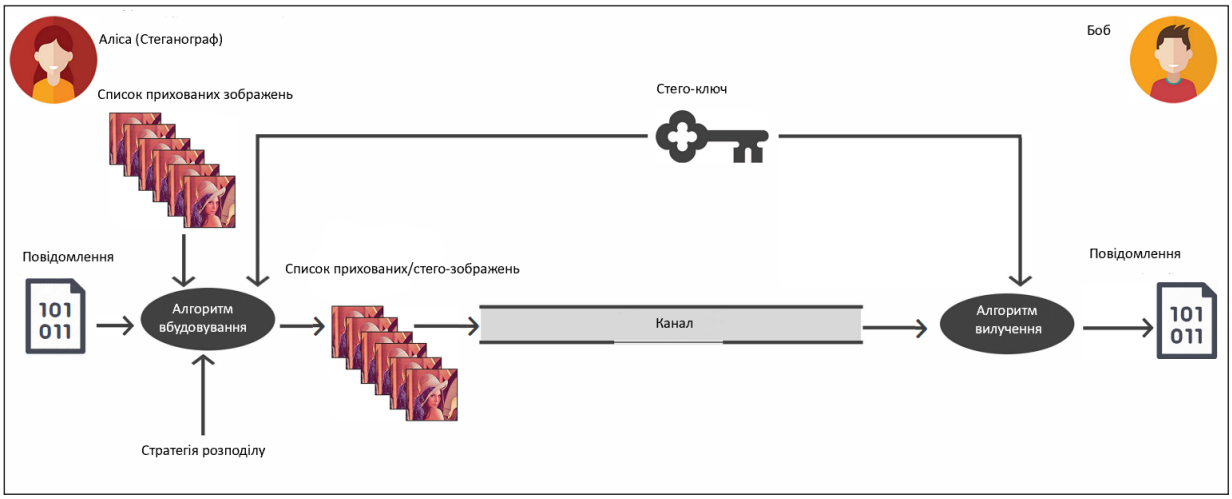

Рисунок 3.1 – Стеганографічний канал та основні елементи для пакетної стеганографії.

Маючи набір послідовностей прихованих зображень $\{X_i\}$, набір ключів K, пов’язаних із прихованими зображеннями (по одному ключу для кожного зображення), набір повідомлень M, які Аліса може вбудовувати, і набір стратегій S, які Аліса може використовувати для розподілу повідомлення, процес вбудовування та вилучення можна виразити таким чином:

$$\begin{aligned} Emb &: \{X_i\}_{i=1}^{i=b} \times K \times M \times S \rightarrow \{X_i\}_{i=1}^{i=b} \\ Emb &: \{X_i\}_{i=1}^{i=b} \times K \times S \rightarrow M \end{aligned}. \quad (4.1)$$

Таким чином, пакетна стеганографія полягає у вбудовуванні повідомлення $m \in \{0,1\}^{|m|}$ у послідовність прихованих зображень із використанням стратегії розподілу $s \in S$.

3.3 Стратегії пакетного розподілу

Метою пакетної стеганографії є те, що Аліса прагне розподілити в послідовності B з b зображень повідомлення із загальним відносним навантаженням α. Значення α пов'язане з відносним навантаженням α_i кожного зображення в послідовності. α_i виражається в бітах на піксель (bpp) для просторових зображень, у бітах на ненульові AC-компоненти (bpnzAC) або бітах на коефіцієнт (bpc) для зображень у форматі JPEG. Довжина повідомлення |m| розподіляється між послідовністю зображень $(x_1,...,x_b)$ таким чином, що довжини фрагментів повідомлення $|m|=(|m_1|,...,|m_b|)$ визначаються наступним чином:

$$|m| = \sum_{i=1}^{b} |m_i|. \quad (3.2)$$

Значення $|m_i||$ також залежить від стеганографічної ємності c_i, яка є максимальною кількістю даних, що Аліса може вбудувати в зображення x_i. Варто зазначити, що стеганографічна ємність є нечітко визначеним поняттям, яке вказує на розмір даних, які можна безпечно вбудувати в зображення, використовуючи конкретну схему вбудовування.

c_i може визначатися різними способами залежно від стратегії розподілу та схеми вбудовування, яка використовується для вбудовування даних. Ці визначення будуть розглянуті далі.

Далі обговоремо стратегії розподілу. У жадібній стратегії Аліса ітеративно вибирає приховані зображення з найбільшою ємністю, які ще не використовувалися, і вбудовує частину повідомлення, що дорівнює оціненій ємності зображення. Після впорядкування зображень за ємністю $c_1 \geq c_2 \geq ... \geq c_b$, Аліса застосовує наступне рівняння:

$$\begin{cases} |m_i| = c_i, \forall i \in \{1, ..., I\}, \\ |m_I| = |m| - \sum_{i=1}^{I=1} |m_i| \\ |m_i| = 0, \forall i \in \{I + 1, ..., b\}, \end{cases} \tag{3.3}$$

де I позначає найменшу можливу кількість зображень із достатньою ємністю, тобто:

$$I = arg \min_i \sum_{j=1}^{i} c_j \geq m, \tag{3.4}$$

де c_i – це максимальна ємність прихованого зображення x_i. У своїх емпіричних експериментах Ker використовував схему вбудовування LSB-replacement, а значення c_i були максимальним обсягом замін, які алгоритм міг виконати, що призводило до фіксованого значення c_i. Крім того, автор розглядав фіксовану кількість даних у кожному зображенні.

У своїх подальших роботах Ker і Pevný, ємність обчислювалася на основі схеми вбудовування, автори використовували значення ємності, оцінене схемою вбудовування під час самого процесу. Наприклад, використовувалися такі схеми, як F5, JPHideSeek і OutGuess. У випадках, коли схема вбудовування не надавала оцінки ємності, автори встановлювали ємність на фіксоване значення. Наприклад, у схемі nsF5 автор встановив оцінку ємності на рівні 0.8 bpnc (біти на ненульовий коефіцієнт).

Автори також запропонували інший спосіб вибору зображень для вбудовування в рамках жадібної стратегії, який імітує ситуацію, коли Аліса

не має попередньої інформації про всі ємності зображень, і тому не може вибрати зображення з високою ємністю. У цьому випадку Аліса випадково обирає зображення x_i для вбудовування, оцінює його ємність і вбудовує частину повідомлення, що дорівнює c_i. Це може призводити до використання більшої кількості зображень. Автори назвали цей підхід "випадковою стратегією" (random strategy).

Лінійна стратегія також пов’язанаа з ємністю, розподіляючи повідомлення m між усіма доступними прихованими зображеннями пропорційно до їхньої ємності. Це формулюється наступним рівнянням, ігноруючи дробові біти у |m|:

$$\left|m_i\right| = \frac{c_i \left|m\right|}{\sum_{j=1}^{b} c_j}. \tag{3.5}$$

Не враховуючи ємності, рівняння (3.5) набуває наступного вигляду:

$$\left| m_i \right| = \frac{\left|m\right|}{b}. \tag{3.6}$$

Для останнього рівняння повідомлення розподіляється рівномірно між усіма зображеннями в наборі, так що розмір повідомлення для кожного зображення дорівнює довжині повідомлення, поділеній на кількість прихованих зображень. Автори назвали це лінійною стратегією, також відомою як рівномірна стратегія (even strategy). Варто зазначити, що в пізніших роботах автори заявляли, що використовували лінійну стратегію у своїх експериментах незалежно від ємності зображень, оскільки вбудовування виконувалося за допомогою сучасних адаптивних схем вбудовування. У решті цього документа будемо посилатися на лінійну стратегію як на ту, що рівномірно розподіляє дані між b зображеннями.

Останньою стратегією, запропонованою авторами, була стратегія квадратного кореня (sqroot strategy), де повідомлення розподіляється між усіма b зображеннями так, що довжина фрагментів пропорційна квадратному кореню їхньої ємності, тобто:

$$|m_i| = \frac{\sqrt{c_i}\ |m|}{\sum_{j=1}^{b} \sqrt{c_i}}. \tag{3.7}$$

Жадібна (greedy) та лінійна (linear) стратегії для розподілу повідомлення обговорюються у статті [14]. Автори припускають, що Аліса не застосовує жодної стратегії вибору носіїв, тобто приховані зображення в послідовності обираються випадковим чином відповідно до розподілу ймовірностей носіїв. Крім того, вони спростили задачу розподілу повідомлення, не враховуючи ємність прихованих зображень.

Додатково автори запропонували стратегію β-використання (uses-β, яка є поєднанням жадібної та лінійної стратегій. Ця стратегія формулюється наступним рівнянням:

$$|m_i| = \frac{|m_i|}{\beta b}, \tag{3.8}$$

le $\beta \in [0,1]$ – це частка доступних прихованих зображень, між якими Аліса рівномірно розподіляє повідомлення. Ця стратегія еквівалентна лінійній стратегії для β=1. Водночас вона також схожа на жадібну стратегію, оскільки розподіляє повідомлення у частині зображень.

В статті [15] запропонували три практичні стратегії розподілу, які перевершили всі попередньо описані стратегії. Ці стратегії були розроблені з метою мінімізації статистичної виявлюваності вбудованих зображень.

Перша стратегія – це стратегія злиття зображень відправником (IMS), у якій Аліса створює унікальне зображення з усіх b зображень із послідовності B, і дозволяє схемі вбудовування (використовується адаптивна схема вбудовування) розподілити повідомлення по всьому цьому "великому" зображенню. Зрештою, створюються зображення однакового розміру для формування великого зображення.

Після вбудовування стего-зображення вихідних зображень відновлюються. Процес конкатенації ілюструється на рисунку 3.2.

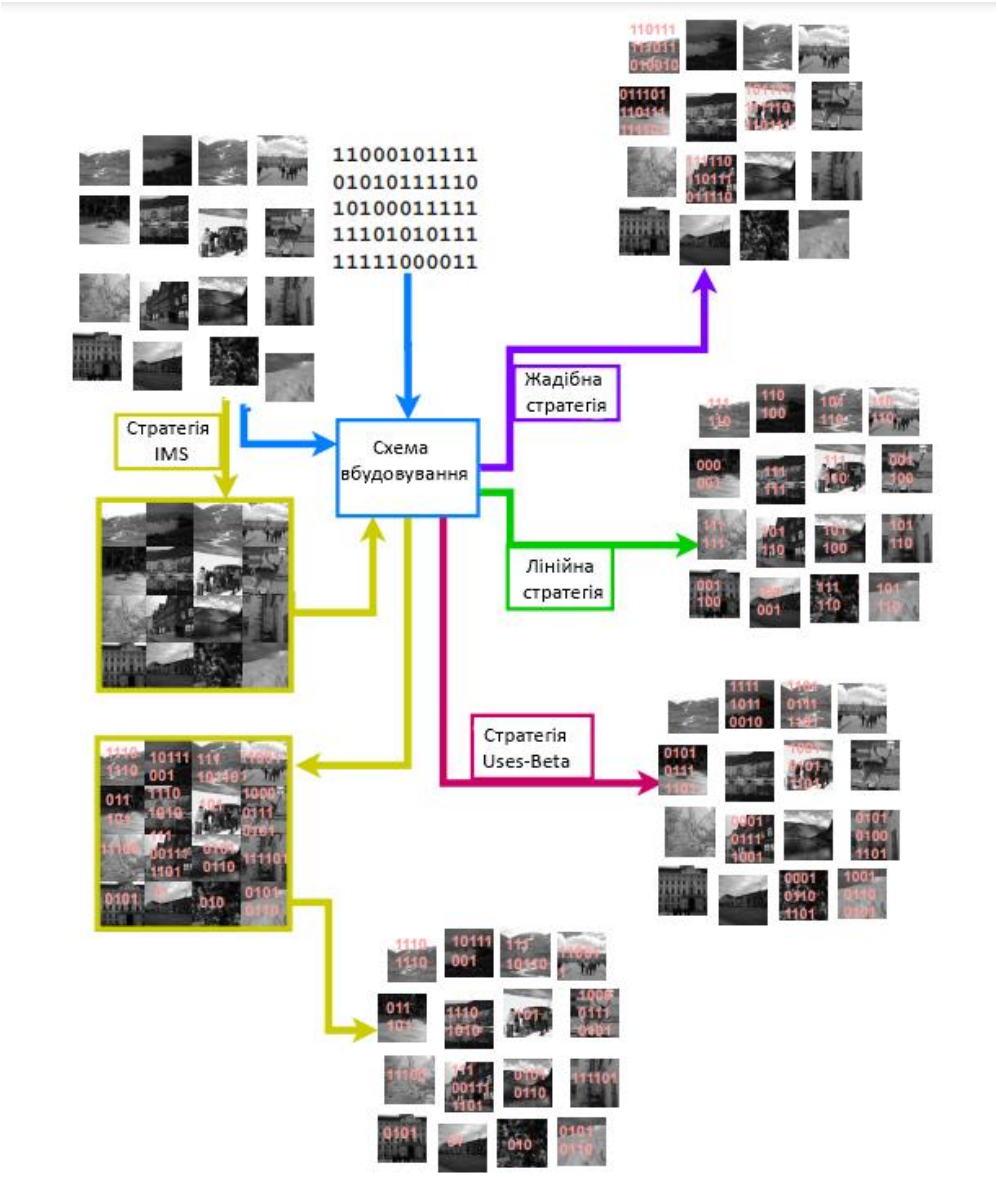

Рисунок 3.2 – Схема, яка ілюструє чотири стратегії пакетного розповсюдження

У стратегії IMS значення вартості обчислюється для кожного DCT-коефіцієнта "великого" зображення (адаптивна схема вбудовування визначає спосіб обчислення карти вартості), а саме вбудовування виконується за допомогою STC.

Друга стратегія – це стратегія відправника з обмеженою виявлюваністю (DeLS), яка працює з урахуванням статистичної виявлюваності. У цій стратегії Аліса приймає модель прихованого зображення і розподіляє повідомлення між b зображеннями, які передають необхідне повідомлення, так, щоб кожне зображення з послідовності вносило однакове значення дивергенції Кульбака-Лейблера (KL) (коефіцієнт відхилення), базоване на моделі MiPOD.

Варто зазначити, що дивергенція KL є фундаментальною мірою безпеки, яка застосовується в галузі стеганографії та стегоаналізу. Модель MiPOD є підходящою в цьому випадку, оскільки вона базується на мінімізації статистичної виявлюваності. Ця схема була створена для роботи в просторі, і для заданого зображення x_i із n пікселів Аліса вибирає змінні K_i, $i \in \{1,...,n\}$, які мінімізують коефіцієнт відхилення, визначений наступним чином:

$$\varsigma = \sqrt{2\sum_{i=1}^{n} \sigma^{=4}{}_i k_i^2} \,. \tag{3.6}$$

Швидкості змін вбудовування спочатку обчислюються шляхом чисельного розв'язання наступних n+1 рівняння (3.7):

$$\kappa_i \sigma_i^{-4} = \frac{1}{2\lambda} \ln \frac{1-2k_i}{k_i} \tag{3.7}$$

та рівняння (4.8):

$$R = \sum_{i=1}^{n} H(k_i). \tag{3.8}$$

Де σ_i – це дисперсія пікселя прихованого зображення x_i, яка обчислюється за допомогою оцінювача дисперсії, λ — множник Лагранжа, а $H(x)=-2x\ln x-(1-2x)\ln(1-2x)$ — це функція тернарної ентропії, виражена в натах. Після обчислення швидкостей дисперсії за допомогою методу множників Лагранжа, вони перетворюються у вартості $\rho_i, i\in\{1,...,n\}$ за допомогою наступного рівняння:

$$\rho_i = ln(\frac{1}{k_i} - 2). \tag{3.9}$$

У MiPOD ці вартості ρ_i потім використовуються кодами трельового синдрому (Syndrome Trellis Codes) для вбудовування повідомлення з навантаженням R під час процесу вбудовування.

Варто зазначити, що було запропоновано новий підхід для розширення використання MiPOD на JPEG-зображення. Новий алгоритм отримав назву J-MiPOD і базується на перетворенні зображення в просторову область для оцінки значень σ_i, застосуванні лінійного перетворення для обчислення дисперсії коефіцієнтів DCT σ_i, що дозволяє продовжити процедуру вбудовування MiPOD на коефіцієнтах JPEG-зображення.

Принцип стратегії розподілу DeLS полягає в тому, щоб для заданої послідовності прихованих зображень $\{x_i\}_{i=1}^{b}$ встановити однакове значення для всіх $\{x_i\}_{i=1}^{b}$. Це дозволяє визначити для кожного зображення швидкість змін n. Потім Аліса бере розраховані навантаження для зображень у наборі та використовує їх для створення набору зображень із застосуванням обраної схеми вбудовування.

Останньою стратегією є стратегія відправника з обмеженим спотворенням (Distortion Limited Sender, DiLS). У цій стратегії Аліса

розподіляє повідомлення між зображеннями таким чином, щоб кожне зображення в наборі вносило однакове значення спотворення. Спотворення кожного зображення обчислюється за допомогою функції спотворення обраної адаптивної схеми вбудовування.

Також було запропоновано нову стратегію розподілу. Ця стратегія близька до IMS і отримала назву Adaptive Batch size Image Merging steganographer (AdaBIM). Вона відрізняється від стратегії IMS тим, що AdaBIM розподіляє повідомлення нерівномірно між усіма зображеннями залежно від міри придатності зображення для стеганографії, яка визначається як:

$$\sqrt[n]{\prod_{i=1}^{n}\sigma_i^2}\ . \tag{3.10}$$

Де n – це кількість пікселів у зображенні, а σi — дисперсія i-го пікселя.

Автори формалізують процес розподілу (вбудовування у "велике" зображення) наступним чином: для заданого набору зображень X і набору (bag) розміром b. Припускається, що l-й набір містить зображення з індексами (l−1)b,...,lb−1. Стратегія IMS використовується для розподілу n×b×αn натів серед b зображень у кожному наборі. Для всіх j∈(l−1)b,...,lb−1, обсяг повідомлення для j-го зображення розраховується за наступним рівнянням:

$$\alpha_j = n\alpha + \frac{\alpha}{2}\ln\left(\frac{\sqrt[n]{\prod_{i=1}^{n}\sigma_i^2}}{\sqrt[b]{\prod_{k=(l-1)b}^{lb-1}\sqrt[n]{\prod_{i=1}^{n}\sigma_{ik}^2}}}\right),$$

де σij— це дисперсія i-го пікселя j-го зображення.

3.4 Стратегії пакетного розподілу та загальна архітектура об’єднаного стегоаналізу

Нагадаємо, що вбудовування у набір зображень (bag) здійснюється при заданому розмірі навантаження (payload), який виражається в бітах на загальну кількість коефіцієнтів (bptc). Значення bptc визначається як розмір повідомлення в бітах, поділений на загальну кількість пікселів (тобто коефіцієнтів AC та DC) усіх зображень у наборі.

Було застосовано наступні стратегії пакетного розподілу:

- жадібна стратегія (Greedy Strategy);
- лінійна стратегія (Linear Strategy);
- стратегія Uses-β (Uses-β Strategy);
- стратегія злиття зображень (Image Merging Sender, IMS);
- стратегія обмеженої виявлюваності (Detectability Limited Sender, DeLS);
- стратегія обмеженого спотворення (Distortion Limited Sender, DiLS).

Тести виконуються із використанням 2 500 прихованих зображень, які раніше не використовувалися, що дозволяє сформувати набір із 5 000 пар прихованих/стего-наборів для фіксованого розміру b та обраної стратегії розподілу (IMS, DeLS, DiLS, Greedy, Linear та Uses-β). Потім середню ймовірність помилки, отриману кожною функцією об’єднання, обчислюють за результатами 10 запусків, кожен із яких виконується з різними навчальними наборами (та тестовими наборами) із 5 000 пар прихованих/стего-наборів, і представляють у результатах.

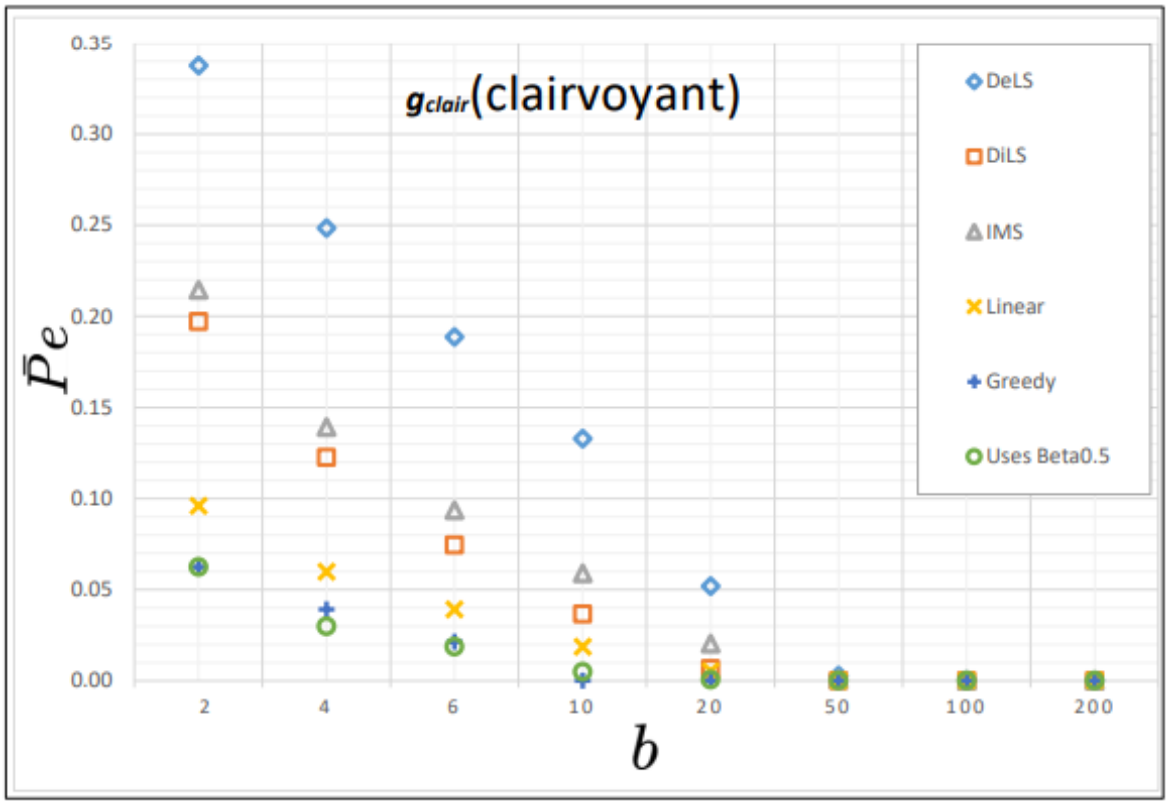

Рисунок 3.3 – Порівняння стратегій розподілу у випадку передбачливого аналізу

На рисунку 3.3 представлені результати, отримані у сценарії передбачливого стегоаналізу, тобто з використанням g_{clairg}. Помітно, що стратегія DeLS є найкращою і перевершує IMS, яка була більш конкурентною до DeLS . Водночас, Greedy є найгіршою стратегією.

Стратегії можна класифікувати на 3 групи:

Група 1. Greedy і Uses-β\betaβ: Найбільш виявлювані стратегії. Їхня виявлюваність досягає значення Pe≈0 для розмірів набору b≥10.

Група 2. DeLS, IMS і DiLS: Найбезпечніші стратегії, які мають меншу виявлюваність.

Група 3. Лінійна стратегія (Linear): Займає проміжне місце між двома іншими групами.

Однак, стратегії DeLS та IMS стають повністю виявлюваними при b=100, коли Pe≈0.

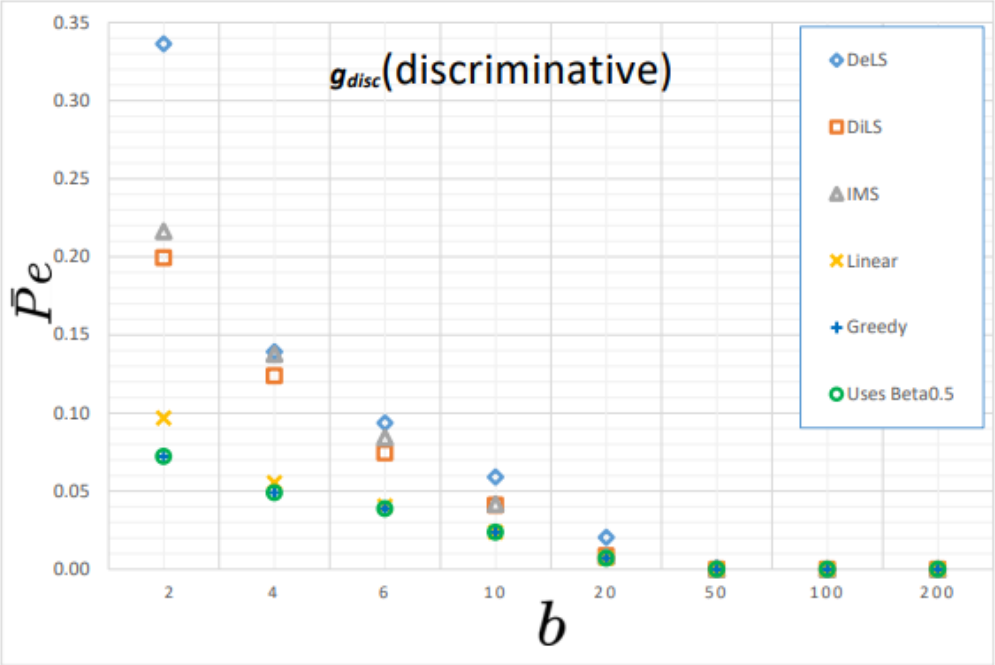

Рисунок 3.4 – Порівняння стратегій розподілу у дискримінативному випадку. Середня ймовірність помилки при рівних апріорних ймовірностях (Pe) як функція розміру набору b∈B для середнього навантаження 0.1 bptc із використанням дискримінативної функції об'єднання gdiscg, навченої на всіх стратегіях.

На рисунку 3.4 представлено виявлення кожної стратегії розподілу за допомогою дискримінативної функції об'єднання gdisc. Найкращі стратегії у порядку спадання їх здатності протистояти функції об'єднання, яка намагається дискримінувати їх: DeLS, IMS, DiLS, Linear, Uses-β, Greedy.

Стратегія DeLS знову демонструє хороші результати в цьому (non-clairvoyant) підході, залишаючись стійкою до b=100, коли середня ймовірність помилки Pe починає збігатися з іншими стратегіями і стає приблизно рівною 0. DeLS та IMS більш помітні за допомогою дискримінативної функції gdisc, ніж за допомогою передбачливої функції gclair.

Розглядаючи гістограми оцінок SVM рисунок 3.5, дійсно спостерігається більший розподіл з gdisc.

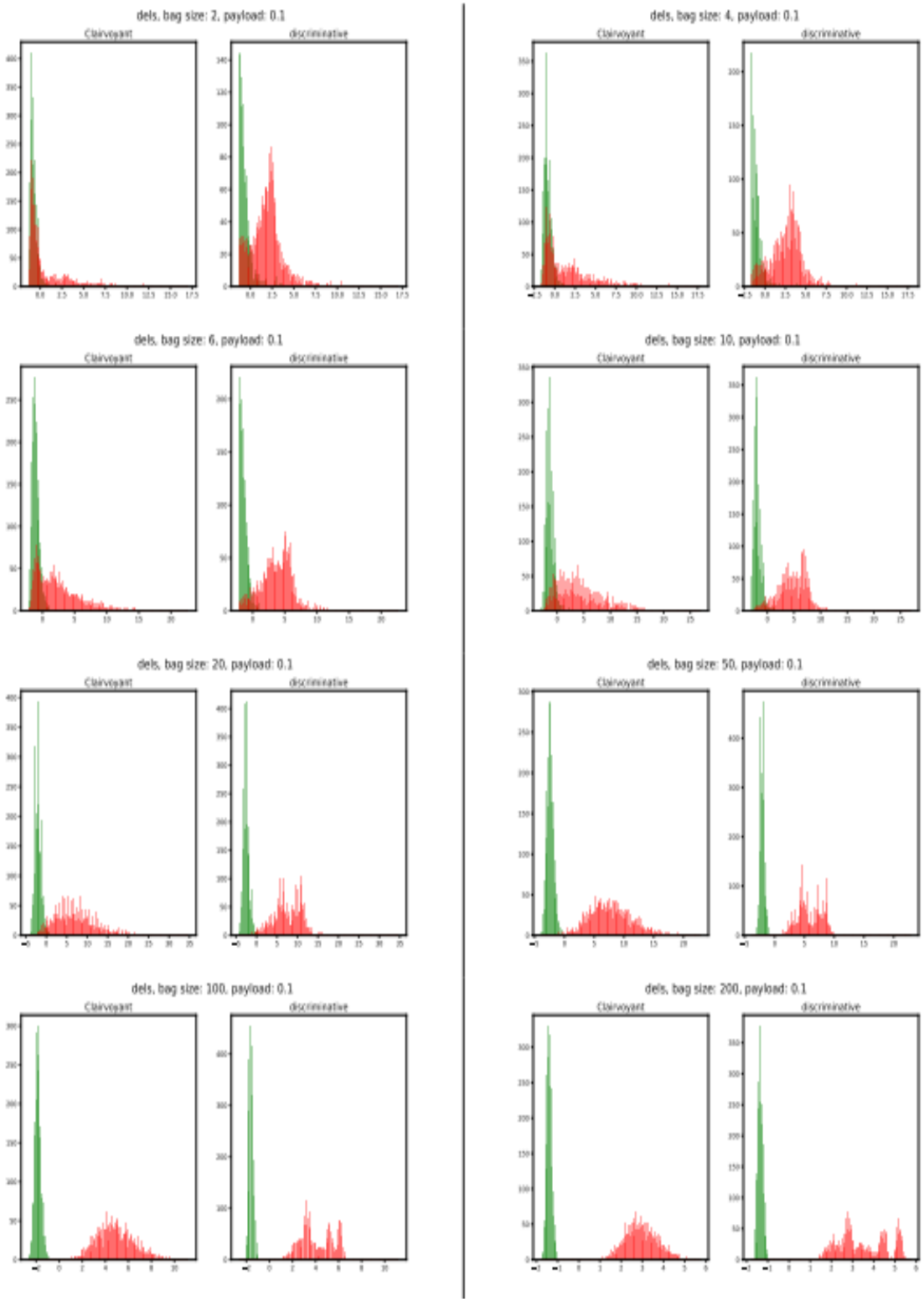

Рисунок 3.5 – Стратегія розподілу DeLS

На рисунку 3.4 також видно, що розрив між значеннями Pe для кожної стратегії менший порівняно з рисунком 3.3.

Ця поведінка, ймовірно, пояснюється тим, що оптимізація (навчання SVM) намагається мінімізувати помилку прогнозування рівномірно для кожної стратегії.

На рисунку 3.6 представлено важливу інформацію про точність методів об'єднання. Показано середню ймовірність помилки при рівних апріорних ймовірностях Pe як функцію розміру набору b∈B для середнього розміру навантаження R=0.1 bptc для кожної функції об'єднання за всіма стратегіями.

Відзначається, що дискримінативна функція об'єднання gdisc перевершує функції gmean та gmax із середньою різницею Pe≈2%P та є ближчою до передбачливої функції gclair із середньою різницею Pe≈0.8%.

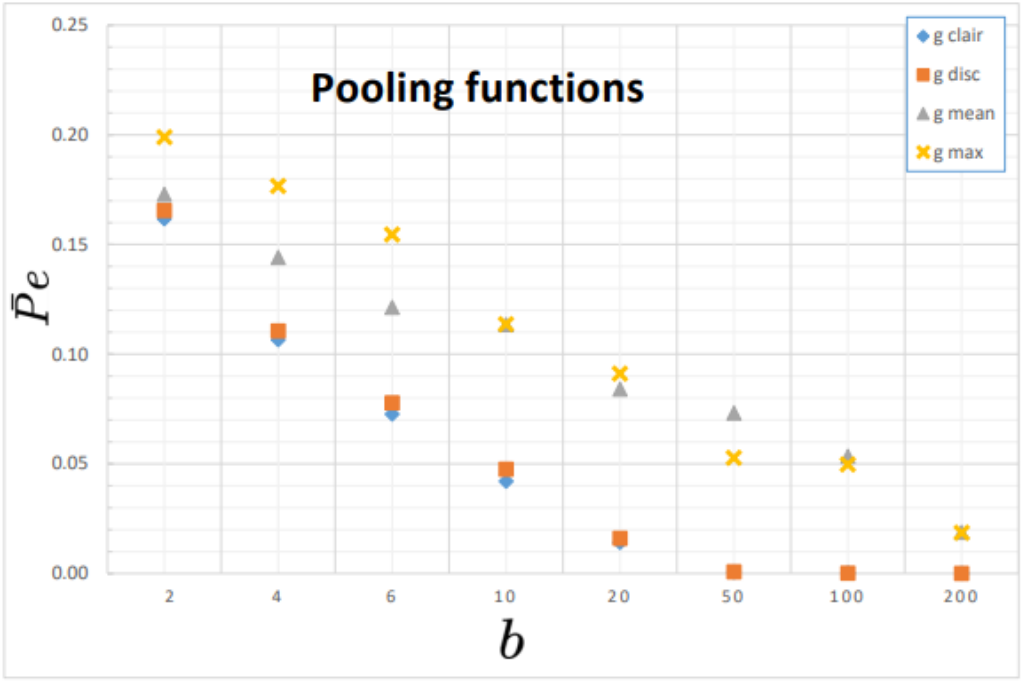

Рисунок 3.6 – Порівняння об'єднаного стегоаналізу

З рисунка 3.6 також видно, що функції gdisc та gclair є стабільнішими, ніж gmean і gmax. Це узгоджується з гіпотезою цього розділу, що дискримінативна функція об'єднання дозволяє отримати кращі результати виявлення порівняно з середньою mean та максимальною max функціями об'єднання.

ВИСНОВКИ

У даній кваліфікаційній роботі було проведено дослідження методів пакетної стеганографії та об'єднаного стегоаналізу для форматів зображень JPEG. Основна увага приділялася адаптивним стратегіям розподілу навантаження між зображеннями та ефективності виявлення прихованих даних за допомогою сучасних методів стегоаналізу.

Основні результати дослідження:

- було розглянуто шість основних стратегій розподілу повідомлень, таких як жадібна, лінійна, Uses-β, IMS, DeLS та DiLS. Встановлено, що адаптивні стратегії, такі як DeLS та IMS, забезпечують значно кращу стійкість до виявлення, особливо при використанні сучасних адаптивних схем вбудовування;

- розглянуто дискримінативну функцію об'єднання, яка дозволяє ефективно аналізувати набори зображень навіть за відсутності інформації про стратегію розподілу. Експерименти показали, що ця функція об’єднання наближається за точністю до передбачливого стегоаналітика (gclair).

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Zhu, Wen-Han, et al. "Structured computational modeling of human visual system for no-reference image quality assessment." International Journal of Automation and Computing 18 (2021): 204-218.

2. Rahman, Md Atiqur, Mohamed Hamada, and Jungpil Shin. "The impact of state-of-the-art techniques for lossless still image compression." Electronics 10.3 (2021): 360.

3. Godse, Atul P., and Deepali A. Godse. *Computer Graphics And Multimedia*. Technical Publications, 2021.

4. Baxes, Gregory A. Digital image processing: principles and applications. John Wiley & Sons, Inc., 1994.

5. Kurak, C., and J. McHugh. "A cautionary note on image downgrading." Proceedings Eighth Annual Computer Security Application Conference. IEEE Computer Society, 1992.

6. Zheng, Dong, Yan Liu, and Jiying Zhao. "A Survey of RST Invariant Image Watermarking Algorithms." 2006 Canadian Conference on Electrical and Computer Engineering.

7. SHARP, Toby. "An implementation of key-based digital signal steganography." Lecture notes in computer science (2001): 13-26.

8. Galand, Fabien, and Gregory Kabatiansky. "Steganography via covering codes." IEEE International Symposium on Information Theory, 2003. Proceedings.. IEEE, 2003.

9. Khalil, Mafaz Mohsin, and Sadoon Hussin Abdullha. "Hidden Messages in JPEG Image Using Steganography." *Journal of Education and Science* 25.1 (2012): 136-0.

10. Wendzel, Steffen, et al. "A revised taxonomy of steganography embedding patterns." Proceedings of the 16th international conference on availability, reliability and security. 2021.

11. Westfeld, Andreas. "F5—a steganographic algorithm: High capacity despite better steganalysis." International workshop on information hiding. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001.

12. Chang, Chin-Chen ; Chen, Tung-Shou ; Chung, LouZo: A steganographic method based upon JPEG and quantization table modification. In: Inf. Sci. 141 (2002), Nr. 1-2, S. 123–138

13. Cogranne, Rémi, Vahid Sedighi, and Jessica Fridrich. "Practical strategies for content-adaptive batch steganography and pooled steganalysis." 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2017.

14. Pevný, T., & Nikolaev, I. (2015, November). Optimizing pooling function for pooled steganalysis. In *2015 IEEE International Workshop on Information Forensics and Security (WIFS)* (pp. 1-6). IEEE.

15. Cogranne, Rémi, Vahid Sedighi, and Jessica Fridrich. "Practical strategies for content-adaptive batch steganography and pooled steganalysis." *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017.

16. "Нарватов О.П., Бологова Н.М. ,ПАКЕТНА СТЕГАНОГРАФІЯ ТА ОБ'ЄДНАНИЙ СТЕГОАНАЛІЗ В JPEG // Проблеми інформатизації : XII міжнародна науково-технічна конференція. – 21-22 листопада 2024. –с.117. doi: https://doi.org/10.32620/PI.24.t2"