Національний лісотехнічний університет України
(повне найменування вищого навчального закладу)

Навчально-науковий інститут комп'ютерних наук
та інформаційних технологій
(повне найменування інституту, назва факультету (відділення))

Кафедра комп'ютерних наук
(повна назва кафедри (предметної, циклової комісії))

Магістерська кваліфікаційна робота

другий (магістерський)
(рівень вищої освіти)

на тему: **Розроблення програмного забезпечення стеганографічного захисту JPEG-файлів**

Виконав: студент VI курсу, групи КН-62м
спеціальності
122 – "Комп'ютерні науки"
(шифр і назва напряму підготовки, спеціальності)
Таланчук О.Р.
(прізвище та ініціали)
Керівник Різник О.Я.
(прізвище та ініціали)
Рецензент _____
(прізвище та ініціали)

Львів – 2024 р.

Національний лісотехнічний університет України
(повне найменування вищого навчального закладу)

ННІ комп'ютерних наук та інформаційних технологій
Кафедра комп'ютерних наук
Рівень вищої освіти другий (магістерський)
Спеціальність 122 "Комп'ютерні науки"
(шифр і назва)

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ Борецька І. Б.
"____"_____2024 року

З А В Д А Н Н Я
НА ДИПЛОМНУ РОБОТУ СТУДЕНТУ

Таланчук Олег Романович
(прізвище, ім'я, по батькові)

1. Тема роботи **Розроблення програмного забезпечення стеганографічного захисту JPEG-файлів**

керівник роботи, Різник Олег Яремович, к.т.н, доцент
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від 13.02.2023 року № С-49

2. Термін подання студентом роботи 05.01.2024 р.

3. Вихідні дані до роботи:
- провести огляд алгоритмів стеганографічного захисту;
- дослідити математичну модель стеганографічного захисту JPEG файлів;
- розробити програмний продукт з інтуїтивним інтерфейсом;
- провести тестування роботи розробленого програмного продукту.

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити)
Розділ 1. Стан проблемної області
Розділ 2. Інформаційне забезпечення
Розділ 3. Математичне забезпечення
Розділ 4. Програмне забезпечення
Розділ 5. Розроблення стартап-проєкту
Висновки

5. Перелік графічного матеріалу:
системний аналіз, розробка та відображення алгоритму роботи стеганографічного захисту JPEG файлів, структура програмного рішення, тестування
Додаток А. Вихідний код розробленого програмного забезпечення

6. Дата видачі завдання 15 лютого 2023 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломної роботи	Строк виконання етапів роботи	Примітка
1	Отримання завдання	15.02.2023	виконано
2	Огляд літературних джерел	30.04.2023	виконано
2	Розділ 1. Стан проблемної області	31.07.2023	виконано
3	Розділ 2. Інформаційне забезпечення	30.08..2023	виконано
4	Розділ 3. Математичне забезпечення	29.09.2023	виконано
5	Розділ 4. Програмне забезпечення	31.10.2023	виконано
7	Розділ 5. Розробка стартап-проєкту	30.11.2023	виконано
8	Оформлення пояснювальної записки	29.12.2023	виконано
9	Подання готової роботи	05.01.2024	виконано

Студент _____ Таланчук О.Р.
(підпис) (прізвище та ініціали)

Керівник роботи _____ Різник О.Я.
(підпис) (прізвище та ініціали)

АНОТАЦІЯ

Магістерська робота містить 94 сторінки пояснювальної записки, 25 рисунків, 3 таблиці, 1 додаток, 20 джерел.

Рисунки є одним з основних елементів великої кількості документів. Захист авторських прав власника, захист торгових марок друкованої продукції, боротьба з піратським копіюванням і несанкціонованим використанням фотографій - тому робота розробці захисту одного з найпопулярніших графічних файлів у форматі JPEG. Захист заснований на вбудовуванні цифрового водяного знака в графічні файли формату JPEG.

Ключові слова: *графічний файл, графічний формат, метод НЗБ, стегодетектування, стегозахист, цифровий водяний знак*.

ANNOTATION

The master's thesis contains 94 pages of explanatory note, 25 figures, 3 tables, 1 appendix, 20 sources.

Drawings are one of the main elements of a large number of documents. Protecting the owner's copyright, protecting the trademarks of printed products, fighting against piracy and unauthorized use of photos is the work behind the development of protection for one of the most popular graphic files in the JPEG format. Protection is based on embedding a digital watermark in JPEG image files.

Keywords: *digital watermark, graphic file, graphic format, LSB method, stegodetection, stegoprotection*.

ТЕХНІЧНЕ ЗАВДАННЯ

Необхідно розробити програмне та алгоритмічне забезпечення стеганографічного захисту JPEG-файлів, а саме:

1. провести попередній аналіз існуючих методів стеганографічного захисту JPEG-файлів;
2. визначити математичну модель стеганографічного захисту JPEG-файлів;
3. визначити основні кроки стеганографічного захисту JPEG-файлів;
4. розробити алгоритм синтезу стеганографічного захисту JPEG-файлів;
5. провести інтерпретацію отриманих результатів;
6. розробити програмне забезпечення для представлення результатів роботи.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ, ОДИНИЦЬ І ТЕРМІНІВ

АРМ	автоматизоване робоче місце
ЕЦП	електронний цифровий підпис
НЗБ	найменш значимі біти
СД	стегодетектування
СЗ	стегозахист
СЛЗ	система людського зору
УВЕ	умовна відносна ентропія
ЦВЗ	цифровий водяний знак
ЦПЗ	цифрове представлення зображень

ВСТУП

Проблема захисту інформації від несанкціонованого доступу вирішувалася протягом всієї історії людства. Дана робота присвячена захисту інформації за допомогою графічних зображень.

Цифровий водяний знак (DWM) — це спеціальний знак, який непомітно вбудований у зображення чи інший сигнал, щоб тим чи іншим чином контролювати його використання.

Важливість дослідження. Зокрема, системи цифрових водяних знаків Stego (DWM) повинні виконувати завдання захисту авторських прав і прав власності на електронні повідомлення у разі різноманітних спроб активного зловмисника спотворити або видалити інформацію про автентифікацію, що міститься в них.

Мета дослідження. Введення цифрового водяного знака у файли зображень JPEG.

Предметом дослідження є графічний файл JPEG, захищений цифровим водяним знаком.

Предметом дослідження є спосіб поступового захисту графічного файлу JPEG.

Цілі дослідження:

- вивчити поетапний захист файлів за допомогою контейнерів зображень JPEG на основі псевдовипадкових послідовностей;
- розробити ефективний алгоритм стегокодування та стегкодування для графічних файлів JPEG.
- розроблений стегозахист не повинен ідентифікуватися за допомогою відомих програм виявлення стего.

Формально системи передачі даних повинні забезпечувати аутентифікацію відправників електронних повідомлень. Подібне завдання можна покласти і на криптографічні системи електронного цифрового підпису (ЕЦП) даних, але на відміну від систем водяних знаків стего, відомі системи ЕЦП не захищають авторство не тільки цифрових, а й аналогових повідомлень і в умовах, коли активний зловмисник вносить спотворення. у повідомленні, яке є захищеною та автентифікованою інформацією.

Інші вимоги безпеки застосовуються до стегосистем, призначених для приховування передачі конфіденційних повідомлень від пасивного зловмисника. Він також має свою особливість у тому, що стегосистеми імітаційно стійкі до введення неправдивої інформації в прихований канал неправдивої інформації.

Практична цінність даної роботи полягає в захисті авторських прав графічних зображень у форматі JPEG.

РОЗДІЛ 1. СТАН ПРОБЛЕМНОЇ ОБЛАСТІ

1.1. Базова модель стегосистеми

Проблема захисту інформації від несанкціонованого доступу існує протягом усього існування людства. Для вирішення цієї проблеми ще в стародавньому світі до сьогодні існувало два основних методи - криптографія і стеганографія. Слово «стеганографія» має грецьке коріння і буквально означає «таємне письмо».

Як і у випадку з криптографічними системами захисту інформації, безпека стегосистем описується та оцінюється на основі їх стійкості (скорочено стеганографічна стійкість або стегостійкість). Під стійкістю різних стегосистем мається на увазі їх здатність приховувати факт прихованої передачі повідомлень від досвідченого порушника, здатність протистояти спробам зловмисника знищити, спотворити або видалити приховані повідомлення, а також здатність підтвердити або спростувати справжність прихованої інформації.

Розглянемо визначення стегорезистентності, опишемо класифікацію атак на стегосистеми та спробуємо визначити умови, за яких стегосистеми можуть бути стійкими. Ми вивчаємо стегосистеми, завданням яких є прихована передача інформації. Криптографічні системи приховують зміст конфіденційного повідомлення від зловмисника, а стеганографія додатково приховує існування такого повідомлення. Тому визначення стійкості та злому цих систем відрізняються. У криптографії система інформаційної безпеки є стабільною, якщо, маючи перехоплену криптограму, зловмисник не може прочитати повідомлення, що міститься в ній. Неофіційно ми визначаємо, що стегосистема є стабільною, якщо зловмисник, спостерігаючи за обміном інформацією між

відправником і одержувачем, не може виявити, що приховані повідомлення надсилаються під кришкою контейнерів, а тим більше прочитати їх. Загалом стегосистема називається нестійкою, якщо протиборча сторона здатна виявити її використання. Розглянемо базову модель стегосистеми (рис. 1.1), в якій у стегокодері використовується стеганографічна функція f, яка використовує секретний ключ K приховуваного повідомлення M в контейнер C, а в стегодекодері стеганографічна функція φ його витягання по тому ж ключу. Із стего по функції φ витягується вбудоване повідомлення $\hat{M}$ і при необхідності контейнер $\hat{C}$.

Через спотворення під час вбудовування, випадкові та навмисні зриви передачі, а також помилки під час відтворення повідомлення M, відновлене одержувачем, може відрізнятися від оригінального M.

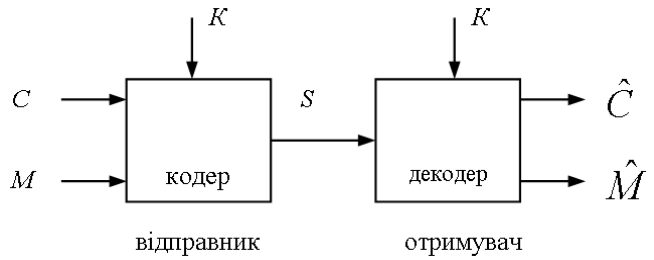

Рис. 1.1. Базова модель стегосистеми

Подібним чином отриманий контейнер C відрізнятиметься від вихідного C. Контейнер точно буде пошкоджено після вбудовування прихованого повідомлення. У багатьох стегосистемах необхідно відновлювати контейнер, оскільки він фізично представляє звичайні повідомлення (зображення, мовні сигнали тощо) кореспондентів відкритого спілкування, під виглядом яких здійснюється таємне спілкування. Ці повідомлення відкритого зв'язку мають бути доставлені одержувачам із якістю, визначеною встановленими вимогами до надійності відкритого зв'язку. Однак, навіть якщо використовуваний контейнер

є лише носієм прихованого повідомлення, міра допустимої помилки контейнера також повинна бути обмежена, інакше зловмисник легко виявить використання стегосистеми.

За використовуваним ключем цю стегосистему класифікують як симетричну. Логічно припустити, що стійкість стегосистеми повинна забезпечуватися використанням явних (загальновідомих) функцій вбудовування f і витягання φ. Безпека стегосистем повинна базуватися на таких принципах їх побудови, щоб якщо зловмиснику не відома секретна ключова інформація, то навіть при повному знанні функцій вбудовування і вилучення прихованої інформації, закони поширення прихованих повідомлень, контейнерів і stego, він не в змозі визначити прихований факт передачі інформації.

1.2. Сучасні методи приховування та виявлення інформації в зображеннях

Розгорнуту класифікацію методів стеганографії зображень наведено в [25]. Всі існуючі методи, засновані на методі вбудовування секретних повідомлень в зображення, можна розділити на наступні основні групи:

1) методи модифікації зображення в просторовій області (прямі методи);
2) методи, що модифікують трансформовані зображення;
3) методи використання фрактального кодування зображень;
4) методи, що використовують особливості формату графічного файлу;

Усі ці методи передбачають різні способи кодування зображення.

1. Прямі методи. Орієнтований на формати растрових зображень (bmp, gif). Типовими представниками цієї групи є методи модифікації молодших бітів зображення (LSB-методи). Основними перевагами цих методів є простота використання, можливість приховати відносно великі повідомлення (до 1/8 обсягу контейнера) і наявність безкоштовних інструментів в Інтернеті (STOols,

Steganos для Windows, Contraband). Стійкість прямих методів до виявлення, модифікації та знищення повідомлень низька. Для ідентифікації цих методів використовуємо:

- «Візуальна атака» передбачає використання спеціального фільтра, який створює зображення на основі кольорових цифр нижчого рівня. Оскільки існують кореляції між найменш значущими частинами зображень (особливо у випадку штучних зображень), спостерігач може легко виявити факт вбудовування.
- Атака хі-квадрат була запропонована Вестфельдом на основі статистичного аналізу першого порядку. Оригінальна версія атаки [26] виявляє послідовно вкладені повідомлення. Він був узагальнений для виявлення випадково надісланих повідомлень.
- Метод PQR [26] запропонований Фрідріхом. Обчислюються статистичні характеристики підозрілого зображення, вставляється приховане тестове повідомлення та перераховується статистика. Якщо результати близькі один до одного, швидше за все, ваші підозри виправдалися. Як статистична характеристика використовується кількість близьких пар кольорів, яка значно збільшується при вбудовуванні повідомлення.

Методи знищення повідомлень у зображеннях є тривіальними – це може бути або проста заміна псевдовипадкового значення на менш важливі біти, або використання процедури стиснення/декомпресії з втратами (наприклад, JPEG).

Прямі методи також включають методи зміни палітри. Характерною ознакою використання цих методів є наявність на зображенні нестандартно розташованих палітр або палітр з незвичайним набором кольорів. Щоб знищити повідомлення, у більшості випадків просто відкрийте та збережіть файл у будь-

якому стандартному редакторі растрових зображень, який змінить розташування палітр.

Методи, засновані на трансформації зображення. У цих методах інформація вкладається в набір коефіцієнтів, які є результатом певного перетворення вихідного зображення. При використанні формату JPEG модифікуються коефіцієнти дискретного косинусного перетворення (DCT), а при використанні формату JPEG2000 — коефіцієнти перетворення Wavelite. У [2] розглядається можливість використання інших перетворень - Фур'є, Кархунена - Лоєва, сингулярного розкладу, але вони мають швидше теоретичне значення, оскільки ці перетворення не використовуються в широко використовуваних форматах зберігання зображень.

Методи, засновані на трансформації зображення, характеризуються підвищеною стійкістю до виявлення і спотворення інформації, що міститься, порівняно з попередніми. Враховуючи поширеність формату JPEG, розглянемо його докладніше. Станом на 2003 рік існує кілька основних методів вбудовування секретних повідомлень у зображення JPEG. Це J-Steg, JPHide/JPSeek, F5 і OutGuess. Усі вони використовують маніпуляції з квантованими значеннями коефіцієнта GCP. J-Steg і JPHide/JPSeek безпосередньо розміщують повідомлення в молодших бітах коефіцієнтів ДКП і тому виявляються за допомогою простої або модифікованої атаки xi-square. Алгоритми F5 і Outguess зберігають статистику першого порядку і тому застосовують до них статистичний аналіз вищого порядку. [26].

Методи використання фрактального кодування зображень. Ідея методів цього класу полягає в тому, щоб створити зображення фрактального коду таким чином, щоб декодоване зображення вже містило вбудовану інформацію.

Сьогодні ці методи мають більшу теоретичну важливість, що пов’язано з обмеженим використанням методів фрактального стиснення зображень.

1.3. Класифікація атак на стегосистеми

Розглянемо класифікацію атак зловмисника, який намагається встановити факт прихованої передачі повідомлення, а коли цей факт встановлено, намагається його переглянути.

1.3.1. Атака лише із стегограмою

Зловмисник знає одну або кілька стегограм і намагається визначити, чи містять вони приховані повідомлення, і, якщо так, намагається їх прочитати.

У цій атаці зловмисникові дуже важко зламати стегосистему. Це пояснюється тим, що якщо невідомий ні оригінальний контейнер, ні будь-яка частина прихованого повідомлення, можна отримати дуже велику кількість неправильних розшифровок, серед яких не можна віддати пріоритет. Девід Кан у своїй відомій книзі описав, що якщо цензор, переглядаючи пошту під час Другої світової війни, не міг одразу знайти сліди прихованих повідомлень, то, швидше за все, проблема не має чіткого вирішення [1].

1.3.2. Атака з відомим контейнером

Зловмисник має доступ до однієї або кількох пар бінів і відповідних їм стегограм. Зверніть увагу, що під час цієї атаки зловмисник знає оригінальний тип контейнера, що дає йому значну перевагу порівняно з першою атакою. Наприклад, відомий зловмисник контейнера може бути студійним записом музичного твору, який транслюється по телевізійному каналу з вбудованою інформацією. Або контейнером служить зображення відомої картини, експонованої в Ермітажі, якісні цифрові копії якої вільно продаються на компакт-дисках.

1.3.3. Атака з вибраним контейнером

Зловмисник може примусово використовувати певний контейнер у стегосистемі, що має деякі переваги в стеганалізі порівняно з усім набором контейнерів. Покращена версія цієї атаки: атака Adaptive Container Selection Attack. Зловмисник надягає контейнер, аналізує отримане стего, щоб оцінити ймовірність прихованої передачі, або оцінити приховане повідомлення, або оцінити використаний ключ стего. На основі отриманих оцінок зловмисник створює інший контейнер з урахуванням іншого стегоключа, уточнює оцінки і так далі, поки не буде виявлено факт наявності або відсутності прихованого з'єднання і можливого прихованого каналу зв'язку, перш ніж обчислити використаний stegokey і читання прихованої переписки. Наприклад, така атака може статися, коли відправник прихованих повідомлень без дозволу використовує канал передачі інформації іншої особи, поки законний власник інформаційного ресурсу проводить розслідування, щоб позбутися непроханих користувачів. Зокрема, в сучасних телекомунікаційних системах відомі спроби безкоштовного використання дорогих послуг стільникового супутникового та наземного зв'язку.

1.3.4. Атака з відомим повідомленням

Зловмисник знає вміст одного або кількох прихованих повідомлень і намагається визначити, чи були вони надіслані та використовувався стегоключ. Наприклад, такий напад здійснює тюремний охоронець Віллі в класичному в’язневому завданні [6]. Віллі, знаючи тип новин про втечу, аналізує листування між ув'язненими, щоб визначити момент майбутньої втечі. Звичайно, набагато легше знайти сліди певного повідомлення в певному наборі надісланих стего, ніж виявити в тому ж наборі факт прихованого повідомлення апріорі невідомого повідомлення.

Якщо зловмисник знає деякі приховані повідомлення та відповідні стегограми, то його завдання полягає в тому, щоб визначити ключ стегосистеми, щоб виявити та прочитати інші приховані повідомлення, або якщо визначення ключа неможливо (висока складність), завдання зловмисника полягає в тому, щоб побудувати безключовий системний метод зчитування або визначення факту передачі прихованої інформації.

1.3.5. Атака з вибраним повідомленням

Зловмисник може нав'язати конкретне повідомлення для відправки через стегосистему і намагається за секретним ключем визначити факт його прихованої передачі. Також можлива атака з адаптивним вибором повідомлень, у якій зловмисник послідовно кидає вибрані повідомлення та ітеративно зменшує свою невизначеність щодо використання стегосистеми та її параметрів.

Наприклад, така атака може бути здійснена, коли є підозра, що конфіденційна інформація витікає з автоматичної робочої станції (АРМ) у локальній мережі установи, а потім таємно надсилається за межі мережі. Щоб виявити канал витоку, адміністратор безпеки генерує повідомлення, які можуть зацікавити недобросовісного користувача, і розміщує їх на мережевих дошках оголошень. Потім адміністратор намагається виявити сліди цих повідомлень в інформаційних потоках, які надсилаються від користувачів робочих станцій через сервер до зовнішніх мереж. Щоб чітко визначити наявність або відсутність секретного каналу зв'язку, адміністратор вибирає повідомлення, які легше виявити, ніж інші, при передачі по стегоканалу.

Крім того, можливі різні комбінації згаданих атак, у яких зловмисник може дізнатися або вибрати використані контейнери та передані приховані повідомлення. Ступінь ефективності атак на стегосистему зростає в міру збільшення знань зловмисника про використовувані контейнери, приховані

повідомлення, кількість захоплених стегограм і його здатність накладати вибрані контейнери та повідомлення.

Наприклад, сучасні DVD-пристрої записують інформацію про географічний регіон їх виробництва та продажу, у межах якого вирішується або обмежується відтворення DVD-дисків із відповідними тегами доступу. Згідно з цим обмеженням доступу, Росія належить до регіону, де ймовірність крадіжки електронного обладнання значно вища, ніж, наприклад, у Західній Європі.

Слід зазначити, що побудова асиметричних CVS та інших стегосистем створює значні практичні проблеми. По-перше, асиметричні системи, як відомо з криптографії, виявляються більш обчислювально складними для реалізації, ніж симетричні системи. По-друге, крім вимог до міцності ключа стегосистеми, існують суворі вимоги до стійкості системи цифрових водяних знаків до будь-яких спроб зловмисника спотворити водяний знак. Асиметричні системи будуються на основі односторонньої функції з секретним проходом, ідею якої запропонували В. Діффі та М. Хеллман [9]. Принципи розробки переважної більшості відомих функцій одностороннього бекдору такі, що будь-яке спотворення, навіть найменше, вихідного значення функції, коли законний одержувач використовує бекдор, призводить до значного збільшення помилок в отриманому повідомленні. Ця відсутність односторонніх функцій також характерна для використовуваних в даний час асиметричних криптографічних систем. Однак це можна компенсувати за допомогою додаткових заходів для підвищення надійності переданих криптограм або цифрових підписів повідомлень. Однак важко застосувати такі ж методи підвищення надійності до стегосистем. По-перше, їх використання відкриває прихований канал. По-друге, активний порушник при атаках на стегосистему центральної нервової системи має величезні можливості для вибору такої деструктивної дії, при якій наявні

методи підвищення надійності приховування інформації можуть виявитися неефективними. Наприклад, якщо для приховування інформації використовується шумостійке кодування, яке захищає приховане повідомлення від однаково ймовірних розподілених помилок, тоді зловмисник вибирає закон розподілу для помилок пакетів, де декодер каналу одержувача не може їх виправити, і множить помилки під час декодування.

1.4. Стійкість стегосистем до виявлення факту передачі приховуваних повідомлень

Для аналізу стійкості стеганографічних систем щодо виявлення факту надсилання прихованих повідомлень розглянемо інформаційно-теоретичну модель стегосистеми з пасивним порушником, запропоновану в [3].

Зловмисник, Єва, спостерігає за повідомленнями, надісланими відправником, Алісою, одержувачу, Бобу. Єва не знає, чи містять ці повідомлення нешкідливий контейнер C чи контейнер stego S, що містить приховану інформацію. Ми вважаємо, що Аліса може перебувати в одному з двох режимів: активному (і тоді стего S передається по спостережуваному каналу) або пасивному (передається порожній контейнер C). Коли Аліса активна, вона перетворює контейнер C, поміщаючи в нього приховане повідомлення M, використовуючи секретний ключ K. Давайте побудуємо стегосистему, в якій сама Аліса може створити відповідний контейнер для приховування повідомлення M. Отримавши стего S, Боб повинна мати можливість витягти з неї повідомлення M за допомогою ключа K. У термінології теорії інформації стегосистема повинна задовольняти відношенню:

1. $H(S/CMK)=0$. Формоване відправником стего S однозначно визначається значеннями контейнера C, ключа $\boldsymbol{K}$ і повідомлення M.

2. $H(M)>0$. Невизначеність в момент передачі прихованого повідомлення M як для одержувача, так і для зловмисника строго більше нуля, інакше невідома одержувачу інформація не буде відправлена через прихований канал і сенс використання стегосистеми буде втрачено. Зверніть увагу, що це може не стосуватися систем цифрового підпису, метою яких є не передача інформації, невідомої одержувачу чи порушнику, а встановлення авторства переданого контейнера.

3. $H(M/sk)=0$. Одержувач Боб повинен однозначно відновити приховане повідомлення M з отриманого стего S за допомогою ключа $\boldsymbol{K}$.

Припускаємо, що ймовірнісні розподіли набору прихованих повідомлень, контейнерів, стего та ключів відомі всім учасникам інформаційного протистояння. Крім того, отримувач Боб знає, чи активний відправник Аліса чи ні. Єва, спостерігаючи за повідомленнями, надісланими Алісою, повинна визначити, чи передається прихована інформація через канал зв'язку чи ні. Щоб визначити, чи використовується стегосистема, Єва намагається визначити, чи відповідає переданий потік інформації розповсюдженню контейнера чи розповсюдженню стего. Якщо Єві вдасться визначити, що повідомлення про розповсюдження стегограм відправляються по спостережуваному каналу, то факт прихованої передачі інформації від Аліси до Боба буде доведено, а система стегограм, яку вони використовують, нестабільна. Сформулюємо задачу доведення стійкості стегосистеми як задачу виділення гіпотез.

Відомо, що в цій моделі стегосистеми впливає імовірнісний розподіл порожніх контейнерів P_C, і ймовірнісний розподіл стегограм, Ps, що позначається. Порушник в контрольованому каналі зв'язку може спостерігати безліч можливих порожніх контейнерів і стегограм. Позначимо цю безліч можливих спостережень Q. Порушник, спостерігаючи передане повідомлення

$q \in Q$, висуває дві гіпотези H_C і H_S. Якщо справедлива гіпотеза H_C, то повідомлення q породжене відповідно до розподілу P_C, а якщо справедлива H_S, то q відповідає розподілу Ps. Правило рішення полягає в розбитті безлічі Q на дві частини так, щоб призначити одну з двох гіпотез кожному можливому повідомленню $q \in Q$. У цій задачі розрізнення можливі два типи помилок: Помилка першого типу, яка передбачає встановлення гіпотези Hs, коли вірною є H_C і помилка другого типа, коли прийнято рішення H_C при вірній гіпотезі Hs. Вірогідність помилки першого типа позначається α, вірогідність помилки другого типа - β.

Метод знаходження оптимального рішення дається теоремою Неймана-Пірсона. Правило прийняття рішення залежить від порогу T. Змінні α і β залежать від T. Теорема встановлює, що для деякого заданого порогу T і допустимій максимальній вірогідності β, вірогідність α може бути мінімізована призначенням такої гіпотези H_C для спостереження $q \in Q$, якщо і лише якщо виконуються

$$\log \frac{P_C(q)}{P_S(q)} \geq T. \tag{1.1}$$

Основним інструментом для розрізнення гіпотез є відносна ентропія (ВЕ), яка є різницею між двома розподілами ймовірностей P_C і P_S, визначувана у вигляді

$$D(P_C \parallel P_S) = \sum_{q \in Q} P_C(q) \log \frac{P_C(q)}{P_S(q)}. \tag{1.2}$$

Відносна ентропія між двома розподілами завжди невід’ємна і дорівнює 0 тоді і тільки тоді, коли вони незрозумілі (конвергентні). Хоча ВЕ не є метрикою в математичному сенсі, оскільки вона не є ані симетричною, ані трикутною,

корисно використовувати її як відстань між двома розподілами, що порівнюються. Двійкова відносна ентропія $d(\alpha,\beta)$ визначається як

$$d(\alpha,\beta)=\alpha\log\frac{\alpha}{1-\beta}+(1-\alpha)\log\frac{1-\alpha}{\beta}.$$

Використовуємо відносну ентропію $D\,(P_c \,||\, P_s)$ між розподілами *Pc* і *Ps* для оцінки стійкості стегосистеми при пасивному противнику. У [3] наведено таке визначення: стегосистема називається ε-стійкою до пасивного порушника, якщо

$$D(P_C||P_S)\le\varepsilon.$$

Якщо $\varepsilon=0$, то стегосистема є досконалою.

Якщо розподіли контейнера і стего однакові, то $D(Pc||Ps)=0$, і така стегосистема є досконалою. Це означає, що ймовірність виявлення передачі прихованої інформації не змінюється в залежності від того, чи спостерігає зловмисник обмін інформацією від Аліси до Боба чи ні. Пасивний зловмисник із довільно великими ресурсами та довільними методами стеганалізу не може виявити використання ідеальної стегосистеми.

Розглянемо умови забезпечення стійкості стегосистем. Зв'язок між ентропією, відносною ентропією та розміром алфавіту $|\mathbf{X}|$ відомий для довільних випадкових величин S і C. Зверніть увагу, що контейнери ***C*** і стего *S* належать одному і тому ж алфавіту X. Якщо змінна *S* рівноймовірна і незалежно розподілена, то

$$H(C)+D(P_C \,||\, P_S)=\log|\mathbf{X}|. \qquad (1.3)$$

Якщо змінна ***C*** є рівноймовірною і незалежно розподіленою, то, як відомо з теорії інформації [10], виконується рівність $H(C)=\log|\mathbf{X}|$ і тоді $D\,(P_C \,||\, P_S)=0$. Отже, якщо випадкові послідовності використовуються як ***C***-контейнери, а приховані повідомлення також описуються випадковими послідовностями, то створений стего *S* не матиме статистичних відмінностей від порожніх

контейнерів, і така стего-система буде ідеальною. Якщо прихована інформація представляє значущі повідомлення, описані рядками нерівних і взаємозалежних символів, то її можна легко привести до необхідної форми, зашифрувавши будь-яким надійним шифром.

Опишемо приклад формально досконалої стегосистеми, в якій контейнерами є послідовності незалежних і рівноймовірних випадкових бітів, а як функція для вбудовування прихованих повідомлень використовується добре відома криптографічна функція «одноразової заміни». Нехай контейнер $\boldsymbol{C}$ — рівноімовірна розподілена випадкова послідовність довжиною n біт. Генератор ключів генерує випадкову рівноймовірну послідовність ключів k довжиною n біт і надсилає її Алісі та Бобу. Якщо Аліса активна, тоді функція вбудовування є сумою бітів за модулем 2, щоб приховати n-бітове повідомлення m, де стего створюється відповідно до правила $s=m\oplus k$. Одержувач Боб витягує приховане повідомлення обчисленням $m=s\oplus k$. Сформоване стего S рівноймовірно розподілено для послідовності n бітів і тому $D\ (PC \parallel PS) = 0$. Таким чином, побудова функції вбудовування як однократної підстановки забезпечує досконаліть стегосистеми, якщо контейнер формується рівноймовірним випадковим джерелом.

Однак реальні повідомлення, що надсилаються через канали зв'язку, які використовуються в стегосистемах як порожні контейнери, далекі від моделі ненадлишкових і рівноімовірних джерел. Таким чином, передача повідомлень, зашифрованих описаним способом, на тлі природних джерел повідомлень негайно відкриває прихований канал зв'язку. Стеганографія характеризується випадком нерівного розподілу змінної С, яка описує виробництво природного джерела, що має деяку значну пам'ять. Повідомлення з таких джерел зазвичай служать контейнерами (зображення, мови тощо) та їх ентропія $H(S)$ звичайна

значно менше величини $\log|\mathbf{X}|$. Для вбудовування прихованих повідомлень з таких контейнерів видаляється деяка надлишковість, а приховані повідомлення вставляються в стиснуті контейнери. У результаті ймовірнісні характеристики згенерованих стегограм відрізняються від характеристик порожніх контейнерів, наближаючись до характеристик випадкового незалежного джерела. У граничному випадку дискретні стегограми описуються розподілом Бернуля. У цьому випадку всі надлишкові контейнери видаляються, а вбудоване повідомлення генерується рівноімовірним випадковим джерелом.

Розглянемо наступний приклад. В якості контейнерів використовуються україномовні повідомлення «ділової прози», для яких відома оцінка ентропії $H(C)$=0,83 біт/буква [11]. Величина $\log|\mathbf{X}|$ для української мови з алфавітом з 32 букв складає $\log 32 = 5$. Отже, в граничному випадку відносна ентропія між звичайними повідомленнями з розподілом P_S і стегограммами з розподілом *Ps* рівна

$$\varepsilon \geq D(P_C||P_S) = \log|\mathbf{X}| - H(C) = 5 - 0{,}83 = 4{,}17 \text{ [біт/буква]}.$$

Звісно, у цьому випадку незайве стего, яке виглядає як випадковий набір українських літер, одразу виділяється на фоні зайвих контейнерів, які є змістовними повідомленнями. Тому використання такої стегосистеми можна легко виявити, візуально досліджуючи повідомлення, надіслані від Аліси Бобу. Використовуючи таку стегосистему, також легко автоматизувати процес пошуку слідів прихованого каналу. Для цього просто обчисліть приблизну оцінку ентропії повідомлень, що передаються. Оскільки стегоентропія приблизно в 5 разів перевищує ентропію звичайних повідомлень, виявити наявність прихованого зв’язку досить легко. У [3] було доведено, що довільні детерміновані перетворення не збільшують VE між двома розподілами.

Лема 1: Хай $\boldsymbol{P_{Qc}}$ і $\boldsymbol{P_{Qs}}$ описують імовірнісні розподіли контейнерів і стего, відповідно, над безліччю спостережень Q. Детерміноване відображення f перетворить безліч спостережень Q в безліч спостережень T вигляду

$$f:Q\rightarrow T,\ t_c=f(q_c),\ t_s=f(q_s), \tag{1.4}$$

де q_c, $q_s\in Q$, t_c, $t_s\in T$. Тоді справедливе вираження

$$D(P_{Tc}||P_{Ts})\leq D(P_{Qc}||P_{Qs}). \tag{1.5}$$

Оскільки розрізнення між гіпотезами HC і HS є приватна форма перетворення, вірогідність помилок $\boldsymbol{\alpha}$ і $\boldsymbol{\beta}$ задовольняють нерівності

$$d(\alpha,\beta)\leq D(P_{Qc}||P_{Qs}). \tag{1.6}$$

Цей зв'язок можна застосувати в наступному вигляді: хай $\boldsymbol{\delta}$ є верхній кордон $D(P_{Qc}||P_{Qs})$ і задана верхня межа ймовірності α. Тоді вираження (1.4) дає нижній кордон вірогідності β. Наприклад, при $\alpha=0$ значення помилки $\beta\geq 2-\delta$.

Використовуючи цю лему, в роботі [3] доводиться наступна теорема.

Теорема 2: Якщо стегосистема є $\boldsymbol{\varepsilon}$-стійкою проти пасивного порушника, то вірогідність $\boldsymbol{\beta}$ невиявлення факту прихованого зв’язку та ймовірність $\boldsymbol{\alpha}$ неправильного визначення факту прихованого зв’язку задовольняють нерівності $d(\alpha,\beta)\leq\varepsilon$. У окремому випадку, якщо $\alpha=0$, то $\beta\geq 2^{-\varepsilon}$.

Нехай Аліса надішле Бобу цифрове зображення C. Використовуючи модель візуальної чутливості, вона може створити набір еквівалентних зображень, які візуально не мають відношення до вихідного C. Незалежно від того, активна Аліса чи ні, вона завантажує вибране зображення з набору C Верховна Аліса і Боб заздалегідь домовилися, яка модифікація зображення відповідає кожному прихованому повідомленню. Формально це означає, що в стегосистемі кожне з зображень C_j, де $j=1, 2, \ldots, |\mathbf{C}|$, по секретному ключу ставиться у відповідність або одне з приховуваних повідомлень M_j, де $j=1, 2, \ldots, N$, і $N<|\mathbf{C}|$, або відсутність

приховуваного повідомлення для $|\mathbf{C}|-N$ випадків. Якщо ця відповідність побудована з рівною ймовірністю та незалежно для набору контейнерів і прихованих повідомлень, то, якщо розподіл контейнерів і стего не спостерігається, зловмисник Єва, спостерігаючи за обміном інформацією між Алісою та Бобом, потенційно не зможе отримати більше інформації ніж вона мала апріорі. Оскільки Єві, за визначенням, відомі статистичні характеристики всієї множини, яка є частиною стегосистеми, вона апріорі знає, що ймовірність активного стану Аліси дорівнює $\mathbf{N}/|\mathbf{C}|$, а вірогідність відсутності передачі приховуваної інформації рівна $(|\mathbf{C}|-\mathbf{N})/|\mathbf{C}|$. Активний і пасивний стани Аліси складають повну групу подій, отже

$$\mathbf{N}/|\mathbf{C}|+(|\mathbf{C}|-\mathbf{N})/|\mathbf{C}|=\mathbf{1}. \qquad (1.7)$$

Отже, якщо Аліса має намір надіслати N прихованих повідомлень під прикриттям $|\mathbf{C}|$ контейнерів, то ймовірність того, що Єва здогадається, що будь-який контейнер містить вкладену інформацію, не може бути меншою ніж $\mathbf{N}/|\mathbf{C}|$. Якщо стегосистема ідеальна, ймовірність того, що зловмисник здогадається, що було відправлено приховане повідомлення, строго дорівнює цьому значенню.

Звідси випливає, що ймовірність пасивного стану Аліси повинна бути у багато разів більшою за ймовірність її активного стану, а використовувані контейнери з урахуванням їх модифікацій мають бути у багато разів більшими за приховані повідомлення.

Розглянемо вплив деякої додаткової інформації на розташування контейнерів і стего, а отже, на стабільність стегосистеми. Нехай на розподіл контейнерів впливають якісь зовнішні події, наприклад прес-релізи чи погода у відомому «завданні арештанта». Ця додаткова інформація позначена літерою Y і відома всім учасникам. Визначимо середню ймовірність форми $\overline{\alpha}=\sum_{y\in Y}P_Y(y)\alpha(y)$ для помилки 1 роду і $\overline{\beta}=\sum_{y\in Y}P_Y(y)\beta(y)$ для помилки 2

роди, де $\alpha(y)$ і $\beta(y)$ означають, відповідно, величину вірогідності помилок 1 і 2 роди для $Y=y$.

Умовна відносна ентропія (УВЕ) між P_C і P_S, що належать одному алфавіту X, залежна від змінної Y, визначається у вигляді

$$D(P_{C/Y} \parallel P_{S/Y}) = \sum_{y \in Y} P_Y(q) \sum_{c,s \in X} P_{C/Y=y}(q) \log \frac{P_{C/Y=y}(q)}{P_{S/Y=y}(q)}. \quad (1.8)$$

З нерівності Ієнсена [10] і з вираження (2.4) виходить, що

$$d(\overline{\alpha}, \overline{\beta}) \le D(P_{C/Y} \parallel P_{S/Y}). \quad (1.9)$$

Стегосистема з додатковою інформацією Y, контейнерами C і стего S називається ε-стійкою проти пасивного противника, якщо умовна відносна ентропія $D(P_{C/Y} \| P_{S/Y}) \le \varepsilon$. Як приклад використання зовнішньої інформації в стегосистемі наведемо «класичну» задачу Г. Сіммонса, в якій ув'язнені таємно обмінюються інформацією про втечу. Надійність передачі повідомлення про евакуацію в темну ніч вище, ніж у світлу ніч. Цей факт добре відомий не лише втікачам, а й їхнім охоронцям, які посилюють контроль над можливими каналами передачі секретної інформації. Тому використання загальновідомої додаткової інформації в стегосистемі полегшує завдання зловмисника. Можна сказати, що ε-стійка стегосистема з додатковою інформацією Y забезпечує більшу секретність зв'язку порівняно з аналогічною ε-стійкою стегосистемою без цієї інформації.

Висновки до розділу 1

Розглянуто методи, що засновані на особливостях форматів графічних файлів. Для вкладення повідомлень використовуються зарезервовані поля у форматах, можливість додавати коментарі до файлів та інші подібні підходи. Повідомлення можна легко виявити та знищити, тому що просто потрібно проаналізувати та скинути ці поля даних.

РОЗДІЛ 2. ІНФОРМАЦІЙНЕ ЗАБЕЗПЕЧЕННЯ

2.1. Дерево цілей та дерево проблем стегозахисту

Розглянемо дерево цілей реалізації захисту файлів на рівні авторського права (рис. 2.1), де необхідні елементи:

1. стегокодування, що складається з:
   – ключ;
   – зображення-контейнер;
   – вбудована інформація;
   – алгоритм вбудовування інформації.
2. стегодекодування, що складається з:
   – ключ;
   – зображення-контейнер з вбудованою інформацією;
   – алгоритм вилучення інформації.

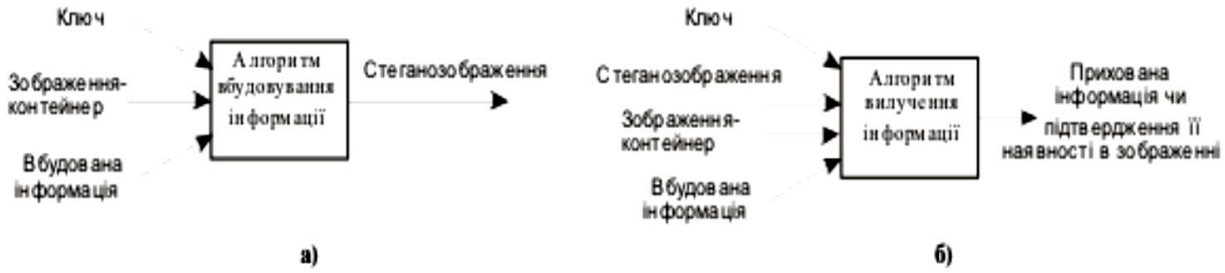

Рис. 2.1. Дерево цілей для здійснення авторського стегозахисту файлів

Розглянемо дерево задач захисту авторських файлів за допомогою цифрової стеганографії (рис. 2.2), де необхідні елементи:

Рис. 2.2. Дерево проблем вибору авторського стегозахисту файлів

Давайте розглянемо подальші методи приховування даних у нерухомих графічних зображеннях.

2.2. Стійкість недетермінованих стегосистем

На підставі аналізу компонування контейнерів і розподілу стего було виявлено використання стегосистеми. Для цього розглянута інформаційно-теоретична модель передбачає, що зловмисник точно знає ймовірнісні характеристики контейнерів, стего, прихованих повідомлень і ключів. Модель також передбачає, що передані стегограми і порожні контейнери не спотворюються при доставці по каналу зв'язку, а відправник прихованих повідомлень вибирає тільки ті контейнери, характеристики яких збігаються з характеристиками всього набору контейнерів. У зв’язку з цим будь-яке відхилення статистики повідомлень, що спостерігається зловмисником у каналі зв’язку, від середньостатистичних характеристик порожніх контейнерів слід розглядати як факт виявлення стегоканалу. Очевидно, що така ідеальна модель не є цілком адекватною реаліям систем приховування інформації. По-перше, зловмиснику відомі не характеристики контейнера, який фактично використовує відправник, а середні характеристики набору повідомлень з деяких джерел, які

можна використовувати як контейнер. По-друге, усі відомі джерела можливих бінів за своєю природою є нестаціонарними, тобто їх точних оцінок немає. По-третє, приховувач інформації може вибирати контейнери з усього набору, характеристики яких відрізняються від відомих елементів, порушуючи характеристики набору, для вбудовування прихованої інформації. Більш того, відправник може вибрати такі контейнери або спеціально згенерувати їх, щоб після вбудовування в них прихованих повідомлень характеристики згенерованого стего не відрізнялися від середніх характеристик порожніх контейнерів. По-четверте, у сучасних телекомунікаційних системах надлишкові повідомлення, що передаються, зазвичай стискаються, вносячи деякі спотворення, прийнятні для їх одержувачів, що змінює їхні характеристики. Наприклад, мовний сигнал кодується за допомогою методів лінійного передбачення мови, зображення стискаються за допомогою алгоритмів JPEG, MPEG або H.263. По-п'яте, канал зв'язку може порушити потік інформації, що передається. І якщо канал ідеальний, то з метою маскування відправник може сам зашумити передані стего і порожні контейнери такими перешкодами, які в допустимих межах спотворюють передані повідомлення і змінюють статистику стего і контейнерів до такої міри. достатньо для приховування.

Ці причини призводять до моделі стегосистеми, в якій зловмисник може визначити, що статистика послідовності, яку він спостерігає в каналі, відрізняється від відомої статистики контейнера, але не може визначити причину цих відмінностей. Таким чином, хоча зловмисник підозрює існування прихованого каналу, він чи вона не може довести чи спростувати це. Необхідні докази можна отримати, якщо зловмиснику вдасться прочитати приховане повідомлення. Використовуючи методи теорії інформації, опишемо стійкість стегосистеми для читання прихованих повідомлень.

У [2] стабільність стегосистеми визначається з кількох інших позицій, ніж у качиному підході [3]. Стегосистема називається стійкою до інформаційної теорії, якщо зловмисник не може отримати будь-яку інформацію про вбудоване повідомлення, аналізуючи стегозахоплення, за умови, що він знає статистичні характеристики порожніх контейнерів. Як відомо з теорії інформації [10], взаємна інформація може бути визначена безумовною та умовною ентропією:

$$I(M;(S,C)) = H(M) - H(M/(S,C)) = 0. \quad (2.1)$$

Це основна умова стабільності стегосистеми виду

$$H(M/(S,C)) = H(M). \quad (2.2)$$

Це визначення інформаційно-теоретичної безпеки стегосистеми дуже схоже на відповідне визначення інформаційно-теоретичної безпеки системи шифрування інформації. Якщо невпевненість зловмисника щодо повідомлення M не зменшується після перехоплення криптограми E, то ця система шифрування ідеальна за визначенням Шеннона. [7]:

$$H(M/E) = H(M). \quad (2.3)$$

Зверніть увагу, що вирази (2.2) і (2.3) вказують на те, що зловмисник не може визначити жодного біта захищеного повідомлення. При цьому система шифрування точно знає, що криптограма містить це повідомлення. У випадку стегосистеми вираз (2.2) може бути задоволений у таких випадках:

1. Стегосистема не використовується.
2. Відбувається прихована передача інформації, для встановлення наявності прихованого зв'язку використовується досконала стегосистема. Якщо зловмисник не може визначити наявність прихованого повідомлення, тим більше він не зможе прочитати будь-яку частину цього повідомлення.

3. Відбувається прихована передача інформації, злочинець може встановити наявність прихованого зв'язку. Однак він не може прочитати жодного фрагмента прихованого повідомлення.

Наприклад, третій випадок був описаний у попередньому параграфі під час вбудовування ненадлишкових прихованих повідомлень у однаково ймовірні випадкові послідовності контейнерів за допомогою функції вбудовування одноразової заміни. Отримані стего легко виявляються злочинцями на тлі звичайних, непотрібних повідомлень. Однак ці повідомлення практично неможливо прочитати, якщо під час вбудовування використовується випадкова рівноімовірна послідовність клавіш.

Вираз (2.2) означає, що невизначеність зловмисника щодо повідомлення *M* не повинна зменшуватися, якщо він знає стего S і контейнер *C*, тобто: *M* має бути незалежним від *S* і *C*. Проаналізуємо умови стабільності стегосистем. Ми віримо, що це не просто алфавіти *S* і *З*, але і їх ентропії *H(S)* і *H(C)* рівні. Розглянемо два випадки.

1. Хай жодне повідомлення *M* не вбудовується в контейнер *C.* Вочевидь, що в цьому випадку, коль *S* і *Із* збігаються, то виконується $H(S/C)=H(C/S)=0$.

2. У стего *S* є повідомлення *M* з ентропією $H(M)>0$. Вочевидь, що за наявності цієї вбудованої інформації у порушника з'являється відмінна від нуля невизначеність відносно *S*, якщо відомо *C* і невизначеність відносно *C*, якщо відоме *S*: $H(S/C))>0$, $H(C/S)>0$. У результаті взаємна інформація між прихованими повідомленнями та їхніми відповідними контейнерами та стего більше не може дорівнювати нулю:

$$I(M;(S,C)) = H(M) - H(M/(S,C)) > 0. \quad (2.4)$$

Тому

$$H(M/(S,C)) < H(M). \quad (2.5)$$

Це означає, що умова стійкості стегосистеми не виконується. Можна показати, що стабільність є необхідною і достатньою умовою:

$$H(S/C)=H(C/S)=0. \quad (2.6)$$

Таким чином, [2] стверджує, що якщо зловмиснику відомі стегограми та відповідні їм біни, то стегосистема не може бути ідеальною. Відповідно до моделі теорії інформації, ця стегосистема при атаці зловмисника з відомим контейнером не в змозі приховати факт передачі прихованого повідомлення. А з виразу (2.6) випливає, що зловмисник також може дізнатися якщо не весь, то частковий зміст цього повідомлення: якщо $I(M;(S,C))>0$, то при відомих S і C невизначеність порушника про це повідомлення менше його ентропії.

Забезпечення необхідної стійкості може бути досягнуто шляхом переходу від детермінованих стегосистем до недетермінованих (імовірнісних) систем. Розглянемо один із можливих варіантів побудови імовірнісної стегосистеми, запропонований у [2]. У розглянутій імовірнісній стегосистемі має виконуватися необхідна і достатня умова формостійкості $H(S/C)=H(C/S)=0$ гарантує, що використовуваний контейнер не буде відомий зловмиснику. Для цього в модель стегосистеми вводиться джерело контейнерів C_S, характеристики якого відомі порушникові. Щоб вставити приховані повідомлення з набору C_S випадково і рівноймовірно виберемо підмножину контейнерів C, яке назвемо підмножиною дійсних контейнерів: $C \subseteq C_S$. Нехай умова виконується $H(CS)$ і $H(C)$ і ймовірнісні характеристики підмножини C відрізняються від відповідних характеристик безлічі C_S. Зажадаємо, аби невизначеність порушника відносно дійсних контейнерів при відомій безлічі C_S була б строго більше нуля: $H(C/Cs)>0$.

Фізично це можна забезпечити, якщо вибір дійсних бінів виконується з використанням випадкового та рівноімовірного значення R, отриманого з виходу генератора випадкових чисел, як показано на рис. 2.3. Необхідна невизначеність

щодо C досягається повністю випадковим вибором кожного бункеру та збереженням вибору в секреті. Прикладом такого процесу може бути вибірка аналогового вхідного сигналу, такого як мова чи відео. Помилка квантувача забезпечує необхідну невизначеність. Якщо зміни в контейнері під час вбудовування інформації знаходяться в межах похибок квантователя, такі маніпуляції не можуть бути виявлені.

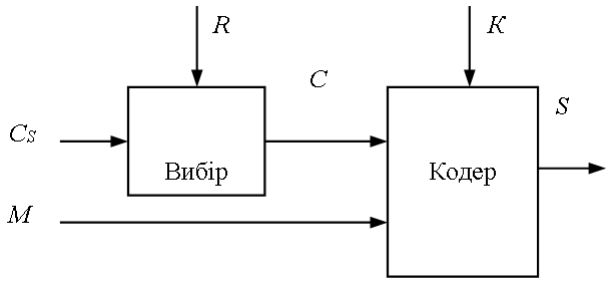

Рис. 2.3. Стегосистема з рандомізованим вибором контейнеру

Встановимо, що для ймовірнісної стегосистеми основна умова стійкості виражається у формі

$$H(M/(S,C_S)) = H(M). \tag{2.7}$$

Це означає, що невизначеність порушника відносно M не може бути зменшена знанням S і C_S, або M є незалежним від S і C_S.

Ми досліджуємо умови, за яких зловмисник не може виявити зміни в повідомленнях контейнера M, які відбулися під час вбудовування з ентропією $H(M)$, спостерігаючи за стего. Для цього ми визначаємо необхідний рівень невизначеності зловмисника по відношенню до контейнера H (C/S). Це можна продемонструвати

$$H(C/S) \geq I(M;(S,C)) = H(M) - H(M/(S,C)). \tag{2.8}$$

У гіршому випадку супротивник здатний повністю визначитися M з S і C: $H(M/(S,C)) = 0$.

Так що в загальному випадку це так

$$H(C/S) \geq H(M). \quad (2.9)$$

Оскільки взаємна інформація не може бути більшою, ніж $H(M)$, невизначеність повинна бути принаймні такою ж величиною, щоб повідомлення було нечитабельним.

У стабільній стегосистемі зловмисник, який спостерігає за стего S, не повинен отримувати інформацію, що перевищує ту, яку він знає апріорі зі знання набору C_S:

$$H(C/C_S)=H(C/S), \quad (2.10)$$

і, тому

$$H(C/C_S)\geq H(M). \quad (2.11)$$

Те саме для зловмисника, який знає характеристики набору C_S, у стабільній стегосистемі невизначеність щодо підмножини дійсних контейнерів C має бути не менше, ніж ентропія прихованих повідомлень.

Визначимо спільну ентропію $H0$ між безліччю C і CS

$$H0=H(C,CS)=H(C)+H(CS/C). \quad (2.12)$$

Оскільки $C \subseteq C_S$ і $H(C_S)\geq H(C)$, то $H(C_S/C)\geq H(C/C_S)$.

Для стабільної стегосистеми ми отримуємо нижню межу значення повної ентропії

$$H_0\geq H(C)+H(C/C_S). \quad (2.13)$$

Використовуючи вираження (3.10), запишемо

$$H_0\geq H(C)+H(M). \quad (2.14)$$

Оскільки $H(C_S)\geq H(C)$, то $H(C_S, S)\geq H(C, S)$. Отже

$$H(C_S, S)\geq H(C, S). \quad (2.15)$$

Відповідно до виразу (2.15) знаходимо, що границю можна визначити у вигляді:

$$H(C_S, S) \geq H(M). \qquad (2.16)$$

Зробимо висновок: зловмисник, який досяг нижньої межі $H(C/S)$ (рівняння 3.8), знаючи, S і C_S, не здатний дістати доступ до приховуваного в стего S повідомлення M. Основна умова стійкості (2.16) може бути виконана.

Розглянемо умови, за яких зловмисник не може визначити ключ до стегосистеми. Бажаємо порушнику, хто знає S і C_S, не міг отримати жодної інформації ні про ключ К, ні про повідомлення M. Це може бути виражено у вигляді

$$I((K, M);(S, C_S))=H(K, M)–H((K, M)/(S, C_S))= \qquad (2.17)$$
$$H(K, M)–H(K/(S, C_S))–H(M/(S, C_S, K))=0.$$

При знанні ключа K, множини C_S із стего S однозначно витягується повідомлення M:

$$H(M/(S, CS, K))=0, \qquad (2.18)$$

Тому з вираження (2.15) отримаємо

$$H(K/(S, CS)=H(K, M) \qquad (2.19)$$

або

$$H(K/(S, CS)=H(\text{M})+H(K/M) \text{ і } H(\text{M}) \qquad (2.20)$$

відповідно, оскільки $H(K/M)\geq 0$.

Тому для зловмисника невизначеність стабільного ключа стегосистеми не може бути меншою, ніж невизначеність переданого прихованого повідомлення. Ця вимога до досконалих стегосистем дуже схожа на вимогу до невизначеності ключа K для досконалих систем шифрування, для яких ентропія ключа K із захопленою криптограмою E повинна бути не меншою за ентропію зашифрованого повідомлення M [7]:

$$H(K / E) \geq H(M). \qquad (2.21)$$

Ми робимо висновок, що фактичний контейнер повинен бути невідомий зловмиснику, щоб забезпечити інформаційно-теоретичну стабільність стегосистеми. Зловмисник не може виявити передачу прихованого повідомлення або прочитати його, якщо виконуються дві умови:

1). Знання S і C_S не зменшує для порушника невизначеності про приховуване повідомлення

$$H(M/(S, CS))=H(M/s)=H(\mathrm{M}). \quad (2.22)$$

2). Умовна ентропія ключа не може бути меншою за ентропію прихованого повідомлення:

$$H(K/(S, C_S))\geq H(M). \quad (2.23)$$

За таких умов може бути забезпечена необхідна стійкість імовірнісних стегосистем.

Робота [2] містить загальний опис можливих імовірнісних стегосистем. Дозвольте відправнику використовувати цифрове альбомне зображення на виході електронної камери, щоб вставляти приховані повідомлення як важливі контейнери. Зловмисник може знати загальний вигляд зробленого зображення та характеристики використовуваної камери. Однак зловмисник і навіть законний одержувач не знають точного положення камери та кута зйомки. Якщо нахилити камеру навіть на частку градуса, то зображення значно відрізняються. Тому, коли зловмисник аналізує захоплене стего, він або вона не може визначити, яке цифрове зображення є справжнім контейнером, і тому не може відрізнити стего від контейнера. Як і багато контейнерів C_S У цьому прикладі використовуються різні варіанти представлення пейзажу з різних ракурсів, враховуючи недосконаліть оптико-електронного перетворювача використовуваної камери.

Другим прикладом ймовірнісної стегосистеми є використання аналогових вибірок випадкових сигналів, таких як мова, як реальних контейнерів значень

вибірки. У різних технічних пристроях для перетворення аналогових сигналів в цифрову форму використовуються аналого-цифрові перетворювачі з певною похибкою квантування дискретизації, а моменти дискретизації визначаються тактовим генератором, положення стробуючих імпульсів також має певну похибку. . У результаті для зловмисника, який точно знає характеристики аналогового сигналу, існує невизначеність між аналоговим і цифровим представленнями сигналу. Використовуючи такий сигнал як контейнер, потенційно можливо побудувати стабільну стегосистему, якщо ентропія вбудованого повідомлення не перевищує певної невизначеності [12].

2.3. Практичні оцінки стійкості стегосистем

2.3.1. Постановка завдання практичної оцінки стегостійкості

Теоретичні оцінки стабільності стегосистем, які обговорювалися раніше, наприклад, теоретиками інформації, припускають, що приховувач інформації та зловмисник мають необмежені обчислювальні ресурси для приховування перетворень та стеганалізу, мають нескінченний час для передачі та виявлення прихованих повідомлень тощо. моделі невідповідні.

В останні роки з'явилися програмні стегосистеми, які дозволяють приховувати інформацію в цифрових відео- та аудіофайлах. Такі програми вільно поширюються, легко встановлюються на персональні комп’ютери, підключаються до сучасних інформаційних технологій і не вимагають спеціальної підготовки при їх використанні. Вони дозволяють вставляти текст у зображення, зображення в зображення, текст у аудіосигнал тощо. У сучасних телекомунікаційних мережах, таких як Інтернет, надсилаються дуже великі потоки мультимедійних повідомлень, які можна використовувати для приховування інформації. Однією з найбільш актуальних і складних проблем цифрової стеганографії є виявлення факту такого приховування. У реальних

умовах найбільш типовим типом атаки для зловмисника є атака лише на стего, оскільки фактичний контейнер зазвичай йому невідомий. У цих умовах виявлення прихованого повідомлення можливе на основі виявлення порушень залежностей, властивих природним контейнерам [14, 16, 17]. Практичний стегоаналіз цифрових стегосистем - дуже молода наука, але вже має в своєму арсеналі ряд методів, які дозволяють з високою ймовірністю ідентифікувати наявність стегоканалу, створеного деякими запропонованими на даний момент стегосистемами. Серед методів практичного стеганалізу розглянемо візуальну атаку та ряд статистичних атак. Ці атаки спочатку були запропоновані для виявлення введення прихованої інформації в молодші біти елементів контейнера, які зазвичай називаються найменш значущими бітами (НЗБ).

2.3.2. Візуальна атака на стегосистеми

Розглянемо принцип побудови візуальної атаки, яка дозволяє виявити наявність прихованого повідомлення, вбудованого в зображення контейнера [14]. Нехай стегосистема побудована таким чином, що НЗБ елементів зображення замінюються прихованими бітами повідомлень. Наприклад, у системі EzStego молодший біт колірної складової кожного пікселя, починаючи з початку зображення, послідовно замінюється на відповідний біт прихованого повідомлення. В інших стегосистемах біти вхідного повідомлення замінюють молодші біти компонента яскравості кожного пікселя зображення. Раніше вважалося, що НЗБ компонентів яскравості або кольору пікселів зображення та бітів молодшого порядку зразків мови або аудіо не залежать один від одного, а також не залежать від останніх бітів елементів даних контейнера. Однак насправді це не так. Молодші біти не є чисто випадковими. Між найменш значущими фрагментами суміжних елементів природних водойм існують

кореляційні зв'язки. Також були виявлені зв'язки між НЗБ і останніми фрагментами природних елементів контейнера.

На рис. 2.4 показано зображення млина, зліва на рисунку показано порожній контейнер, з правого боку приховане повідомлення вбудовано послідовно, крок за кроком, у кожен колірний компонент пікселів НЗБ. Різниця між контейнером і стегою візуально непомітна. Але якщо зображення повністю складається з 3D стегопікселів, ви можете легко побачити сліди прикріплення. На рис. 2.5 зліва показано зображення, що складається з порожнього контейнера. Ви бачите, що характер зображення суттєво не змінився. Праворуч зображено найменш значущі частини контейнера, наполовину заповненого прихованим повідомленням. Ви бачите, що верхня частина зображення, де введено повідомлення, є випадковим сигналом. У цій стегосистемі приховане повідомлення, яке потрібно вставити, зашифровано таким чином, що кожен його біт є майже однаково вірогідним і незалежним від сусідніх бітів, що полегшує візуальне виявлення факту його вбудовування шляхом порівняння зображень молодших бітів стего та відповідно порожні природні баки. У деяких стегосистемах повідомлення стискаються перед вбудовуванням. Це доцільно як для зменшення розміру введеної прихованої інформації, так і для того, щоб її було складніше прочитати неавторизованим особам. Історики даних перетворюють стиснене повідомлення в послідовність бітів, яка є досить близькою до випадкової. Чим вищий рівень стиснення, тим більш випадковою є послідовність на виході архіватора, і тим легше визначити наявність стегоканалу під час візуальної атаки. Однак, навіть якщо приховане повідомлення не зашифроване або стиснуте перед вбудовуванням, його імовірнісні характеристики не збігаються з імовірнісними характеристиками повторно виявлених контейнерів, що використовуються. Зауважте, що відправник

повідомлення може вибрати контейнер із правами розповсюдження, які відповідають правам розповсюдження конкретного вбудованого повідомлення. У цьому випадку візуальна атака, як і статистичні атаки, неефективна. Однак труднощі з підбором необхідної ємності можуть зробити таку стегосистему непрактичною.

У відомій програмі Steganos [13] повідомлення довільної довжини вбудовуються в усі НЗБ пікселі контейнера, тому це виявляється візуальною атакою.

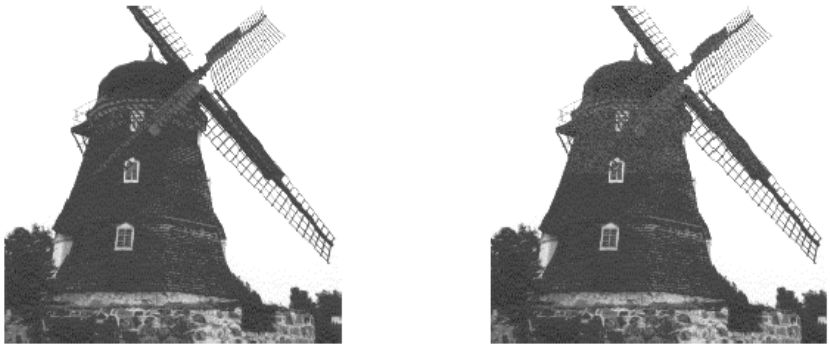

Рис. 2.4. Фото млина, ліворуч порожня тара, праворуч - з прикріпленим повідомленням

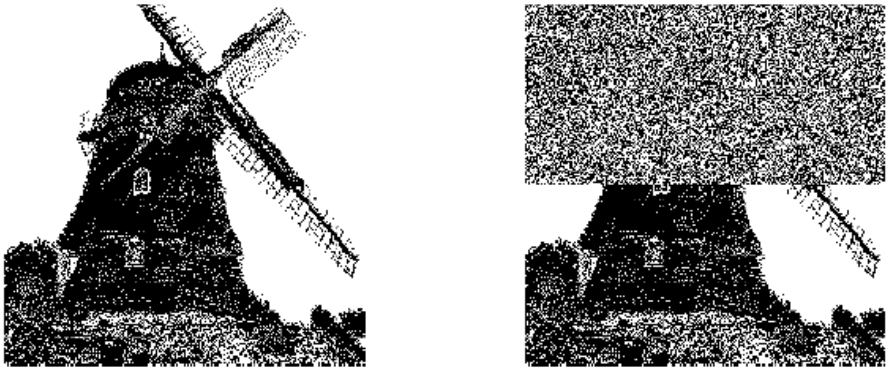

Рис. 2.5. Зображення млина, ліворуч – порожній контейнер, праворуч – порожній контейнер із вбудованим повідомленням

Візуальна атака ефективна, коли контейнер повністю заповнений, але в міру того, як ступінь наповнення зменшується, людському оку стає все важче побачити мітки кріплення між елементами контейнера, що зберігаються.

У багатьох системах стеганографії приховані елементи повідомлення вбудовані в молодші біти коефіцієнтів перетворення Фур'є контейнера. Наприклад, 8×8 пікселів $f(x,y)$ блоки зображення спочатку перетворюються на 64 коефіцієнти дискретного косинусного перетворення (ДКП) відповідно до правила

$$F(u,v)=\frac{1}{4}C(u)C(v)\left[\sum_{x=0}^{7}\sum_{y=0}^{7}f(x,y)\cdot\cos\frac{(2x+1)u\pi}{16}\cos\frac{(2y+1)v\pi}{16}\right], \qquad (2.24)$$

де $C(u)$ і $C(v)=\frac{1}{\sqrt{2}}$ коли u і v дорівнюють нулю і $C(u),\ C(v)=1$ в інших випадках.

Отримані коефіцієнти квантуються з округленням до цілого:

$$F^{Q}(u,v)=Integer\_Round\left(\frac{F(u,v)}{Q(u,v)}\right), \qquad (2.25)$$

де $Q(u,v)$ є таблиця квантування з 64 елементів.

Найменші значущі біти квантування ДКП коефіцієнтів, за виключенням $F^{Q}(u,v)=0$ і $F^{Q}(u,v)=1$, в стегосистемі є надлишковими бітами і замість них упроваджуються біти приховуваного повідомлення.

Візуальна атака малокорисна для таких методів приховування, оскільки зміна будь-якого заданого коефіцієнта перетворення призводить до змін у багатьох пікселях зображення. Наприклад, у програмі Jsteg перетворення виконується на матриці 16×16 пікселів контейнера. У результаті вбудовування прихованого повідомлення в молодші біти коефіцієнтів перетворення призведе до відносно невеликих змін у кожному з 256 пікселів, що візуально непомітно.

Тому ми розглядаємо другий клас практичних стегоатак для виявлення прихованого каналу передачі інформації на основі аналізу відмінностей між статистичними характеристиками природних контейнерів і стего, створеного з них.

2.3.3. Статистичні атаки на стегосистеми із зображеннями-контейнерами

Для виявлення слідів каналу прихованої передачі інформації можна оцінити величину ентропію елементів контейнерів. Стего, що містять вкладення прихованих даних, мають більшу ентропію, чим порожні природні контейнери. Для оцінки ентропії доцільно використовувати універсальний статистичний тест Маурера [18].

Розглянемо атаку на основі аналізу статистики хі-квадрат. У програмі EzStego молодший біт колірної компоненти кожного пікселя контейнера-зображення замінюється бітом приховуваного повідомлення. Досліджуємо закономірності у вірогідності появи значень колірної компоненти в природних контейнерах і сформованих програмою EzStego стего. При заміні молодшого біта колірної компоненти чергового пікселя контейнера на черговий біт заздалегідь зашифрованого або стислого повідомлення номер кольору пікселя стего або дорівнює номеру кольору пікселя контейнера, або змінюється на одиницю. У роботі [14] для пошуку слідів вкладення запропонований метод аналізу закономірностей у вірогідності появи сусідніх номерів кольору пікселів. Номер кольору, двійкове представлення якого закінчується нульовим бітом, назвемо лівим (L), а сусідній з ним номер кольору, двійкове представлення якого закінчується одиничним бітом - правим (R). Хай колірна гамма вихідного контейнера включає 8 кольорів. На рис. 2.6 зліва показана одна з типових гістограм вірогідності появи лівих і правих номерів кольору в природних

контейнерах. Справа показана гістограма вірогідності появи лівих і правих номерів кольору в стего, сформованого з цього контейнера програмою EzStego. Видно, що вірогідність появи лівих і правих номерів кольору в природних контейнерах істотно розрізняється між собою у всіх парах, а в стего ця вірогідність вирівнялася. Це є явною демаскуючою ознакою наявності приховуваної інформації. Відмітимо, що середнє значення вірогідності для кожної пари в стего не змінилося в порівнянні з контейнером (показано на рис. 2.6 пунктирною лінією).

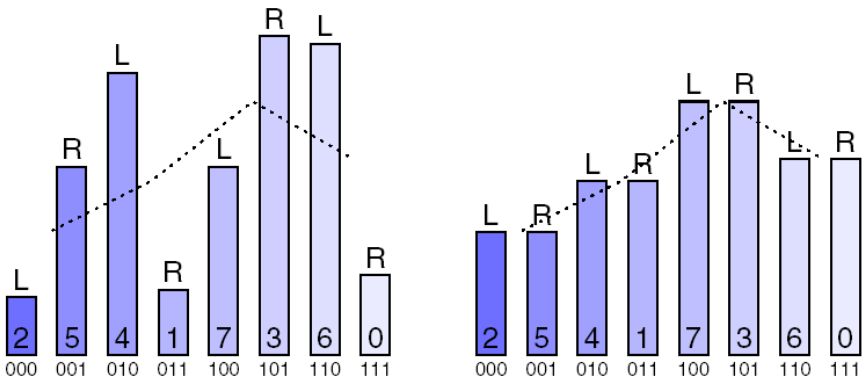

Рис. 2.6. Гістограма частоти появи номерів кольорів зліва і справа, зліва - до вбудовування, справа – після

Коли біти вхідного повідомлення замінюють біти яскравості піксельних компонентів нижчого порядку контейнера зображення, виявляються аналогічні статистичні відмінності.

Ступінь різниці між ймовірнісними розподілами елементів природного резервуару та отриманим з них стего може бути використаний для оцінки ймовірності існування стегоканалу. Цю ймовірність можна зручно визначити за допомогою тесту відповідності хі-квадрат [19]. Тест хі-квадрат порівнює, наскільки близький розподіл досліджуваної послідовності до характеристик розподілу стегограми. У послідовності, що перевіряється, має значення те,

скільки разів n_i її елемент x_i набув даних значень, де всього k елементів. Наприклад, на гістограмі лівих і правих кольорових чисел у лівій частині рис. Колір 3.4 номер 000 з'явився 2 рази ($n_0^* = 2$), а номер 001 - 5 разів ($n_1^* = 5$). При вбудовуванні чергових бітів приховуваного повідомлення в НЗБ цієї пари номер кольору 000 повинен з'являтися в середньому n_0 раз

$$n_0 = \frac{n_0^* + n_1^*}{2}. \tag{2.26}$$

Вірогідність появи цих елементів в стего по правилу: $p_i = n_i / n$. Відповідно, для досліджуваної послідовності вірогідність рівна: $p_i^* = n_i^* / n$.

Величина Хі-квадрат стего рівна

$$\chi^2 = \sum_{i=1}^{\nu} \frac{(n_i - np_i)^2}{np_i}, \tag{2.27}$$

де ν є число мір свободи. Число мір свободи дорівнює числу k мінус число незалежних умов, накладених на вірогідності p_i^*. Накладемо одну умову вигляду

$$\sum_{i=1}^{k} p_i^* = 1. \tag{2.28}$$

Вірогідність *p* того, що два розподіли однакові, визначається як

$$p = 1 - \int_0^{\chi^2} \frac{t^{(\nu-2)/2} e^{-t/2}}{2^{\nu/2} \Gamma(\nu/2)} dt, \tag{2.29}$$

де Γ є гамма-функція Ейлера.

Чим більше значення *p,* тим більша ймовірність вбудовування прихованої інформації в досліджувану послідовність.

Розгляньте можливість використання тесту хі-квадрат, щоб знайти сліди стегоканалу, створеного за допомогою програми EzStego. Нехай зображення контейнера «Młyn», показане зліва на рис. 2.3, 3600 байт прихованого

повідомлення послідовно вбудовуються в НЗБ спектральних коефіцієнтів зображення, починаючи від його верхнього краю до середини. На рис. 2.7 показано ймовірність вбудовування прихованої інформації в залежності від розміру досліджуваної послідовності. Початок графіка отримано шляхом аналізу першого фрагмента стего, який становить одну соту всього стего. P-значення становило 0,8826. Потім до аналізованого фрагменту додавали ще одну соту стего і так далі. На другому кроці ймовірність була 0,9808, а потім під час стегоаналізу вона не опускалася нижче 0,77. Коли ми перейшли до аналізу нижньої частини зображення, яка не містить прихованої інформації, *p*-значення швидко впало до нуля.

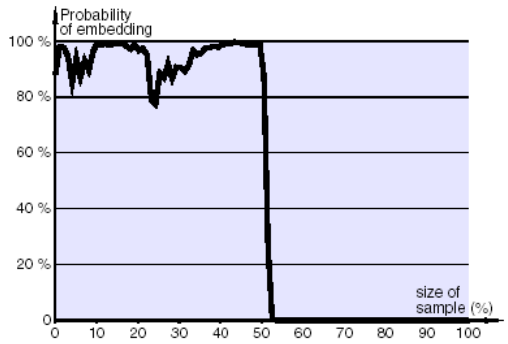

Рис. 2.7. Вірогідність вбудовування по критерію хі-квадрат при аналізі EzStego

У Steganos вбудоване двійкове повідомлення будь-якої довжини підлаштовується під довжину контейнера (кількість пікселів зображення). Таким чином, тест хі-квадрат під час вбудовування довільно малого повідомлення за допомогою Steganos дає ймовірність існування стегоканалу, який практично не відрізняється від одного.

У S-Tools вбудоване повідомлення рівномірно розподілено по всьому контейнеру. Коли бункер повністю заповнений, тест хі-квадрат впевнено виявляє сліди включення сторонньої інформації з низькою ймовірністю помилки (менш

10^{-16}), чим можна знехтувати, але при заповненому контейнері на третину і менш сліди стегоканалу не виявляються.

Як і EzStego, Jsteg будує приховане повідомлення послідовно в коефіцієнтах трансформації контейнера. На рис. 2.8 показано ймовірність вбудовування хі-квадрат під час аналізу стего, згенерованого за допомогою Jsteg. Видно, що статистична атака успішно виявляє сліди прихованої інформації в першій частині досліджуваної послідовності, що містить приховане повідомлення, і не викликає помилкової тривоги в її другій частині, тобто порожньому контейнері.

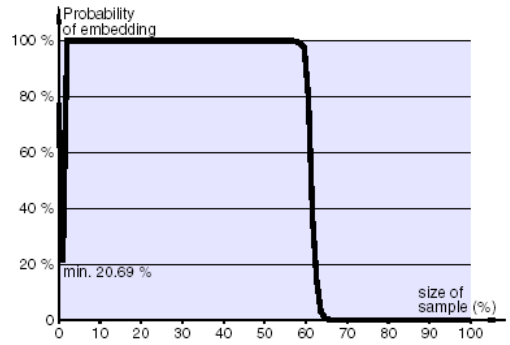

Рис. 2.8. Надійність вбудовування хі-квадрат в аналіз Jsteg

Алгоритм JPEG часто використовується для стиснення зображень (рис. 2.9).

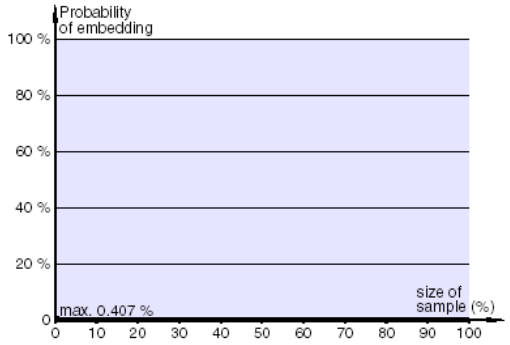

Рис. 2.9. Імовірність помилкового позитивного результату хі-квадрат для стиснення JPEG із пустим контейнером

На рис. 2.9 видно, що хибнопозитивна ймовірність використання критерію хі-квадрат при аналізі порожніх контейнерів, стиснутих за алгоритмом JPEG, не перевищує малого значення 0,407%, яким можна знехтувати.

Висновки до розділу 2

Одним з найкращих методів є приховання повідомлення в молодші біти коефіцієнтів перетворення файлу JPEG, що призводить до відносно невеликих змін у кожному з 256 пікселів, якы є візуально непомітні.

РОЗДІЛ 3. МАТЕМАТИЧНЕ ЗАБЕЗПЕЧЕННЯ

3.1. Напрями підвищення захищеності стегосистем від статистичних атак

Таким чином, різні стегосистеми, які використовують принцип заміни найменш значущих частин елементів контейнера бітами вбудованого повідомлення, виявилися нестійкими до статистичних атак. Їх стабільність можна підвищити різними способами, наприклад, шляхом переходу на операції вбудовування шляхом зваженої збірки елементів контейнера з вбудованими елементами повідомлень. Такі операції не збалансовують ймовірність появи відповідного контейнера та стего елементів, роблячи їх більш стійкими до аналізу їх статистики.

Очевидним шляхом є зменшення обсягу заповнення контейнера фрагментами прихованого повідомлення, тобто зменшення пропускної здатності стегоканалу в обмін на підвищення його безпеки. Статистичні атаки на основі критерію хі-квадрат, запропонованого в [14], у більшості випадків не можуть виявити стегоканал, коли контейнер заповнений на 50% або менше, особливо якщо введене повідомлення розкидано по всьому контейнеру. Ці атаки завжди починаються з початку досліджуваної послідовності та використовують рівномірно зростаюче вікно аналізу. Вони виявляють наявність стегоканалу, якщо статистична характеристика постійно спотворюється від початку контейнера. Невикривлені проміжні ділянки в контейнері можуть спричинити неправильний результат тесту. Тому в [15] була запропонована вдосконалена статистична атака, яку автор назвав розширеним тестом хі-квадрат. Тест використовує вікно аналізу фіксованого розміру, яке переміщується уздовж послідовності, що тестується. Ця атака виконує локальний пошук і дозволяє визначити розташування прихованого вкладення повідомлення. У тій же роботі

був запропонований метод підвищення безпеки від статистичних атак на стегосистеми шляхом вбудовування прихованого повідомлення в НЗБ контейнера. Процес вбудовування прихованої інформації в контейнер ділиться на 3 етапи:

1) визначення зайвих бітів, які можна замінити без пошкодження контейнера;
2) вибір НЗБ, в який буде вбудована прихована інформація;
3) корекція статистичних змін наявного стего.

Перший крок оцінює кількість контейнерних НЗБ, які можна замінити прихованими бітами повідомлення без втрати якості контейнера зображення. Насправді для вбудовування можна використовувати не більше половини виявлених бітів. Якщо знайдених додаткових бітів недостатньо, потрібно змінити контейнер. Потім секретний ключ використовується для визначення рівномірно розподілених депозитів безпеки всередині контейнера, які замінюються фрагментами прихованої інформації. Згенероване стего потім оцінюється за допомогою статистичних тестів, і якщо виявлено відхилення від статистичних характеристик природних бінів, решта зайвих бітів використовується для виправлення цих відхилень. Типовим методом корекції є збереження взаємної кореляції та значення ентропії, розрахованого за допомогою тесту Маурера. Дійсно, якщо під час вбудовування якийсь молодший біт змінюється з 0 на 1, тоді рекомендується змінити сусідній НЗБ-біт з 1 на 0 тощо. Хоча цей метод зберігає значення ентропії та коефіцієнт кореляції під час вкладення прихованого повідомлення в контейнер, він має макроскопічні статистичні недоліки. Це виражається в спотворенні гістограми коефіцієнтів ДКП. Якщо лівий коефіцієнт змінено так, щоб стегогістограма була такою самою, як гістограма вихідного відсіку, правий коефіцієнт має бути змінено на таку саму величину.

Регламентовані перетворення повинні відповідати вимогам:

1) для будь-якого даного фрагмента зображення розподіл стегокоефіцієнтів ДКП має бути подібним до їх розподілу в порожньому контейнері;
2) кількість поправок, необхідних для виправлення статистичних характеристик, має бути невеликою.

У роботі [15] наведено результати тестування алгоритму корекції вбудовування повідомлень у контейнерні зображення розміром 640X480 пікселів. Середня кількість коефіцієнтів ДКП, які можна використовувати для вбудовування, становить 46000 і коливається від 30000 до 97000. До вбудовування ймовірність збігу сусідніх надлишкових бітів становить 63,8% зі стандартним відхиленням ±3,4% по безлічі зображень. Довжина стисненого прихованого повідомлення становить 14700 біт. Перетворення, що коректують, привели до 2967±434 додатковим змінам в надлишкових бітах. Це становило приблизно 20% від розміру прихованого повідомлення. Середня кількість спотворень, які не вдалося виправити, становила 186-400.

У табл. 3.1 представлені результати статистичних тестів для перевіреного алгоритму. Видно, що без корекції коефіцієнт кореляції між надлишковими бітами помітно зменшився, а їх ентропія зросла. Виправлення робить вбудовування прихованих повідомлень статистично невиявленим.

Таблиця 3.1

Досліджувана послідовність	Коефіцієнт кореляції	Універсальний тест Маурера
Вихідний контейнер	63,77 % ± ,37 %	6,704 ± 0,253
Стего без корекції	59,10 % ± 3,19 %	6,976 ± 0,168
Стего з корекцією	62,91 % ± 3,36 %	6,775 ± 0,231

Таким чином, якщо коригувальні перетворення застосовуються до стего, то використані методи статистичного стеганатичного аналізу не можуть продемонструвати існування стегоканалу. Слід, однак, зазначити, що можуть бути створені й інші статистичні атаки, нейтралізація яких потребує додаткового використання надлишкових бітів, що ще більше знизить швидкість передачі прихованої інформації.

Загалом, вдосконалення стегосистем можна описати як ітераційний процес. Стегосистеми розроблені та запропоновані до використання авторами. Вони досліджуються за допомогою відомих методів стеганалізу, при необхідності для них розробляються нові методи аналізу і так далі, поки їх не вдасться зламати. Потім, враховуючи виявлені недоліки в принципах побудови стегосистеми, вони вдосконалюються, але при цьому розробляються стегоатаки. Цей процес триває ітераційно, поки не буде доведено, що на поточному рівні розвитку стеганалізу ця стегосистема є практично стабільною. Цей процес розвинувся в аналізі та синтезі криптосистем, і зрозуміло, що він також стосується стегосистем. Але потрібно враховувати, по-перше, що порівняно з криптосистемами стегосистеми мають додатковий параметр – контейнер, а по-друге, практична стійкість стегосистем може мати значно більшу кількість інтерпретацій.

3.2. Теоретичний підхід до оцінки стійкості стеганографічних систем

Інформаційно-теоретичні моделі стійкості стеганографічних систем, розглянуті в [2], [3], мають суттєві недоліки. Вперше це було помічено в статті[19]. Як зазначено в цій статті, теоретико-інформаційні методи, які успішно використовуються для аналізу криптосистем, погано підходять для аналізу стегосистем. Це пояснюється тим, що процедуру виявлення прихованих повідомлень не можна моделювати як безперервний процес. Насправді

зловмисник може отримати лише два результати аналізу підозрілого каналу зв’язку: або він виявляє наявність стегосистеми, або ні. Отже, ми маємо справу з розривним процесом, для якого методи теорії інформації непридатні. Це інше у випадку криптографії, де зловмисник може отримати часткове знання про відкрите повідомлення (або ключ), але система буде практично стабільною. За словами Шенона, стегосистема повинна бути абсолютно стабільною. На рис. 3.1 показана різниця між криптосистемами та стегосистемами на якісному рівні.

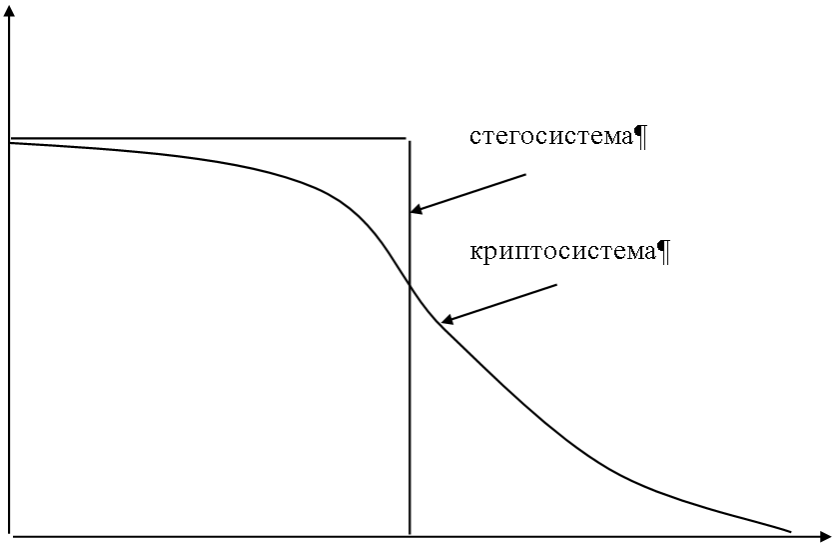

Рис. 3.1. Порівняння криптосистем і стегосистем. По осі ординат відкладено ступінь секретності системи, а по осі абсцис – обчислювальні ресурси поркшника

Усвідомлення того, що теоретико-інформаційні моделі малопридатні для аналізу стегосистем, призвело до появи теоретичних підходів до оцінки їх стійкості [20]. У цій роботі було по-новому досліджено концепцію стійкості стегосистеми та побудовано конструктивну модель стабільної стегосистеми у вигляді імовірнісної поліноміально-часової гри між зловмисником і приховувачем інформації. До основних недоліків інформаційно-теоретичних моделей стегосистем можна віднести:

1) Як і в криптографії, на практиці неможливо реалізувати абсолютно стабільну стегосистему. Можна показати, що реалізація такої стегосистеми зводиться до одного блоку (так званий шифр Вернама). Тому теоретико-інформаційні моделі стегосистем не є конструктивними.

2) Розподіл ймовірностей бінів невідомий на практиці або відомий з точністю якоїсь дуже, дуже приблизної моделі.

3) Використовувані контейнери зовсім не є реалізацією випадкового процесу, а найчастіше це оцифровані зображення реальних фізичних об’єктів.

4) Було б цілком реально припустити, що зловмисник має доступ лише до обмежених обчислювальних ресурсів. Як і в криптографії, вам потрібно лише вимагати, щоб стегосистема пройшла всі поліноміальні тести для її ідентифікації. Теоретичні інформаційні моделі також не враховують це питання.

Розглянемо модель стегосистеми, запропоновану в [20]. Припустимо, що існує набір можливих контейнерів, елементи яких генеруються деяким поліноміальним алгоритмом. Вбудоване повідомлення вибирається з набору можливих повідомлень. Стегосистема визначається за допомогою потрійних поліноміальних алгоритмів.

Алгоритм — це процес генерації ключа, який у відповідь на вхідний рядок одиниць генерує псевдовипадковий стегоключ. Відповідно до принципу Керхоффа, стабільність залежить від ключа, а його довжина є параметром секретності стегосистеми. Алгоритм реалізує інформацію, згенеровану на основі стего. Алгоритм отримує повідомлення за допомогою ключа, якщо контейнер дійсно містив вбудоване повідомлення. Щоб визначити наявність стегосистеми, зловмисник повинен вирішити наступну проблему: визначити з контейнера, чи існує ключ генерації повідомлення.

Атака (гра) відбувається наступним чином. Зловмисник мав можливість генерувати контейнери та відповідний їм стего кілька разів, намагаючись дізнатися структуру алгоритму стего. Існує обмеження, що вся процедура має бути поліноміальною з точки зору довжини ключа та розміру ящика. Після закінчення роботи він отримує дві навмання обрані ємності: одну порожню, іншу заповнену. Стегосистема називається умовно стабільною, якщо порушник не має можливості правильно визначити стего з імовірністю трохи більше 1/2. У [20] є визначення терміну «дещо відрізняється» та математичний опис моделі, наведеної вище. Умовно стабільна стегосистема підтримує цю властивість для всіх можливих ключів і всіх можливих контейнерів.

Очевидно, що концепція умовно стійкої стегосистеми є слабшою, ніж концепція стегосистеми, яка є інформаційно стійкою і включає її як окремий випадок. Звичайно, ми отримаємо стабільну стегосистему у наведеній вище моделі, якщо знімемо поліноміальне обмеження під час гри.

Як побудувати умовно стійку стегосистему? Однією з можливостей, яка широко використовується в криптографії, є взяти за основу якусь математичну задачу, яка є обчислювально складною, наприклад, інверсію односторонньої функції (розкладка на множники, дискретний логарифм тощо). Потім залишається продемонструвати зв'язок між неможливістю розв'язання цієї проблеми і неможливістю розкриття стегосистеми - і умовно стійка стегосистема побудована. З криптографії відомо, що, на жаль, питання побудови односторонньої функції не вирішено. У роботі [20] показано, як побудувати стегосистему на основі відомого криптоалгоритму RSA.

3.3. Імітостойкость системи передачі прихованих повідомлень

Раніше була перевірена стійкість стегосистем до спроб пасивного зловмисника визначити, чи приховуються повідомлення. Окрім вимог

секретності зв’язку, можуть існувати вимоги щодо запобігання нав’язуванню неправдивих повідомлень на стегоканалі активним суб’єктом. Наприклад, у творчості Г. Сіммонса т. зв проблема ув'язнених [6]. У цьому завданні заарештовані Аліса та Боб намагаються домовитися про втечу через прихований канал зв’язку. Охоронець Віллі намагається не тільки виявити факт обміну інформацією, але й нав'язати Бобу неправдиву інформацію від імені Аліси. Тому розглянемо особливості конструкції стегосистем з можливістю аутентифікації переданих повідомлень, можливі атаки зловмисників та визначимо оцінки імітації стегосистем.

Опишемо формально структуру стегосистеми з аутентифікацією таємно відправлених повідомлень. Нехай стегосистема використовує секретний ключ, який отримує значення. Набір бінів C розбивається на n підмножин, кожна з яких описується власним розподілом ймовірностей. Давайте зіставимо підмножину контейнерів із секретними ключами. З дійсним ключем автентифікації повідомлення, доставлене по секретному каналу зв’язку, вважається автентичним одержувачем, якщо воно міститься в контейнері, що належить до підмножини з розподілом. Якщо з дійсним ключем заповнений контейнер не належить до підмножини, повідомлення, отримане з нього, вважається одержувачем неправильним. Таким чином, за допомогою дійсних ключів весь набір контейнерів ділиться на дійсні, в яких одержувач розпізнає достовірність повідомлень, які вони містять, і недійсні, які відправник не може вибрати для передачі прихованих повідомлень. Відновлення таких контейнерів із вбудованими повідомленнями означає, що вони нав’язані зловмисником. Якщо отриманий stego S має розподіл, узгоджений з розподілом набору дійсних контейнерів з дійсним ключем, то функція перевірки автентичності прихованих у них повідомлень приймає одне значення, і отримане повідомлення вважається

дійсним, а якщо розподіли не збігаються, функція приймає нульове значення, і повідомлення відхиляється:

$$\boldsymbol{X}(S,K_i)=\begin{cases}1, & если \quad P_S \in P_{C_i},\\ 0, & если \quad P_S \notin P_{C_i}.\end{cases} \tag{3.1}$$

Функція автентифікації при побудові стегосистеми з автентифікацією повідомлень може бути задана аналітично, графічно або у вигляді таблиці. У аналітичному завданні кожне значення ключа пов’язується з власною підмножиною дійсних контейнерів. Ці підмножини відрізняються своїми законами розподілу або параметрами. Наприклад, використовуються різні розподіли ймовірностей безперервних бінів (Normal, Raisa, Nakagami та інші). Або підмножини контейнерів зображень відрізняються за спектральними характеристиками. Наприклад, у кожній підмножині енергія спектра зображення зосереджена у власному частотному діапазоні. Відомо, що зображення можна розділити на високочастотне, основна спектральна енергія якого належить верхній смузі частот, і низькочастотне. Ви також можете розділити контейнери зображень на підмножини залежно від типу об’єкта: пейзаж, портрет, натюрморт тощо. Хоча строго математично визначити функцію з точки зору законів розподілу при розподілі сюжетів важко, на практиці завдання такого функція не складна. Набір усіх бінів розбивається на n непересічних підмножин бінів. Наприклад, контейнери можна розділити на підмножини на основі їх перетину. Використовуючи правильний ключ, відправник вибирає підмножину контейнерів. Приховане повідомлення вбудовується в контейнер цієї підмножини, утворюючи стегограму. Одержувач стегограми перевіряє, чи відповідає вона правильному ключу. Перевіряє, чи є результуюча стегограма дійсною, враховуючи дійсний ключ. Ця рівність виконується, якщо стегограма належить до підмножини контейнерів. Таким чином, повідомлення, витягнуте зі

стегограми, є автентичним. Проте, якщо отримана стегограма не належить до дійсної підмножини контейнерів, функція перевірки обчислює нуль, а отримане повідомлення відхиляється як помилкове. Графічний опис функції автентифікації показано на рис. 3.2. Дозволити надсилання різних повідомлень через stegochannel. Набір ключів стегосистеми складається з n ключів, з яких вибирається дійсний і випадковий ключ.

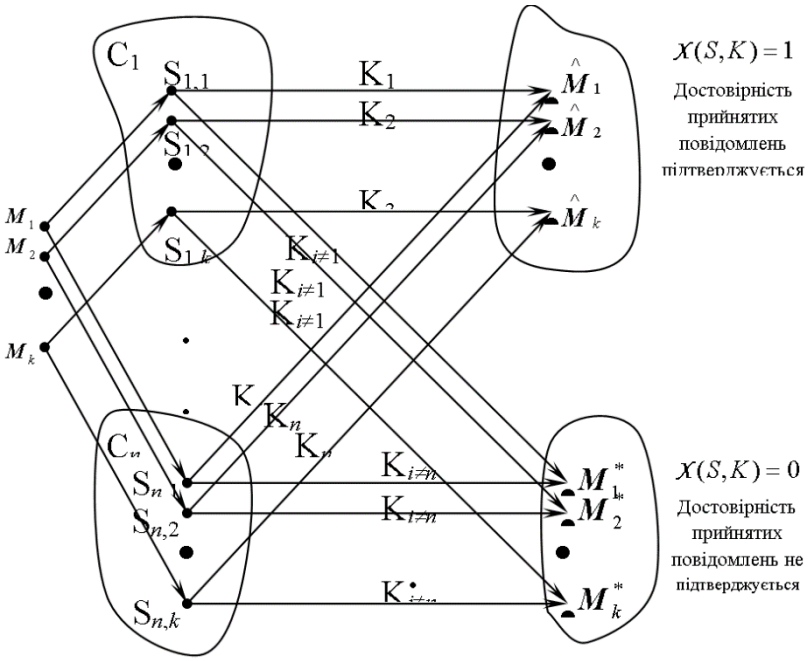

Рис. 3.2. Графічний опис функції автентифікації прихованого повідомлення

З рис. 3.2 легко побачити, що підмножини бінів мають однаковий розмір. Якщо приховані повідомлення однаково ймовірні, а вибрана ключова інформація рівноймовірна, то для зловмисника, який не знає дійсного ключа, набір повідомлень, автентичність яких буде підтверджено під час перевірки, буде в n-1 разів менший за набір повідомлень. повідомлень, відхилених під час перевірки як фейкових.

Розглянемо можливі атаки зловмисників на достовірність прихованих повідомлень та оцінимо імітацію стегосистем під ці атаки. З криптографії відомо,

що активний зловмисник може здійснити імітаційну атаку або атаку підміни [13]. Під час атаки спуфінгу, також відомої як спуфінг порожнього каналу, зловмисник, не чекаючи перехоплення автентифікованого повідомлення, створює помилкове повідомлення від імені відправника. Позначимо ймовірність успіху зловмисника в імітаційній атаці через *M*. З рис. 3.2 очевидно, що для зловмисника, який не знає дійсного ключа і нав'язує будь-яке повідомлення з набору, ймовірність успіху не може бути меншою за кількість усіх повідомлень, поділену на кількість усіх стегограм *S*:

$$P_i \geq \frac{|M|}{|S|} = \frac{k}{n \cdot k}. \qquad (3.2)$$

Критерій Сіммонса для систем автентифікації стверджує, що вираз (3.2) задовольняється, якщо виконуються дві умови:

1. Імітаційна атака є оптимальною, тобто вона має однакову ймовірність успіху для зловмисника, враховуючи однаково ймовірний випадковий вибір будь-якої нав’язаної стегограми.
2. Для кожної стегограми ймовірність її створення відправником однакова для всіх ключів автентифікації, для яких.

Якщо ці умови виконуються, то для заданого розміру набору прихованих повідомлень і набору стегограм ймовірність шахрайства мінімальна. Слідуючи Сіммонсу, стегосистему з прихованою автентифікацією повідомлень можна назвати ідеальною щодо імітаційної атаки, якщо вона задовольняє рівність виразів (3.2). З виразу (3.2) випливає, що ймовірність шахрайства низька, тобто забезпечується висока імітаційність стегоканалу. Слід зазначити, що в жодному правилі побудови стегосистеми не можна отримати величину, меншу за наведену у виразі (3.2).

У другій стратегії імітації стегоканалу, яка називається атакою заміни першого порядку, зловмисник, перехопивши стегограму від законного відправника, замінює її недійсною. Атака заміни вважається успішною, якщо нав’язане стего декодується одержувачем у будь-яке сповіщення, дійсне для даної стегосистеми, і помилкове повідомлення не може збігатися з дійсним сповіщенням від законного відправника. Позначимо ймовірність шахрайства в сурогатній атаці *M*. Якщо зловмисник перехоплює стего, що містить якесь невідоме йому повідомлення, і замінює його будь-яким іншим стего, то очевидно (див. рис. 2.9), що для непересічних підмножин *S* будь-якого повідомлення отримане зі стего за дійсним ключем не буде одночасно розпізнано як справжнє одержувачем і збігається з дійсним сповіщенням, надісланим законним відправником. Тож зловмисник має шанс нав’язати одне з повідомлень, що залишилися, використовуючи один із стего. Таким чином, ймовірність успішного виконання заміни першого порядку в атаці не перевищує

$$P_d \le \frac{|M|-1}{|S|-1} = \frac{k-1}{n \cdot k-1}. \qquad (3.3)$$

Зауважте, що, як і у випадку з імітаційною атакою, стегоканал дуже стійкий до імітації у випадку сурогатної атаки першого порядку. Наведені вище умови є необхідними, але недостатніми для задоволення виразу (3.3) зі знаком рівності. Давайте визначимо стегосистему з прихованою автентифікацією повідомлень, яка є ідеальною щодо атаки підстановки першого порядку, якщо вона задовольняє рівність у виразі (3.3).

Пояснимо на простому прикладі стратегію імітації та оцінки захисту від шахрайства для наступного типу стегосистеми. Створимо табличний опис функції аутентифікації, наведеної в табл. 3.1. Попросіть двох ув’язнених, Алісу та Боба, погодитися пізніше створити секретний канал передачі з

автентифікацією повідомлень. Для цього вони заздалегідь (до арешту) погодилися, що приховані повідомлення відповідають умовним сигналам. Вони також визначили, що з дійсним ключем деякі повідомлення дійсні (Аліса може їх надіслати), а інші недійсні (Аліса їх не надсилатиме). Табл. 3.2 вказує, які повідомлення є дійсними, якщо ключ автентифікації (або) дійсний.

Нехай Аліса і Боб організують передачу прихованих повідомлень наступним чином. Щоранку Боба виводять на прогулянку, і він спостерігає за вікном камери Аліси. Для таємної передачі повідомлень Аліса ставить у вікно комірки горщики з геранню, кількість яких дорівнює номеру умовного сигналу згідно з табл. 3.2. Якщо ключ автентифікації дійсний для цього дня, то повідомлення «вихід сьогодні» відповідає 2 банкам, а повідомлення «вихід скасовано» відповідає 6 банкам.

Таблиця 3.2

Приховувані повідомлення	Номер умовного сигналу	Приховувані повідомлення	Номер умовного сигналу	Діючий ключ аутентифікації
Втеча сьогодні	2	Втеча скасована	6	K_1
Сьогодні втеча	5	Скасована втеча	3	K_2
Втеча призначена на сьогодні	1	Втеча сьогодні скасована	4	K_3

Давайте розглянемо можливі стратегії введення тюремним охоронцем Віллі помилкової інформації в цей таємний канал зв'язку. Перший варіант Віллі реалізується через імітаційну атаку. Охоронець припускає, що через квіти можна донести приховану інформацію. Не чекаючи дій Аліси, вона виставляє кілька горщиків з геранню за вікном своєї камери. У випадку з 2 або 6 елементами, коли Боб отримує фальшиве повідомлення, він вважає, що воно насправді надіслано Алісою, оскільки ці повідомлення дійсні за умови, що ключ дійсний. У цих випадках зловмиснику вдалося нав'язати неправильне повідомлення, хоча Віллі

не знає, що це таке. Але якщо Віллі вибере для імітації умовні сигнали 1, 3, 4 або 5, Боб чітко встановить, що отримане повідомлення було навіяно зловмисником.

Тому, враховуючи однаково ймовірний вибір неправильного повідомлення, ймовірність успіху Віллі в імітаційній атаці дорівнює $P_i = 1/3$.

Розглянемо другу імітаційну стратегію – заступну атаку першого порядку. Віллі зауважує, що Аліса, наприклад, поставила за вікном 2 горщики з квітами. Охоронець припускає, що це приховане повідомлення, і змінює умовний сигнал на інший. Якщо Віллі накладає умовний сигнал 1, 3, 4 або 5, Боб вважатиме, що отримане повідомлення є неправильним. Але якщо Віллі використовує умовний сигнал 6, то симуляція буде успішною і замість сигналу «втеча сьогодні» Боб отримає сигнал «втечу скасовано» з усіма витікаючими наслідками. Таким чином, у цій атаці заміни ймовірність успішного нав'язування помилкового повідомлення, якщо вони однаково ймовірно будуть обрані, дорівнює $P_d = 1/5$.

Виявилось, що $P_d < P_i$, однак слід зазначити, що успіх ґвалтівника в атаці підміни завдає більшої шкоди порівняно з імітацією атаки, оскільки після успіху атаки підміни ґвалтівнику вдається нав'язати діаметрально протилежне повідомлення. Зауважте, що, навпаки, у випадку атаки з імітацією нав'язування вона вважається успішною, якщо зловмиснику вдалося нав'язати будь-яке повідомлення, навіть таке, яке відповідає тому, яке мала намір надіслати Аліса.

Описана стегосистема фактично використовує лише два прихованих повідомлення: «рейс сьогодні» та «рейс сьогодні скасовано», надісланих за допомогою 6 стегограм. Слід зазначити, що незважаючи на простоту цієї стегосистеми, її використання забезпечує рівноправність виразів (3.2) і (3.3), тобто одночасно реалізується імітаційна атака та атака заміни першого порядку.

У стегосистемах з аутентифікацією, порівняно з криптосистемами, що забезпечують контроль достовірності переданих повідомлень, виникає така практична проблема. У спуфінговій атаці не так важливо, як розділити набір контейнерів на підмножину, оскільки для зловмисника в момент накладення всі контейнери (стегограми) однаково ймовірні. Інша ситуація в атаці на заміну. Якщо, захопивши стегограму, зловмисник зможе виявити, до якої підмножини контейнерів вона належить, тоді зловмисник повністю або частково визначив правильний ключ і виявив можливість накласти його з неприйнятно високою ймовірністю. Тому, щоб забезпечити високу імітаційність системи стего, визначення, до якої підмножини належить даний стего, повинно бути обчислювально складним. Очевидний спосіб досягти цього — випадковим чином розділити набір C на підмножини $C_1, C_2, \ldots, C_n$. Результатом цього поділу є секретний ключ автентифікації, який має бути відомий лише законному відправнику та одержувачу автентифікованих повідомлень. Однак обсяг цієї секретної інформації занадто великий для практичних стегосистем. Другий метод полягає у формуванні або виборі контейнерів відповідно до функцій формування або вибору з використанням секретної інформації автентифікації з обмеженим обсягом, забезпечуючи при цьому автентичність. Якщо отримане стего можна згенерувати або вибрати за допомогою правильного ключа, тоді витягнуте з нього повідомлення вважатиметься автентичним. Подібні функції відомі в криптографії та є стійкими до аналізу зловмисників [8]. Однак значна складність полягає в тому, що такі стабільні функції повинні генерувати не лише послідовності, які обчислювально не відрізняються від випадкових, але й послідовності, які також не відрізняються від послідовностей, створених природними джерелами (мова, відео).

У криптографічних системах контроль достовірності інформації, що передається, забезпечується за допомогою імітацій або цифрових підписів [8]. Імітації вставок і цифрових підписів сертифікованих повідомлень описуються законом розподілу Бернуля [14]. Тому вони можуть бути легко помічені зловмисником, на відміну від контейнерів з природними джерелами, що погіршує секретність повідомлень, сертифікованих стегоканалом. Як наслідок, стійкі до імітації стегосистеми не можуть копіювати принципи побудови криптографічних систем контролю достовірності інформації, що передається.

Підводячи підсумок, зазначимо, що стегосистеми з аутентифікацією таємно переданих повідомлень теоретично і практично знаходяться на дуже ранній стадії розробки і чекають досліджень.

Більшість досліджень присвячено використанню зображень як стегоконтейнерів. Це відбувається з наступних причин:

- наявність практично важливого завдання захисту фотографій, зображень, фільмів від незаконного відтворення та розповсюдження;
- відносно великий обсяг представлення цифрових зображень (ЦПЗ), що дозволяє реалізувати ЦПЗ великого обсягу або підвищити продуктивність реалізації;
- розмір контейнера відомий заздалегідь, ніяких обмежень не вимагає реального;

Не випадково стегоалгоритми враховують властивості зорової системи (СЛЗ) людини, як і алгоритми стиснення зображення. Інформацію можна вводити у вихідне зображення одночасно зі стисненням зображення-контейнера або в уже стисненому алгоритмі зображення JPEG. Тому розглядаються властивості людського зору та їх включення в алгоритми стиснення зображення.

3.3. Принципи стиску зображень

Квантування нульового дерева базується на спостереженні, що якщо коефіцієнт малий, його діти в дереві часто також малі. Це пояснюється тим, що значні коефіцієнти з'являються біля контурів і локальних текстур. Неважко зрозуміти, що це своєрідне передбачення. Можна припустити, що якщо будь-який коефіцієнт незначущий, то всі його нащадки також будуть незначущими. Дерево або піддерево, яке містить (або принаймні може це робити) лише менші коефіцієнти, називається нульовим деревом.

У [3] запропоновано наступний алгоритм квантування вейвлет-коефіцієнта. Спочатку кожен вузол квантується квантувачем, оптимальним для щільності розподілу Лапласа. Якщо значення вузла менше за вказане порогове значення, його дочірні елементи ігноруються. Ці діти будуть відновлені декодером як нулі. В іншому випадку він переходить до чотирьох дочірніх вузлів і повторює процедуру. Якщо вузол не має дітей (є листом), починається обробка наступного кореневого вузла і так далі.

Цей алгоритм ефективний з двох причин. По-перше, за рахунок гарної «упаковки» енергії через вейвлет-перетворення, а по-друге, за рахунок спільного кодування нулів. Кодер довжини серії зазвичай використовується для кодування нулів. Щоб покращити вхідну продуктивність цього кодера, коефіцієнти мають подаватися в певному порядку. Наприклад, JPEG використовує зигзагоподібне сканування. Можливо, найважливішим внеском цієї роботи була демонстрація того, що область вейвлет-коефіцієнта добре підходить для роботи кодера довжини хвилі. По суті, генерується величезна серія нулів, і немає необхідності передавати їхню довжину, оскільки відома висота дерева. Як і JPEG, цей алгоритм є типом скалярно-векторного квантування. Кожен (значущий) коефіцієнт квантується окремо, а знаки, що відповідають малим коефіцієнтам,

утворюють вектор. Цей вектор складається з нульового символу дерева та послідовності нулів до кінця дерева.

Більшість алгоритмів стиснення зображень на основі вейвлет-перетворення мають можливість виділяти дві складові швидкості та дві складові спотворення. Алгоритми оптимізують розподіл бітів між цими компонентами, враховуючи обмеження загальної швидкості кодування зображення.

Одна з компонент пов'язана з «обнуленням» коефіцієнтів, які не перевищують певний поріг, інша пов'язана з квантуванням великих коефіцієнтів («значущих») і передачею їх розташування. Ефективність алгоритму стиснення залежить від правильного визначення порогу для прийняття рішень про значущість коефіцієнтів, а також від обраного методу квантування значущих коефіцієнтів і способу передачі інформації про їх розташування.

3.4. Приховування даних в просторовій області

Стегокодер з використанням ШПС показаний на рис. 3.3. Приховане повідомлення зашифровано ключем $k1$ і кодується завадостійким кодом, внаслідок чого виходить кодоване повідомлення m. Це повідомлення модулюється псевдовипадковою послідовністю з виходу генератора, початкове заповнення якого дорівнює $k2$. Отриманий сигнал з розширеним спектром піддається перестановкам відповідно до ключа $k3$ і складається із зображенням-контейнером. Декодер виконує зворотні дії. Кореляційний приймач служить детектором ЦПЗ.

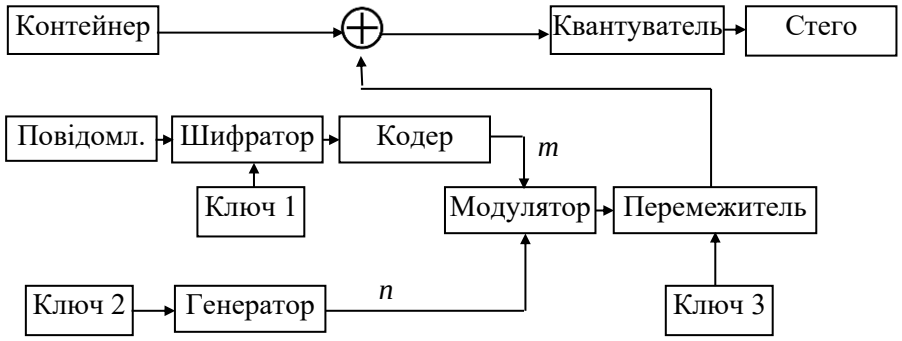

Рис. 3.3. Стегокодер на основі шумоподібної послідовності

Генератор послідовності на основі ідеальних кільцевих шаблонів найчастіше пропонується як датчик псевдовипадкової послідовності, оскільки ця послідовність має хороші кореляційні властивості.

3.5. Приховування даних в коефіцієнтах дискретного косинусного перетворення

Використання ДКП для приховування інформації вперше описано в [17]. У цьому випадку ДКП застосовувався до всього зображення в цілому.

Як правило, контейнер ділиться на блоки 8x8 пікселів. ДКП застосовується до кожного блоку, в результаті чого утворюються матриці коефіцієнтів ДКП, також розміром 8x8. Коефіцієнти позначимо через $c_b(j,k)$, де b- номер блоку, коефіцієнти позначатимемо через $c_{b,j}$. Коефіцієнт в лівому верхньому кутку $c_b(0,0)$ зазвичай називається DC-коефіцієнтом. Містить інформацію про яскравість всього блоку. Ці останні коефіцієнти називаються коефіцієнтами AC. Іноді ДКП виконується на всьому зображенні, а не на окремих блоках. Давайте розглянемо деякі запропоновані алгоритми впровадження ЦПЗ у сфері публічних комунікацій ДКП.

У цьому алгоритмі 1 біт ЦПЗ вбудовано в блок 8x8.

Вбудовування інформації здійснюється наступним чином: для передачі біта 0 різниця абсолютних значень коефіцієнтів більше деякого позитивного значення, а для передачі біта 1 ця різниця менше деякого негативного значення:

$$\begin{aligned} &\left| c_b\left(j_{i,j},k_{i,1}\right)\right| - \left| c_b\left(j_{i,2},k_{i,2}\right)\right| > \varepsilon, \quad \text{если } s_i = 0, \\ &\left| c_b\left(j_{i,j},k_{i,1}\right)\right| - \left| c_b\left(j_{i,2},k_{i,2}\right)\right| < -\varepsilon, \quad \text{если } s_i = 1. \end{aligned} \tag{3.4}$$

Для читання ЦПЗ в декодері виконується така сама процедура:

$$\begin{aligned} &s_i = 0, \quad \text{если } \left| c_b\left(j_{i,j},k_{i,1}\right)\right| > \left| c_b\left(j_{i,2},k_{i,2}\right)\right|, \\ &s_i = 1, \quad \text{если } \left| c_b\left(j_{i,j},k_{i,1}\right)\right| < \left| c_b\left(j_{i,2},k_{i,2}\right)\right|. \end{aligned} \tag{3.5}$$

Цей алгоритм можна вважати вдосконаленою версією попереднього. Блоки зображень, які не дуже гладкі і не містять невеликої кількості контурів, вважаються вбудованою інформацією.

Вбудовування біта ЦПЗ виконується наступним чином. Три коефіцієнти ДКП блоку вибираються псевдовипадково. Якщо ви повинні інвестувати 1, коефіцієнти змінюються таким чином, що третій коефіцієнт стає меншим за кожний з перших двох; якщо ви хочете вбудувати 0, він буде більшим за решту. У цьому випадку, якщо така модифікація призводить до дуже сильного погіршення зображення, коефіцієнти не змінюються і цей блок просто не використовується.

Зміна трьох коефіцієнтів замість двох, а тим більше відмова від їх зміни при неприпустимих спотвореннях, зменшує похибки, які вносить центральний блок. Декодер завжди зможе ідентифікувати блоки, де ЦПЗ не вбудовано, повторивши аналіз, виконаний у кодері.

При виявленні ЦПЗ цей алгоритм вимагає наявності вихідного зображення в детекторі. Вбудовані дані моделюються за допомогою справжнього випадкового процесу з нормальним розподілом, одиничною дисперсією та

нульовим середнім. Для кожного коефіцієнта ДКП визначається порогове значення, вище якого зміна може призвести до погіршення зображення. Цей поріг залежить від положення коефіцієнта в матриці (тобто частотного діапазону, за який він відповідає). Причому поріг визначається властивостями самого зображення: контрастністю і яскравістю блоку.

Вбудовування відбувається наступним чином: якщо абсолютне значення коефіцієнта менше порогового значення, воно не змінюється. В іншому випадку до нього додається добуток порогового значення та значення цифрового водяного знака ЦВЗ.

Під час виявлення ЦВЗ коефіцієнти вихідного зображення спочатку віднімаються з відповідних коефіцієнтів модифікованого зображення. Потім розраховується коефіцієнт кореляції та визначається наявність ЦПЗ.

У цьому алгоритмі декодера ЦПЗ також потрібне оригінальне зображення. Однак декодер не визначає наявність модуля цифрової обробки, а вибирає вбудовані дані. ЦПЗ — це чорно-біле зображення розміром у половину контейнера.

Для впровадження біта ЦПЗ s_i в коефіцієнт $c_b(j,k)$ знаходиться знак різниці коефіцієнта поточного блоку і відповідний йому коефіцієнт з попереднього блоку

$$d_1(i) = sign(c_b(j,k) - c_{b-1}(j,k)). \tag{3.6}$$

Якщо треба вбудувати 1, коефіцієнт $c_b(j,k)$ змінюють так, щоб знак різниці став додатнім, якщо 0 - то щоб знак став від’ємним.

Автори також запропонували низку вдосконалень основного алгоритму: нарешті, був включений процес квантування коефіцієнтів:

$$d_2(i) = sign\left(\left\lfloor \frac{|c_b(j,k)|}{Q(j,k)} \right\rfloor Q(j,k) - \left\lfloor \frac{|c_b(0,0)|}{Q(0,0)} \right\rfloor Q(0,0)\right). \tag{3.7}$$

Ще одним удосконаленням цього алгоритму є запропонований авторами порядок сортування блоків цифрових водяних знаків. Блоки ЦВЗ розташовані в порядку спадання кількості одиниць, які вони містять. Блоки вихідного зображення контейнера також упорядковані в порядку спадання дисперсії. Потім виконується відповідне вкладення даних.

Зверніть увагу, що цей алгоритм не захищений від стиснення JPEG.

Для виявлення ЦВЗ детектору потрібен вихідний бункер. Під час вбудовування ДКП використовуються коефіцієнти ДКП, які мають мінімальний крок квантування в таблиці JPEG. Кількість і розташування цих коефіцієнтів не залежить від зображення.

Алгоритм роботи наступний. По-перше, блоки поділяються на 6 категорій, в залежності від ступеня гладкості і наявності контурів. Для кожного блоку розраховується адитивний коефіцієнт чутливості до шуму і блоки впорядковуються відповідно до цього коефіцієнта. Потім енергія вбудованого ЦВЗ визначається або цим коефіцієнтом (залежно від зображення), або на етапі квантування (незалежно від зображення) (залежно від того, що більше).

Щоб виявити ДКП, вихідне зображення спочатку віднімається від отриманого зображення. Потім розраховується ЦВЗ вихідного та різницевого зображень і використовуються статистичні методи для перевірки гіпотез.

Цей алгоритм є роботизованим для багатьох операцій обробки сигналу. Цифровий водяний знак, вбудований у нього, ідентифікується з оригінального зображення. Вхідні дані — це послідовність дійсних чисел із нульовим середнім і одиничною дисперсією. Щоб вставити інформацію, для вбудовування інформації використовуються кілька найвищих коефіцієнтів ДКТ змінного струму для всього зображення. Автор запропонував три способи вбудовування ЦВЗ цифрових водяних знаків відповідно до наступних виразів:

$$c_i^{'} = c_i + \alpha s_i, \tag{3.8}$$

$$c_i^{'} = c_i(1 + \alpha s_i) \tag{3.9}$$

і

$$c_i^{'} = c_i e^{\alpha s_i}. \tag{3.10}$$

Вираз (3.10) можна використовувати, коли енергія ЦВЗ порівнянна з енергією модифікованого коефіцієнта. В іншому випадку або ЦВЗ буде простоювати, або спотворення будуть дуже великими. Тому вбудовувати інформацію таким чином можна тільки при невеликому діапазоні зміни енергетичних значень коефіцієнтів.

Після виявлення цифрового зображення виконуються зворотні операції: розраховується ДКП вихідного та модифікованого зображень і знаходять різниці між відповідними коефіцієнтами найбільшої величини.

Цей алгоритм є вдосконаленням попереднього і також виконує ДКП всього зображення. У ньому детектор більше не потребує вихідного зображення, тобто схема є сліпою. Для вбудовування ЦПЗ використовуються не найбільші коефіцієнти АС, а середні. Будь-який рядок бітів діє як ЦПЗ

Вибрані коефіцієнти модифікуються наступним чином:

$$c_i^{'} = c_i + \alpha s_i \left| c_i \right|. \tag{3.11}$$

Потім виконується інверсний ДКП і виконується додатковий етап обробки: вихідне та модифіковане зображення додаються до вагових коефіцієнтів:

$$l^{''}(x, y) = \beta(x, y) l^{'}(x, y) + (1 - \beta) l(x, y). \tag{3.12}$$

Тут $\beta \approx 1$ для текстурованих областей. β доцільно використовувати нормалізовану дисперсію блоків.

У детекторі ЦВЗ обчислюється кореляція між модифікованим зображенням і ЦВЗ, $\sum_{i=1}^{n} c_i'' s_i''$.

Як показали автори, каскадне використання двох різних алгоритмів дає хороші результати з точки зору робототехніки.

Для цього автори використали наступне перетворення

$$I \to \frac{1024}{\sqrt{XY}} \frac{I - \hat{I}}{\sigma(I)}, \tag{3.13}$$

де $\sigma(I)$- відхилення стандартне, $\hat{I}$ - середнє значення яскравості. ЦВЗ є послідовністю чисел $\{-1;1\}$.

Потім на основі послідовності дійсних чисел, визначених виразом, будується функція індексування

$$t_0 = 1, t_{i+1} = \frac{1+\alpha}{1-\alpha} t_i, \tag{3.14}$$

де $\alpha \in (0,1)$ параметр . Індексна функція

$$ind(t) = (-1)^i, \qquad якщо\ t \in [x_i, x_{i+1}). \tag{3.15}$$

Індекс зміниться лише в тому випадку, якщо до t додати/відняти число, що перевищує значення αt.

Для впровадження біта ЦВЗ s_i в коефіцієнт c_j останній змінюється не менше, ніж на 100α відсотків так, щоб $ind(|c_j'|) = s_i$. Якщо буде значення коефіцієнта мале (менше 1), то в нього інформація не вбудовується.

$$Corr(I, I') = \frac{\sum_i |c_j'|^{\beta} ind(|c_j'|) s_i}{\sum_i |c_j'|^{\beta}}, \tag{3.16}$$

де β параметр визначає зважування.

Автори пропонують використовувати $\beta \in (0.5,1)$.

Якщо зображення було змінено, це стандартне відхилення $\sigma(I')$ відрізняється від $\sigma(I)$. Знаючи $s = \sigma(I)/\sigma(I')$ можна було б уточнити вираз для коефіцієнта кореляції:

$$Corr(I, I', s) = \frac{\sum_i |c_j'|^{\beta} ind(s|c_j'|)s_i}{\sum_i |c_j'|^{\beta}}. \quad (3.17)$$

Проте, як було вказано, значення s вибирається таким чином, щоб максимізувати значення коефіцієнта кореляції:

$$Corr(I, I') = \max_{s\in(1-\Delta;1+\Delta)} Corr(I, I', s). \quad (3.18)$$

Інформація вбудовується в середні частотні коефіцієнти ДКП шляхом множення перетвореного значення ЦВЗ на параметр α і додавання результату зі значенням коефіцієнта. Початкове кодування ЦВЗ виконується за наступним алгоритмом.

Вхід алгоритму: повідомлення довжиною M, що складається з символів $m_i \in \{1,...,B\}$.

Вихід алгоритму: ЦВЗ N довжини, що складається з дійсних чисел s_i.

Для кодування символу m_i генерується $N + B + 1$ чисел псевдовипадкової послідовності $r_i \in \{-1,1\}$. Цю послідовність називатимемо i-м випадковим вектором.

Перші m_i чисел цього вектора пропускаються, а наступні N чисел утворюють вектор V_i, що використовується при подальшому додаванні.

Як ЦВЗ використовується сума векторів V_i. Якщо M досить велика, то ЦВЗ матиме розподіл Гауса. i-й символ вихідного повідомлення можна отримати після обчислення взаємної кореляції ЦВЗ з i-м випадковим вектором, де N змінює значення від 1000 до 10000.

При достатній енергії ЦВЗ з'являється блокуючий артефакт, а також при високих ступенях стиснення в стандарті JPEG. Перетворення ортогонального перекриття (ПОП) спочатку було запропоновано для подолання недоліків ДКП у стисненні зображення. У [25] було запропоновано використовувати його для вбудовування інформації. Результатом є алгоритм, який достатньо стійкий до багатьох атак.

3.6. Алгоритм вбудовування цифрового водяного знаку

Щоб описати структуру файлу JPEG, я описав кілька етапів перетворення інформації бітового потоку в піксельний колір. Відразу встановимо, що, наприклад, функція GetByte повертає 1 поточний байт з файлу (або потоку) і переміщує покажчик файлу на наступний байт. Функція GetBytes(Count) повертає вказану кількість байтів і переміщує покажчик файлу на кількість прочитаних байтів. Прочитавши перші два байти з будь-якого файлу JPEG, ви можете переконатися, що вони завжди дорівнюватимуть FF16 і D816 відповідно. А тепер повернемося до тегів. Кожен маркер починається з байта FF16, після якого йде байт, що вказує на тип маркера. Зверніть увагу, що, наприклад, такі значення, як FF16 і C016, не є тегом, якщо вони знаходяться в області даних іншого тегу, тобто теги, що описують структуру файлу JPEG, не мають субмаркерів, тому в цьому випадку значення FF16 C016 не є тегом, хоча воно присутнє у файлі.

3.6.1. Маркери

Нижче в табл. 3.3. наведено ідентифікатори всіх можливих тегів, які зустрічаються під час декодування файлів JPEG.

Таблиця 3.3

Маркери формату JPEG

Тип маркеру	Ідентифікатор	Визначення стандартом	Зміст
1	2	3	4
SOF_0	$C0_{16}$	Baseline DCT	Початок кадру, базовий метод
SOF_1	$C1_{16}$	Extended sequential DCT	Початок кадру, розширений, послідовний метод
SOF_2	$C2_{16}$	Progressive DCT	Початок кадру, прогресивний метод
SOF_3	$C3_{16}$	Lossless (sequential)	Початок кадру, метод стиску без втрат
SOF_5	$C5_{16}$	Differential sequential	Початок кадру, диференційний
		DCT	послідовний метод
SOF_6	$C6_{16}$	Differential progressive DCT	Початок кадру, диференційний прогресивний метод
SOF_7	$C7_{16}$	Differential lossless (sequential)	Початок кадру, диференційний метод стиску без втрат
JPG	$C8_{16}$	Reserved for JPEG extensions	Резерв для наступних розширень формату JPEG
SOF_9	$C9_{16}$	Extended sequential DCT	Початок кадру, розширений послідовний метод, арифметичне кодування
SOF_{10}	CA_{16}	Progressive DCT	Початок кадру, прогресивний метод, арифметичне кодування
SOF_{11}	CB_{16}	Lossless (sequential)	Початок кадру, метод стиску без втрат, арифметичне кодування
SOF_{13}	CD_{16}	Differential sequential DCT	Початок кадру, диференційний послідовний метод, арифметичне кодування
SOF_{14}	CE_{16}	Differential progressive DCT	Початок кадру, диференційний прогресивний метод, арифметичне кодування
SOF_{15}	CF_{16}	Differential lossless (sequential)	Початок кадру, диференційний метод без втрат, арифметичне кодування
DAC	CC_{16}	Define arithmetic coding conditioning(s)	Визначення умов арифметичного кодування

Продовження табл. 3.3

1	2	3	4
DHT	$C4_{16}$	Define Huffman table(s)	Визначення таблиць Хафмана
RST_0… RST_7	$D0_{16}$...$D7_{16}$	Restart marker number 0…7	Визначення проміжку перезапуску від 0 до 7
SOI	$D8_{16}$	Start of image	Початок зображення
EOI	$D9_{16}$	End of image	Кінець зображення
SOS	DA_{16}	Start of scan	Початок скіну
DQT	DB_{16}	Define quantization table(s)	Визначення таблиць квантування
DNL	DC_{16}	Define number of lines	Визначення числа ліній
DRI	DD_{16}	Define restart interval	Визначення проміжку перезапуску
DHP	DE_{16}	Define hierarchical progression	Визначення ієрархічної прогресії
EXP	DF_{16}	Expand reference component(s)	Відкриття додаткових компонент
COM	FE_{16}	Comment	Коментар
APP_0… APP_{15}	$E0_{16}$...EF_{16}	Reserved for application segments	Зарезервована інформація про додатки
JPG_0… JPG_{13}	$F0_{16}$...FD_{16}	Reserved for JPEG extensions	Резерв для наступних розширень формату JPEG
TEM	01_{16}	For temporary private use in arithmetic coding	Для тимчасового локального визначення налаштувань арифметичного кодування
RES	02_{16}...BF_{16}	Reserved	Резерв для наступних розширень формату JPEG

У наведеній табл. 3.3. теги, які майже завжди присутні в будь-якому файлі JPEG і вимагають обов'язкової обробки, виділені зеленим кольором. Поширені теги, які не вимагають обов'язкової обробки для отримання зображення, також

виділені жовтим кольором. Решта тегів не містяться у файлах JPEG у 90% випадків і не потребують обробки.

Ось структура типового тегу (рис. 3.4).

Ідентифікатор	Довжина	Данні маркеру

Рис. 3.4. Структура маркера

Основна ідея алгоритму полягає в тому, щоб розділити інформацію в зображенні за рівнем важливості, а потім відкинути менш важливу частину, тим самим зменшивши загальний обсяг збережених даних.

Це досягається шляхом перетворення матриці значень кольору в матрицю амплітуд, відповідних конкретним частотам розподілу зображення. (Наприклад, звукові коливання можна математично розділити на прості синусоїдальні гармоніки різних амплітуд і частот, якщо додати їх для відтворення вихідного сигналу.) Рядок або стовпець пікселів зображення також можуть бути представлені амплітудами та частотами. В даному випадку мова йде не про спектральний склад світла, а про форму подання кривих, що складають графіки, якщо координатами служать значення пікселів. Зверніть увагу, що формула для перетворення піксельної матриці в матрицю амплітуди не проста. Стиснення JPEG відхиляє деякі високочастотні компоненти зображення, залишаючи низькочастотні компоненти. Людське око менш критично ставиться до високочастотних змін кольору, оскільки загальний вигляд зображення залежить від низьких частот. Значення пікселів, отримане під час реконструкції зображення, дещо відрізняється від початкового значення, оскільки деяка інформація була втрачена, хоча вони зазвичай дуже близькі.

3.6.2. Кроки квантування для області частот

Розглянемо кроки квантування для областей нижніх та високих частот

6	4	4	6	9	11	12	16
4	5	5	6	8	10	12	12
4	5	5	6	10	12	14	19
6	6	6	11	12	15	19	28
9	8	10	12	16	20	27	31
11	10	12	15	20	27	31	31
12	12	14	19	27	31	31	31
16	12	19	28	31	31	31	31

Рис. 3.5. Типова матриця квантування

Заключний етап алгоритму кодування стиснення JPEG. Після обробки матриці ДКП за допомогою матриці квантування результатом є велика кількість нулів у вихідній матриці, особливо в області високих частот. (правий нижній кут (рис. 3.6.)).

59	2	-2	0	0	0	0	0
14	0	0	0	0	0	0	0
1	0	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

Рис. 3.6. Порядок обходу матриці зигзагом

Першим кроком є заміна значення у верхньому лівому кутку матриці на відносне значення. Оскільки сусідні блоки зображення схожі один на одного,

кодування наступного елемента (0,0) за рахунок відмінності від попереднього буде більш ефективним. Другим кроком є використання самого алгоритму повторного кодування (LZW) для обробки більшої кількості найближчих нулів. Експериментальні дослідження показали, що кращих результатів можна досягти, переміщаючись по матриці зигзагоподібно, як показано на рис. 3.6. [26].

Дослідження проводили для ділянок зображення з різною яскравою структурою (однорідні, слабо неоднорідні, сильно неоднорідні, ділянки з контрастними контурами). Був встановлений порядок використання певних частот на окремих ділянках зображення - перша, друга і т. д. Це значно підвищило стабільність осадження.

Основним і найбільш ефективним методом підвищення продуктивності будь-якого стеганографічного алгоритму можна вважати впровадження в нього механізму адаптації (надання йому адаптивних властивостей).

Адаптивність системи означає її здатність змінювати свою структуру і функціонування в залежності від середовища (потоку інформації, що надходить). У випадку стеганографічної системи вхідним потоком є стеганографічний контейнер, а адаптивність — здатність однаково працювати з різними типами контейнерів при заданих робочих параметрах.

Розглянемо проблему адаптації більш детально стосовно схеми вбудовування інформації в статичні зображення. У цьому випадку різноманітність вхідного потоку створює набір усіх зображень, які можна використовувати як контейнер для ковдри. Якщо адаптація неможлива, продуктивність системи буде досить низькою. Це пов’язано з тим, що така система не зможе використовувати весь потенціал вбудовування окремого контейнера. Замість цього буде використано лише невелику частину коефіцієнтів області перетворення, що дасть однаково хороші результати для всіх зображень.

При цьому експериментально доведено, що зі збільшенням неоднорідності зображення можливості вбудовування інформації в окремі групи коефіцієнтів можуть збільшуватися в кілька разів. Неадаптивна система не в змозі врахувати такі особливості, оскільки працює з усіма зображеннями однаково (зі зміною розташування неоднорідних ділянок від зображення до зображення). Адаптивна система позбавлена таких недоліків. Головною відмінністю є наявність підсистеми попереднього аналізу. Ця підсистема перевіряє зображення та на основі отриманих результатів змінює алгоритм вбудовування, щоб зробити його максимально ефективним для даного контейнера.

Адаптивність була введена в розроблений алгоритм шляхом попереднього аналізу порожнього контейнера з метою визначення ступеня неоднорідності яскравості та кольору окремих його ділянок. Потім за результатами попереднього аналізу було визначено кількість інформації, що міститься в кожному розділі. Найбільше змін зазнали високогетерогенні блоки, тоді як однорідні частини зображення (де людський зір чутливий навіть до незначних змін) не змінилися або модифікація була мінімальною.

Важливим моментом при створенні адаптивного аналізу є те, що аналіз повинен давати однакові результати як для порожнього, так і для повного контейнера. Дійсно, інакше було б неможливо правильно витягнути вбудоване повідомлення.

Висновки до розділу 3

Проведений адаптивний аналіз – це спосіб збільшення розміру вбудованих повідомлень, який не впливає на стабільність алгоритму. Розробка такого механізму спирається на поглиблені дослідження алгоритму стиснення, формату зберігання та людського зору, щоб використовувати весь потенціал формату приховування повідомлень.

РОЗДІЛ 4. ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ

4.1. Вимоги до програмного та апаратного забезпечення

Програмне забезпечення stegoprotection було розроблено та протестовано в інтегрованому середовищі Delphi. Ніяких додаткових компонентів не використовувалося, тільки стандартні. Це програмне забезпечення працює в операційних системах від Windows XP до Windows 11.

Розглянемо мінімальні вимоги до роботи програмного забезпечення з апаратної точки зору:

- процесор з частотою не менше 2,0 ГГц; Для великих контейнерів рекомендована частота процесора 3,0 ГГц або вище;
- ОЗУ понад 4 ГБ, рекомендовано 16 ГБ;
- жорсткий диск ємністю понад 500 Гб для збереження результатів програми стегозахисту; Для оптимальної продуктивності рекомендується використовувати швидкий SSD накопичувач;
- відеокарта з роздільною здатністю не менше 1366 на 768 пікселів, рекомендована роздільна здатність 1920 на 1080 пікселів.

4.2. Запуск програмного продукту

Розглянемо меню програмного продукту (рис. 4.1).

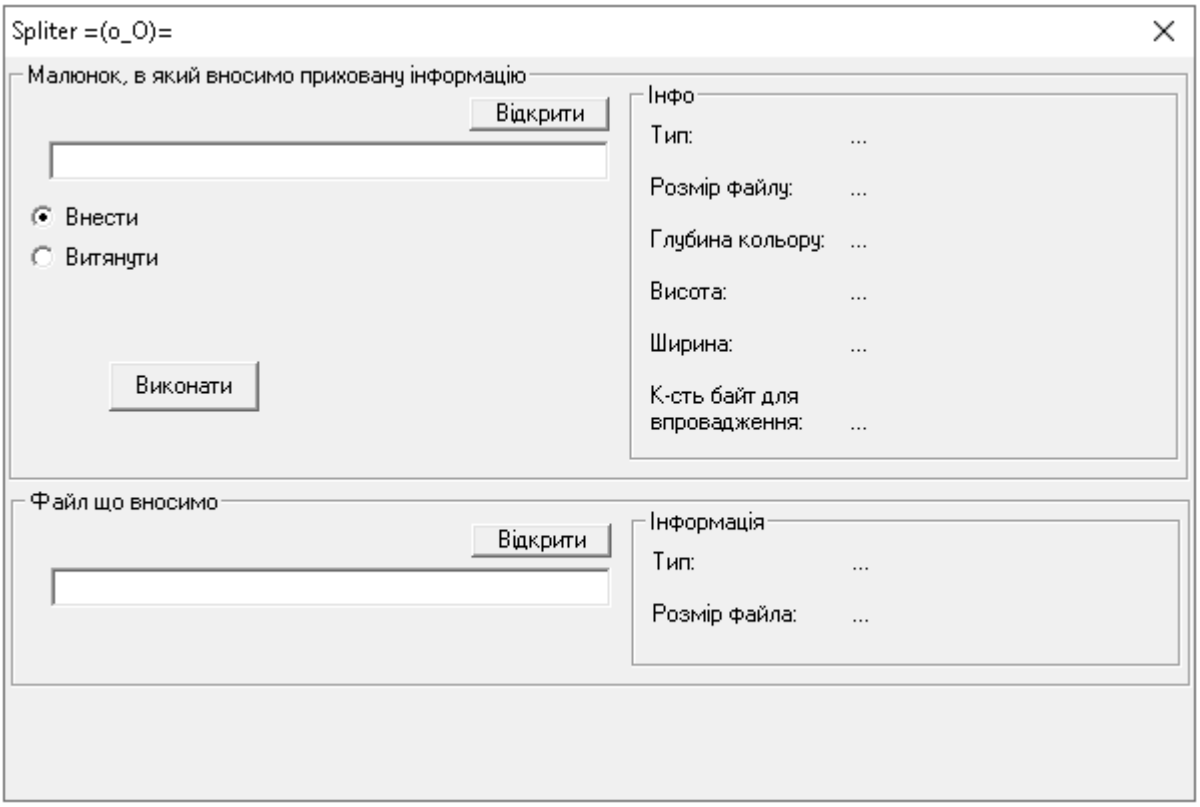

Рис. 4.1. Меню програмного продукту стегозахисту

Після запуску програми перед запуском необхідно встановити стегозахист:

– Шлях до файлу формату JPG, який буде використовуватись як контейнер (рис. 4.2);
– Контейнер для стегозахисту (рис. 4.3);
– Шлях до файлу, який будемо вносити в контейнер для стегозахисту (рис. 4.4);
– Файл, який вноситься в контейнер для стегозахисту (рис. 4.5).
– Поставити мітку «Внести» або «Витянути» (рис. 4.2, 4.6).

Після цього запускається процес стегозахисту. Для цього необхідно натиснути кнопку <Виконати>.

Введемо шлях до файлу 1.jpg формату JPEG, який буде використовуватись як контейнер для стегозахисту (рис. 4.2).

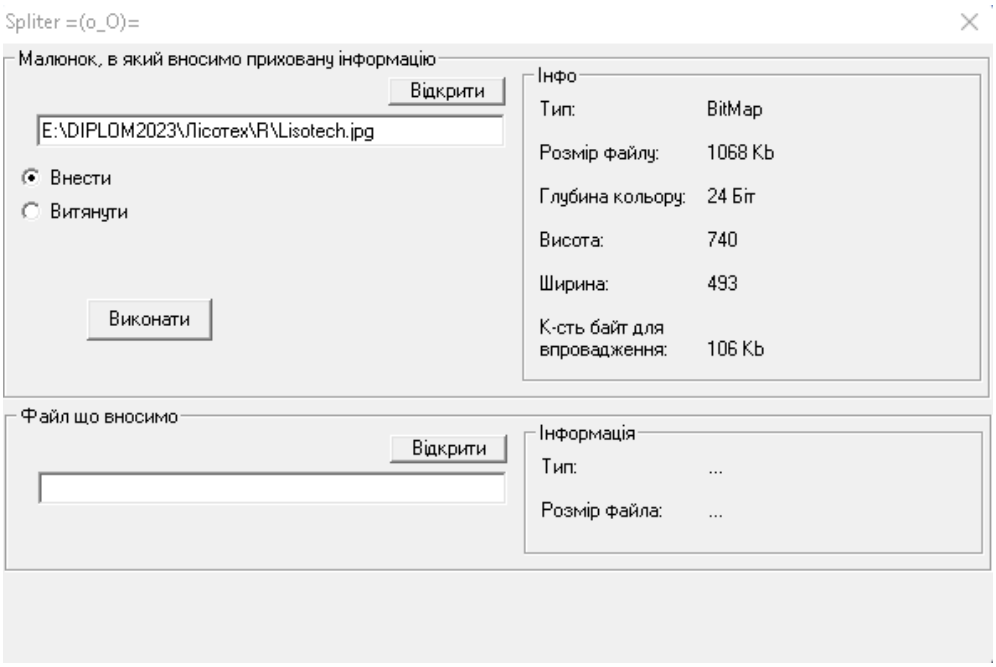

Рис. 4.2. Шлях до файлу Lisotech.jpg формату JPEG, який буде використовуватись як контейнер для стегозахисту

Виберемо контейнер Lisotech.jpg для стегозахисту (рис. 4.3).

Рис. 4.3. Контейнер Lisotech.jpg для стегозахисту

Введемо шлях до файлу Lisotech.txt, який будемо вносити в контейнер Lisotech.jpg для стегозахисту (рис. 4.4).

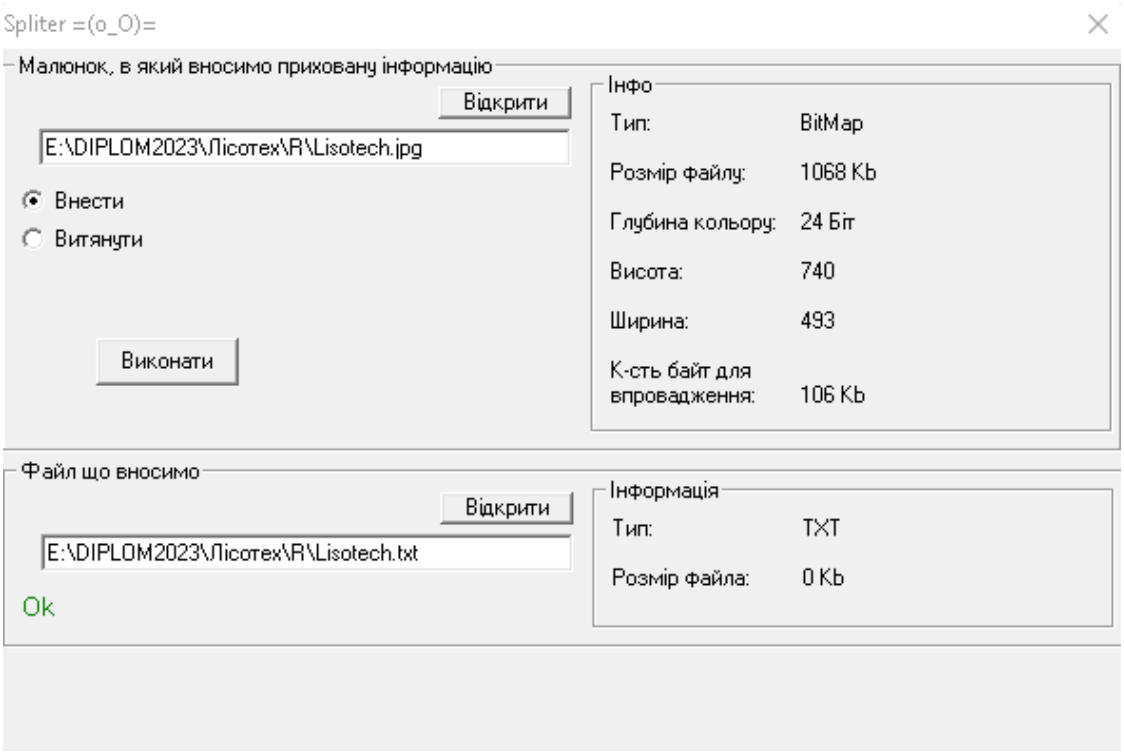

Рис. 4.4. Шлях до файлу Lisotech.txt, який будемо вносити в контейнер 1.jpg для стегозахисту

Виберемо файл Lisotech.txt, який вноситься в контейнер Lisotech.jpg для стегозахисту (рис. 4.5).

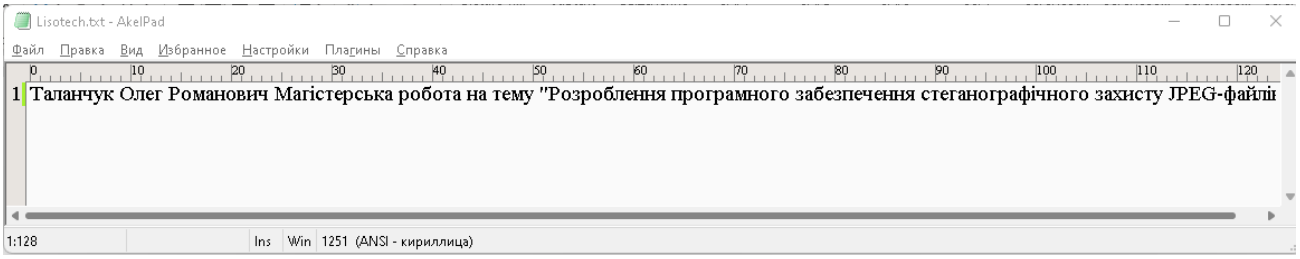

Рис. 4.5. Файл Lisotech.txt, який вноситься в контейнер Lisotech.jpg для стегозахисту

Запустимо процес видобування інформації з контейнеру стегозахисту Lisotech2.jpg (рис. 4.6).

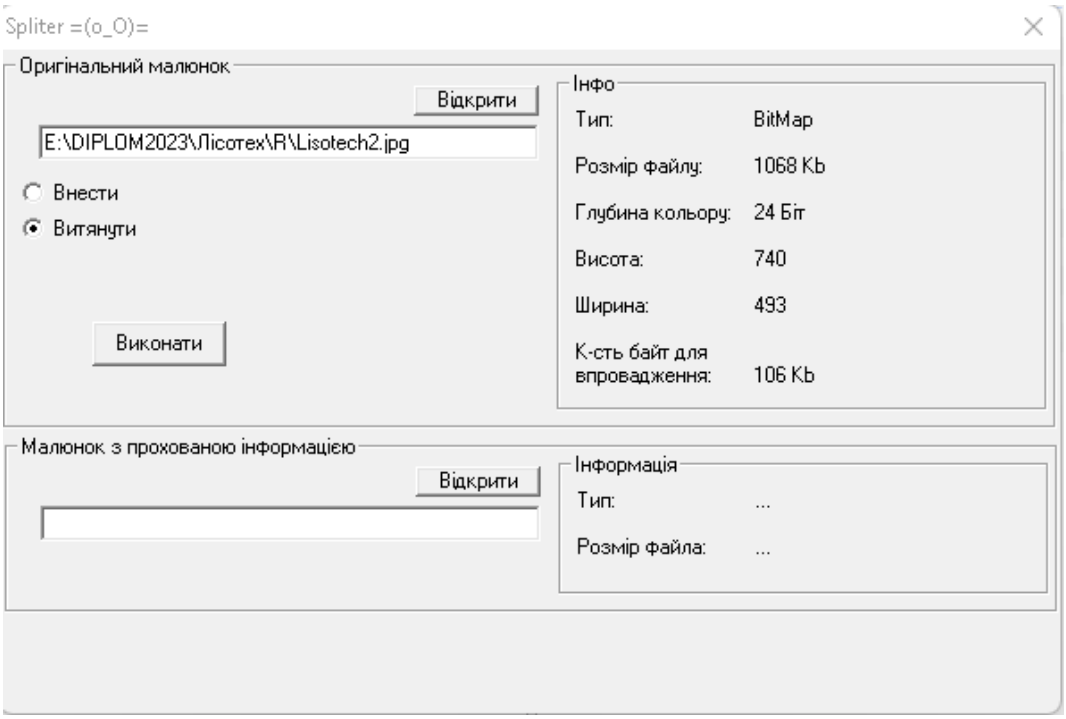

Рис. 4.6. Видобування інформації з контейнеру стегозахисту Lisotech2.jpg

Введемо шлях до файлу Lisotech2.txt, що видобуваємо з контейнеру Lisotech2jpg (рис. 4.7).

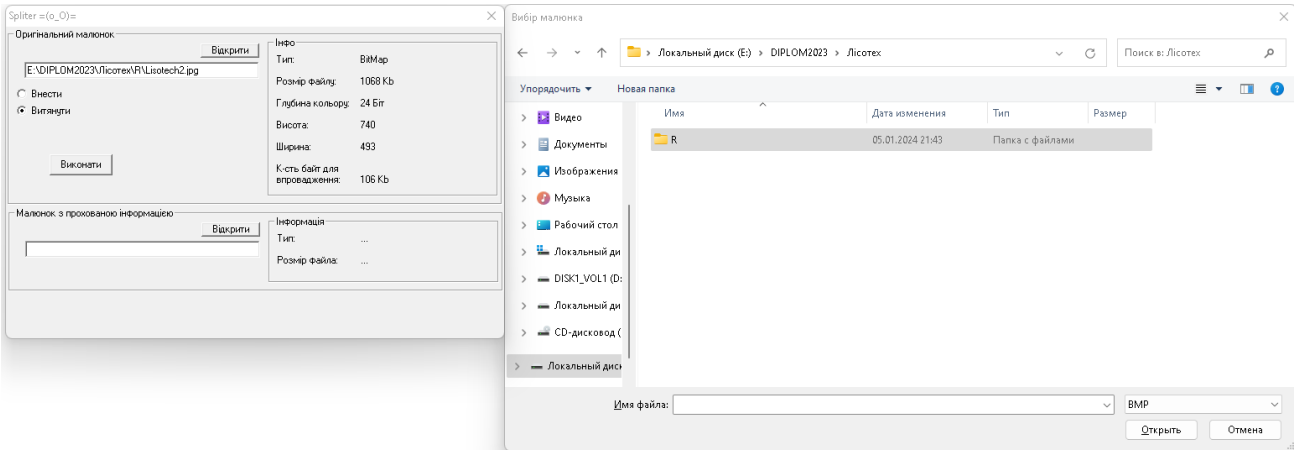

Рис. 4.7. Шлях до файлу Lisotech2.txt, що видобуваємо з контейнеру Lisotech2jpg

Отримаємо результат програмного продукту щодо видобуття інформації у файл Lisotech2.txt з контейнеру стегозахисту Lisotech2.jpg (рис. 4.8).

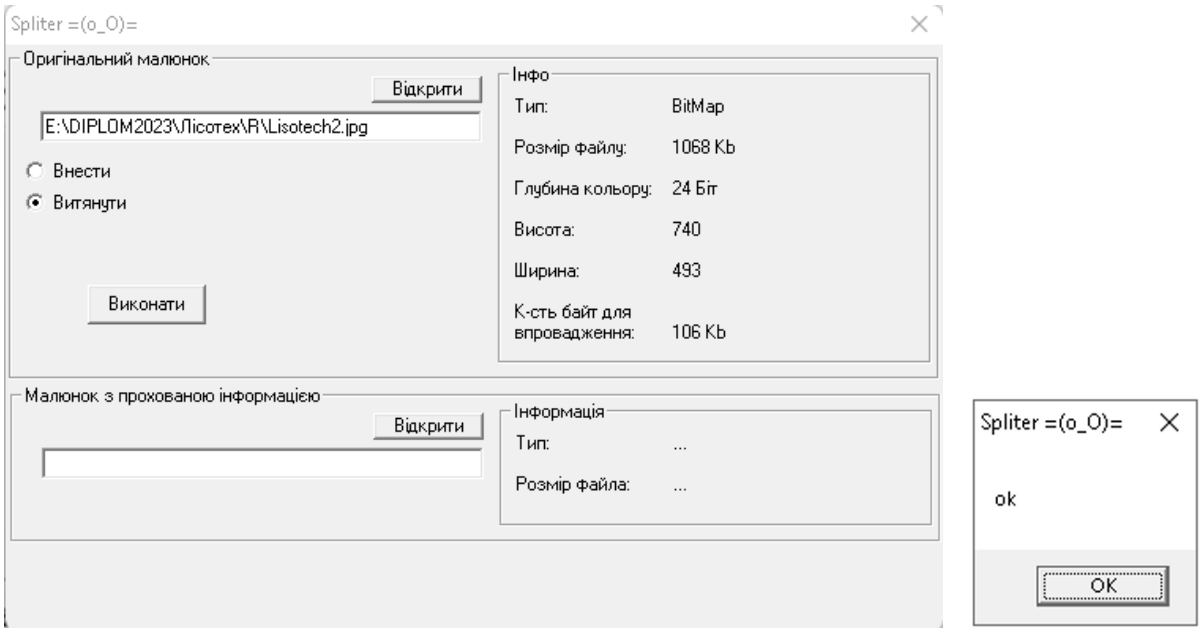

Рис. 4.8. Результат програмного продукту щодо видобуття інформації у файл Lisotech2.txt з контейнеру стегозахисту Lisotech2.jpg

Результат видобування інформації у файл Lisotech2.txt з контейнеру стегозахисту Lisotech2.jpg (рис. 4.9).

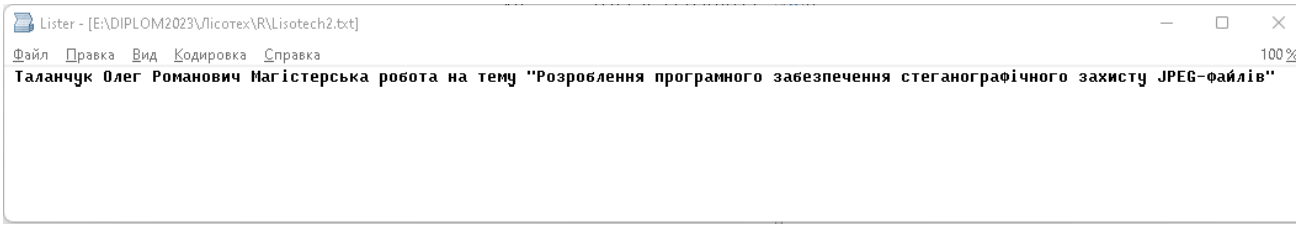

Рис. 4.9. Видобута інформація у файл Lisotech2.txt з контейнеру стегозахисту Lisotech2.jpg

Відображення контейнеру Lisotech2.jpg з файлом захисту Lisotech.txt (рис. 4.10).

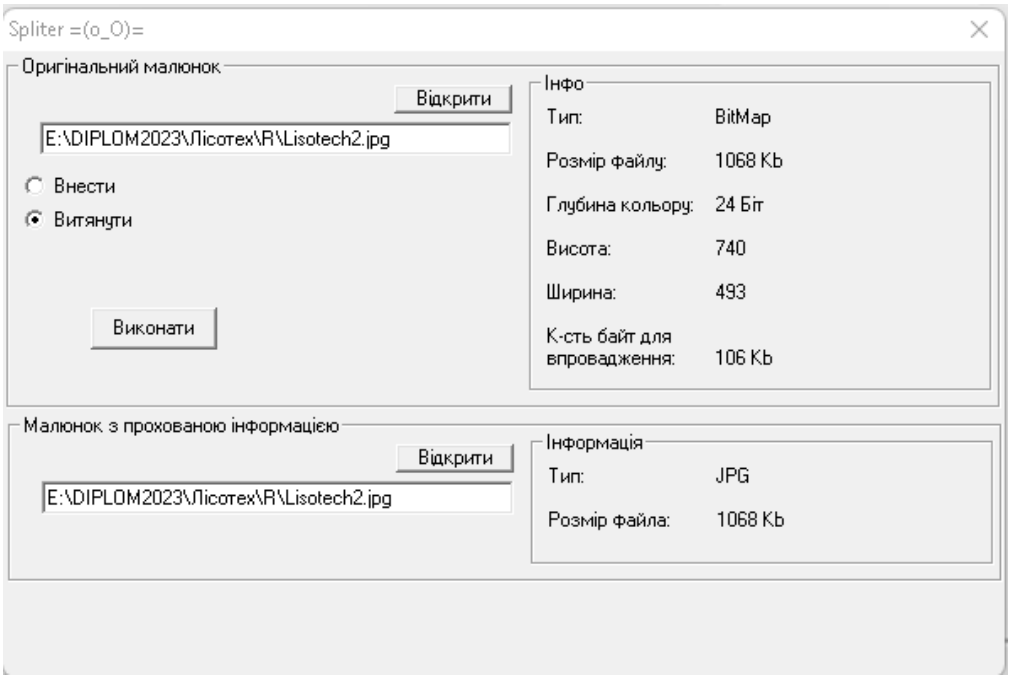

Рис. 4.10. Відображення контейнеру Lisotech2.jpg з файлом захисту Lisotech.txt

Отримаємо відображення контейнеру Lisotech2.jpg з файлом захисту Lisotech.txt (рис. 4.11).

Рис. 4.11. Контейнер Lisotech2.jpg з файлом захисту Lisotech.txt

Як ви бачите на рис. 4.3 і 4.11, зображення візуально не відрізняються. Тому створене програмне забезпечення працює відповідно до передбачуваної мети захисту стего.

Висновки до розділу 4

Розроблений програмний продукт забезпечує стеганографічний захист JPEG-файлів, який не ідентифікується відомими програмами стегодетектування. Візуально та програмно за розміром порожній та повний контейнери JPEG-файлів не відрізняються.

РОЗДІЛ 5. РОЗРОБЛЕННЯ СТАРТАП-ПРОЄКТУ

5.1 Опис ідеї проєкту

Метою даного проєкту є дослідження та реалізація основних методів реалізації цифрових водяних знаків, які передбачають впровадження водяних знаків на основі псевдовипадкових послідовностей через недостатній опис таких методів у відкритих джерелах інформації.

Предметом дослідження є захист авторських прав файлів із використанням водяних знаків.

Предметом дослідження є засоби захисту авторських прав зображень за допомогою водяних знаків.

5.2. Розроблення ринкової стратегії

Серед усього спектру методів захисту даних від небажаного доступу стеганографічні методи займають особливе місце. На відміну від інших методів, вони засновані виключно на властивостях самої інформації та не використовують властивості її матеріальних носіїв, специфіку вузлів її обробки, передачі та зберігання.

Образно кажучи, стеганографічні методи вибудовують бар'єр між інформацією, що захищається, і реальним або потенційним зловмисником із самої інформації. Звичайно, стеганографічний захист полягає в першу чергу в прихованні та шифруванні даних.

Цифрова стеганографія як наука буквально народилася останніми роками. Вона включає наступні основні методи:

- запровадження інформації для секретної передачі;
- вбудовування цифрових водяних знаків;
- внесення ідентифікаційних номерів (відбитків пальців);

– вбудовування титрів (підписів).

5.3. Розроблення маркетингової програми

У цьому проєкті проводимо дослідження методів стеганографії захисту інформації у зображеннях, а зображення в стеганографії — це контейнери, в яких захована інформація.

Розглянемо поняття контейнера докладніше. Перед стегокодером стоїть порожній контейнер, після нього - заповнений контейнер, тобто. стего. Стего має бути візуально невідмінним від порожнього контейнера.

Контейнер потоку є безперервною послідовністю бітів. Повідомлення впроваджується в режимі реального часу, тому кодер заздалегідь не знає, чи достатній розмір контейнера для відправки всього повідомлення. Декілька повідомлень можна помістити в один великий контейнер. Відстань між впровадженими бітами визначається генератором псевдовипадкової послідовності з рівномірним розподілом інтервалу між відліками.

Основна складність - знайти необхідну псевдовипадкову послідовність, реалізувати синхронізацію та визначити початок та кінець послідовності.

Для вбудовування секретного повідомлення в контейнер використовується алгоритм вбудовування інформації, що лежить в основі стегосистеми, основне завдання якого вносити в контейнер зміни, невидимі для людини.

Ця вимога зазвичай обмежує ємність контейнера, яка є максимальним обсягом прихованої інформації, що може зберігати контейнер. Місткість контейнера залежить від характеристик самого контейнера, алгоритму розміщення інформації, інколи ж і від секретного повідомлення.

Спеціальний алгоритм вилучення інформації призначений для перевірки наявності секретного повідомлення усередині контейнера та його вилучення.

5.4. Вимоги до технічного та програмного забезпечення

Перед впровадженням у контейнер для підвищення безпеки та компактності секретне повідомлення стискається та шифрується.

Дані, що містять приховане повідомлення, можуть бути схильні до навмисних атак або випадкового втручання.

Алгоритм впровадження цифрового водяного знаку складається із трьох основних етапів:

– генерація повідомлень;

– вбудовування повідомлення в кодувальник;

– виявлення повідомлення у детекторі;

Молодший біт (LSB) зображення містить найменшу кількість інформації. Відомо, що людина зазвичай не здатна помітити зміну цього біта. Насправді, це шум. Тому його можна використовувати для вбудовування інформації. Так, у випадку з растровим зображенням обсяг впроваджених даних може становити 1/8 обсягу контейнера.

Перевагами цього є його простота і щодо великий обсяг впроваджуваних даних. Однак він має серйозні недоліки.

По-перше, приховане повідомлення легко знищити.

По-друге, не забезпечується таємність впровадження інформації. Порушнику відоме точне місце розташування всього центру.

Для подолання цього останнього недоліку було запропоновано вбудовувати цифровий водяний знак не в усі пікселі зображення, а тільки в деякі з них, що визначаються за псевдовипадковим законом за ключом, відомому тільки авторизованому користувачеві. При цьому знижується пропускна спроможність.

Практична цінність фірмового методу кодування файлів полягає у присвоєння так званого коду авторського права, який захистить файл від несанкціонованого доступу.

Висновки до розділу 5

Представлено можливість стегозахисту файлів графічних зображень на основі методу найменш значущих бітів з використанням псевдовипадкових послідовностей.

Запропонований захист не ідентифікується відомими програмами стегодетектування. Апробований метод дозволяє здійснювати простий стеганографічний захист графічних зображень.

ВИСНОВКИ

1. Представлено рішення для захисту файлів із використанням контейнерів графічних зображень JPEG на основі псевдовипадкових послідовностей.

2. Розроблено ефективний алгоритм стегокодування та стегкодування файлів JPEG.

3. Розроблений захист Stego не ідентифікується відомим програмним забезпеченням виявлення Stego.

СПИСОК ЛІТЕРАТУРИ

1. Барсуков В.С. Компьютерная стеганографія вчора, сьогодні, завтра. Технології інформаційної безпеки 21 століття / В.С. Барсуков, А.П. Романцов ; – К : "Спеціальна Техніка", 2017. – 225 с.
2. Городецький В.І., Самойлов В.І. Стеганографія на основі цифрових зображень. – 2022. (http://www.iias.spb.ua).
3. Грибунин В.Г. Цифрова стеганографія / В.Г. Грибунин, И.Н. Оков, И.В. Туринцев; – К : СОЛОН-Пресс, 2019. – 261 с.
4. Грибунин В.Г., Оков И.Н., Туринцев И.В. Цифрова стеганографія. – К.: Солон-Пресс, 2022. –258 с.
5. Домарев В.В. Захист інформації та безпека комп'ютерних систем. — К.: Видавництво ДіаСофт, 2019. — 480 с.
6. Жельников В . Криптографія від папіруса до комп'ютера / В. Жельников. – К. : АБФ, 2019. – С.12-15. Закордонна радіоелектроніка. Спеціальний випуск. 2019, N2 12.
7. Захист інформації. 2019–2022 рр., №№ 1–4.
8. Захист інформації “Конфідент”. 2005–2008 рр., №№ 1–6.
9. Захист програмного забезпечення. пер. с англ./Д. Гроувер, Р. Сатер, Дж. Фіпс і др.; Під ред. д. Гроувера. — К.: Мир, 2018. — 285 с.
10. Зегжда Д.П., Івашко А.М. Основи безпеки інформаційних систем. — К.: Телеком, 2000. — 452 с.
11. Зегенда Д.П. і др. Захист інформації в комп'ютерних системах/Під ред. проф. Е.М. Шмакова. — К., 2009. — 100 с.
12. Конахович Г . Ф., Пузиренко А. Ю. Комп’ютерна стеганографія. Теорія і практика. – К.: «К-Прес», 2016. – 288 с.An Overview of Steganography for the

Computer Forensics Examiner. 2004 [Електронний ресурс]. – Режим доступу: http://www.garykessler.net/library/fsc_stego.html.

13. Buckland M., Goldberg E. Emanuel Goldberg and His Knowledge Machine. – Libraries Unlimited. – 2016. – 70p.
14. Conway M. Steganography, Signals Intelligence, and Terrorism // Knowledge, Technology and Policy. – 2023. – V.16, No2. – P. 45-47.
15. http://ua.wikipedia.org/wiki/YCbCr
16. http://www.alrise.ua/2021/10/30/skrytie-dannyx
17. Markus G. Kuhn. Compromising emanations: eavesdropping risks of computer displays. Technical Report UCAM-CL-TR-577, University of Cambridge, Computer Laboratory, December 2023.
18. The Third International Conference on Vailability, Reliability and Security A Statistical Algorithm for Linguistic Steganography Detection Based on Distribution of Words. [Електронний ресурс]. – Режим доступу: http://www.giac.unibel.uk
19. Voloshynovskiy S., Pereira S., Iquise V., Pun T. Attack Modeling: Towards a Second Generation Watermarking Benchmark // Preprint. University of Geneva, 2019. 58p.
20. Watson A. The cortex transform: rapid computation of simulated neural images // Computer Vision, Graphics, and Image Processing. 2017. Vol. 39. No 3. P. 311-327.

ДОДАТОК А. ТЕКСТ ПРОГРАМИ СТЕГОЗАХИСТУ ГРАФІЧНИХ ФАЙЛІВ JPEG

```
unit ikb;
interface
uses avl;
type
int = integer;
pint = ^int;
  Tinfo_ikb = record
              c_ikb:int;
              c_comb_ikb:int;
              count_bit_on_a_code:byte;
              elem : array of byte;
              mas: array of array of byte;
             end;
  Tmap_bitmap = record
                pHead:pByte;
                pdata:pByte;
                end;
  Tc_m = record
         str: string;
         n:int;
        end;
//=======================================================
function CetInfo_ikv(file_path: string):boolean;
implementation
var info_ikv:Tinfo_ikb;
    code_m : array of Tc_m;
procedure copy_mem(bin,p:pointer;i:int);assembler;
asm
pushad
mov ecx,i
mov edi,p
mov esi,bin
rep movsb
popad
```

```
fill_str(@code_m[0].str[1],info_ikv.c_ikb,'0');
beg:=0;
for i:=1 to info_ikv.c_comb_ikb do
 begin
  sum:=0;
  if beg>0 then
   begin
    ror(@info_ikv.elem[0],beg,true);
    ror(@code_m[i-1].str[1],beg,true);
   end;
  beg:=0;
  ye:=false;
  SetLength(code_m[i].str,info_ikv.c_ikb);
  fill_str(@code_m[i].str[1],info_ikv.c_ikb,'0');
  for j:=0 to info_ikv.c_ikb-1 do
   if info_ikv.elem[j]=i then
    begin
     code_m[i].str[j+1]:='1';
     ye:=true;
     break;
    end;
  j:=0;
  if not ye then
  while j<info_ikv.c_ikb do
   begin
    sum:=sum+info_ikv.elem[j];
    if sum=i then
     begin
      Fill_Str(@code_m[i].str[1],j+1,'1');
      break;
     end else
    if sum >i then
     begin
      j:=0;
      inc(beg);
      sum:=0;
      ror(@info_ikv.elem[0],1);
```

```
Continue;
        end;
       inc(j);
     end;
  end;
  if beg>0 then
    begin
     ror(@info_ikv.elem[0],beg,true);
     ror(@code_m[i-1].str[1],beg,true);
    end;
end;

function bild_comb: boolean;
begin
end;
function CetInfo_ikv(file_path:string):boolean;
procedure T2M(st:String);
var p1,p2:pChar;
   p3:pbyte;
begin
p1:=@st[1];
p2:=p1;
p3:=@info_ikv.elem[0];
while p1^<> #0 do
  begin
    if (p1^ = ',') or (p1^ = #0) then
     begin
       p1^:=#0;
       p3^:=StrToint(String(p2));
       inc(p3);
       inc(p1);
       p2:=p1;
     end else inc(p1);
  end;
  p3^:=StrToint(String(p2));
end;
var inf:TIniFile;
```

```
  s:string;
  i:int;
begin
inf:=TIniFile.Create(file_path);
info_ikv.c_ikb:=inf.ReadInteger('settings','count_IKB',-1);
if info_ikv.c_ikb = -1 then
 begin
  ShowMessage('Немогу определить с настроек, количество елементов IKB');
  inf.Free;
  exit;
 end;
info_ikv.count_bit_on_a_code:=inf.ReadInteger('settings','count_bit_on_a_code',-1);
if info_ikv.count_bit_on_a_code=-1 then
 begin
  ShowMessage('Немогу определить с настроек, количество бит на один символ');
  inf.Free;
  exit;
 end;
info_ikv.c_comb_ikb:=inf.ReadInteger('settings','count_comb_IKB',-1);
if info_ikv.count_bit_on_a_code=-1 then
 begin
  ShowMessage('Немогу определить с настроек, количество комбинаций IKB');
  inf.Free;
  exit;
 end;
s:=inf.ReadString('Settings','IKB','');
if s='' then
 begin
  ShowMessage('Немогу определить с настроек, последовательность IKB');
  inf.Free;
  exit;
 end;
SetLength(info_ikv.elem,info_ikv.c_ikb+1);
SetLength(code_m,info_ikv.c_comb_ikb+1);
T2M(s);
Find_;
SetLength(info_ikv.mas,info_ikv.c_comb_ikb);
```

```
 for i:=0 to info_ikv.c_comb_ikb-1 do
  SetLength(info_ikv.mas[i],info_ikv.c_ikb);
bild_comb;
inf.Free;
end;
end.
```