Міністерство освіти і науки України

Харківський національний університет радіоелектроніки

Факультет навчально-науковий центр заочної форми навчання
(повна назва)

Кафедра електронних обчислювальних машин
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА

Пояснювальна записка

Рівень вищої освіти другий (магістерський)

Метод нанесення цифрового водяного знаку
на цифрове зображення
(тема)

Виконав:
студент II курсу, групи КСМзм-20-1
Дячкін М.А.
(прізвище, ініціали)

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

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

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

Керівник: проф. Рубан І. В.
(посада, прізвище, ініціали)

Допускається до захисту

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

2021 р.

Харківський національний університет радіоелектроніки

Факультет навчально-науковий центр заочної форми навчання

Кафедра електронних обчислювальних машин

Рівень вищої освіти другий (магістерський)

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

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

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

ЗАТВЕРДЖУЮ:
Зав. кафедри ____________
(підпис)
"____" ______________ 20___ р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

студенту Дячкіну Микиті Андрійовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Метод нанесення цифрового водяного знаку на цифрове зображення

затверджена наказом по університету від " 25 " жовтня 2021 р. № 169Стз

2. Термін подання студентом роботи до екзаменаційної комісії 13 грудня 2020 р.

3. Вхідні дані до роботи Набір цифрових зображень, перелік критеріїв порівняння методів вбудови ЦВЗ

4. Перелік питань, що потрібно опрацювати у роботі
аналіз сфер застосування цифрових водяних знаків; аналіз форматів представлення зображень; огляд сучасних методів нанесення ЦВЗ; розробка програмного комплексу для тестування методів нанесення ЦВЗ

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп’ютерних ілюстрацій (слайдів) 16 слайдів презентації

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

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

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Аналіз сфер застосування цифрових водяних знаків;	09.11.21-12.11.21	
2	Аналіз форматів представлення зображень	13.11.21-18.11.21	
3	Огляд сучасних методів нанесення ЦВЗ	19.11.21-22.11.21	
4	Розробка програмного комплексу	23.11.21-29.11.21	
5	Проведення експериментів	30.11.21-03.12.21	
6	Оформлення матеріалів кваліфікаційної роботи	04.12.21-07.12.21	
7	Подання кваліфікаційної роботи керівникові та	08.12.21-09.12.21	
	її попередній захист		
8	Подання кваліфікаційної роботи на рецензування	10.12.21-11.12.21	

Дата видачі завдання 25 жовтня 2021 р..

Студент ____________________
(підпис)

Керівник роботи ____________________ проф. Рубан І. В.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 86 с., 12 рис., 17 табл., 1 дод., 16 джерел.

ЗОБРАЖЕННЯ, ЦИФРОВИЙ ВОДЯНИЙ ЗНАК, СТЕГАНОГРАФІЯ, АВТОРСЬКІ ПРАВА.

Метою кваліфікаційної роботи є дослідження є порівняння існуючих алгоритмів, виявлення їх переваг та недоліків та обгрунтування вибору ефективного методу нанесення ЦВЗ.

Як критерії порівняння обрані такі показники, як стійкість алгоритмів до атак, час роботи, пропускна здатність стегосистем.

Завданнями дослідження є детальне вивчення предметної галузі, існуючих алгоритмів генерування та вбудовування ЦВЗ у зображення, створення програмного комплексу, що реалізує найбільш популярні алгоритми, детальне порівняння реалізованих алгоритмів за кожним критерієм. При цьому результати будуть представлені у вигляді таблиць та графіків і для кожного з алгоритмів буде визначено його оптимальну сферу застосування.

ABSTRACT

Master’s thesis: 86 pages, 12 figures, 17 tables, XX appendices, 16 sources.

IMAGE, DIGITAL WATERMARK, STEGANOGRAPHY, COPYRIGHT.

The major goal of this thesis is to compare the existing algorithms, identify their advantages and disadvantages and justify the choice of an effective method of applying DW.

Indicators such as algorithm resilience to attacks, runtime, and stego bandwidth capacity were selected as comparison criteria.

The objectives of the study are a detailed study of the subject area, the existing algorithms for generating and embedding DW in the image, creating a software package that implements the most popular algorithms, a detailed comparison of implemented algorithms for each criterion. The results will be presented in the form of tables and graphs and for each of the algorithms will determine its optimal scope.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

ЦВЗ – цифровий водяний знак

СЛЗ – система людського зору

ПВП – псевдовипадкова послідовність

НЧ – сигнал низької частоти

EBCOT – ієрархічне кодування блоків з оптимізованим усіченням

LSB – Least Significant Bit

ВСТУП

Розвиток інформаційних технологій призвів до широкого використання зображень, DVD фільмів, музики у цифровому форматі, у зв’язку з цим стали актуальними питання захисту інформації та авторського права. Увага дослідників була прикута до стеганографії як методу приховування секретної інформації у повідомленні-контейнері. Слово "стеганографія" в перекладі з грецької буквально означає "тайнопис" (steganos - секрет, таємниця; graphy - запис). На відміну від криптографії, де зловмисник може визначити чи є повідомлення, що передається зашифрованим текстом очевидним чином, методи стеганографії дозволяють вбудовувати секретні дані в повідомлення так, щоб неможливо було запідозрити саме існування таємного послання. Приховування повідомлення методами стеганографії значно знижує можливість виявлення самого факту передачі інформації.

В даний час одним із найбільш популярних та перспективних напрямів стеганографії є створення та дослідження алгоритмів генерування та вбудовування цифрових водяних знаків, які найчастіше є основою для систем захисту авторських прав.

Цифрові водяні знаки можуть застосовуватися для захисту від копіювання та несанкціонованого використання інформації. На відміну від звичайних водяних знаків, ЦВЗ зазвичай невидимі для людського ока. Методи мають на увазі вбудовування прихованих маркерів, стійких до різних перетворень контейнера (атакам), з метою збереження цілісності та аутентифікації повідомлення.

Більшість досліджень присвячено використанню як контейнери зображень. Це зумовлено такими причинами:

- існуванням завдання захисту фотографій та цифрових зображень від незаконного тиражування та розповсюдження;
- великим обсягом цифрових зображень, що дозволяє впроваджувати

ЦВЗ великого обсягу чи підвищувати робастність використання;

- відомим до вбудовування та не змінним розміром контейнера, відсутністю обмежень, що накладаються вимогами систем реального часу;
- наявністю у зображеннях текстурних областей, що мають шумову структуру та добре підходять для вбудовування інформації;
- слабкою чутливістю людського ока до певних змін кольорів зображення, його яскравості, спотворень поблизу контурів;
- методами цифрової обробки зображень, що розвиваються.

В рамках кваліфікаційної роботи буде проведено дослідження найпопулярніших алгоритмів генерування та вбудовування цифрових водяних знаків у зображення. Основними цілями даного дослідження є порівняння існуючих алгоритмів, виявлення їх переваг та недоліків. Як критерії порівняння обрані такі показники, як стійкість алгоритмів до атак, час роботи, пропускна здатність стегосистем.

Завданнями дослідження є детальне вивчення предметної галузі, існуючих алгоритмів генерування та вбудовування ЦВЗ у зображення, створення програмного комплексу, що реалізує найбільш популярні алгоритми, детальне порівняння реалізованих алгоритмів за кожним критерієм. При цьому результати будуть представлені у вигляді таблиць та графіків і для кожного з алгоритмів буде визначено його оптимальну сферу застосування.

1 ПРАКТИЧНЕ ЗАСТОСУВАННЯ ТЕХНОЛОГІЇ ЦВЗ

1.1 Захист фотографій від незаконного тиражування

В даний час у зв’язку з розвитком глобальних мереж велику важливість набуває завдання захисту фотографій від незаконного тиражування та розповсюдження. Безліч досліджень присвячено використанню стеганографії як засіб захисту графічної інформації.

За чинним законодавством право на будь-який твір, чи то фотографія, чи будь-яке зображення, належить автору, проте підтвердити авторство в деяких ситуаціях досить важко.

Загальноприйнятою практикою для захисту авторських прав є вбудова знака копірайту ® в контейнер, що захищається. Недоліком цього є можливість видалення знака при обрізанні зображення.

Для усунення цієї проблеми замість знака копірайту можна використовувати невидимі та невіддільні від контейнера без втрати якості зображення цифрові водяні знаки.

Для вирішення проблеми ідентифікації власника компанія Digimarc розробила спеціальну систему, яка здатна визначати автора зображення навіть після внесених спотворень. Ця система була вбудована в популярний графічний редактор Adobe Photoshop, що сприяло її поширенню. Принцип дії системи полягає в наступному: детектор Digimarc розпізнає ЦВЗ, звертається до бази даних, розташованої в Інтернеті та, використовуючи виділений водяний знак як ключ, знаходить дані про автора роботи [1].

До методів цифрової стеганографії при створенні такого роду стегосистем застосовують такі вимоги як робастність (стійкість прихованої інформації до різного роду спотворень), прозорість (відсутність відмінностей між вихідним і модифікованим зображеннями) і стійкість до атак (спроб видалення або спотворення вбудованого повідомлення).

1.2 Захист витворів мистецтва

Можливість використання інтерактивного доступу до творів мистецтва призвела до створення в мережі Інтернет віртуальних музеїв та галерей, більше того, власники баз даних зображень зацікавлені у поширенні зображень.

Але таке поширення породжує потребу у захисті прав на зображення. Фактично потрібно захистити зображення, розміщені в мережі, від несанкціонованого копіювання та правильно визначити власника прав на зображення. Для вирішення цієї проблеми було використано алгоритми нанесення цифрових водяних знаків, які дозволяють включити до електронної репродукції невидимий код без втрати якості зображення.

Наприклад розглянемо простий сценарій, пов’язаний із захистом культурної спадщини за допомогою водяних знаків. Припустимо, залучення нових відвідувачів галерея вирішила розробити веб-сайт, що містить набір електронних репродукцій. Такий сценарій накладає низку вимог на метод нанесення водяних знаків: по-перше, алгоритм має бути індивідуальним, оскільки галерея зацікавлена у безпроблемному виявленні наявності водяного знака на зображенні, яке використане не санкціоновано; ніхто, крім галереї, не повинен мати змоги вилучити вбудований код; водяний знак повинен бути непомітним, щоб користувач мережі Інтернет або пошукова машина могли легко знайти репродукції в мережі, не маючи оригіналу. Нарешті, водяний знак може бути робастним.

Було розроблено систему цифрових водяних знаків, яка б задовольняла цим вимогам. Результати експерименту показали, що водяні знаки стійкі до існуючих способів обробки зображень, включаючи стиснення JPEG, накладання фільтрів: "низького проходу" і "медіани";

Для демонстрації практичного застосування цієї методики водяні знаки були нанесені на зображення, включені до Віртуальної Художньої Галереї області Тоскана, Італія та області Гіфу, Японія (спільна розробка DIE, Флоренція та ManART, Tokishi). Користувач, який переглядає цю галерею в Інтернеті, може перевірити наявність водяного знака на кожному із зображень, просто проставляючи у відповідному полі правильні коди, необхідні для роботи детектора водяних знаків. Якщо введено неправильний код або детектору водяних знаків пред'явлено зображення, яке не входить до цієї Художньої Галереї, відповідь буде негативною [2].

1.3 Вбудовува інтерактивної інформації

Ще однією сферою застосування ЦВЗ є вбудовування інтерактивної інформації у друковану продукцію. Отримана додаткова інформація доповнює текст друкованого видання. Таке зображення називають "розумним" [3].

У цій галузі працює компанія Digimarc. ЦВЗ може вбудовуватись у зображення в журналах, на квитках, обкладинках дисків та книг, пластикових картках, купонах і навіть на упаковках продуктів.

Технологія, реалізована Digimarc, стає можливою завдяки вбудові в область зображення посилань на структури даних, які розташовані в локальних базах даних або в Інтернеті. Ці структури містять інформацію про права власності, створення зображення, контент та інструкції з використання спеціалізованого ПЗ. Ці дані вбудовані в зображення так, що спроба їх примусового видалення веде до руйнування зображення-контейнера. Вбудований ЦВЗ стійкий до афінних перетворень, фільтрації, стиснення, а також до друку та сканування.

Отримане зображення роздруковується на принтері та розміщується на відповідній друкованій продукції. Використовуючи цифрову камеру або спеціальний сканер, спеціальний додаток, попередньо завантажений із сайту компанії користувачем продукції, знаходить та зчитує ЦВЗ. Виявлений ЦВЗ є індексом у базі даних, що розташована в Інтернеті. Цьому індексу відповідає певна URL-адреса. Використовуючи цей URL, веб-браузер виводить відповідну веб-сторінку або запускає програму, створену автором зображення [3]. Отримана додаткова інформація доповнює текст друкованого видання, а зображення з ЦВЗ можна назвати «розумним», що містить як переваги друкованої продукції, так і Інтернет-технологій.

1.4 Автентифікація змісту повідомлення

Редагування цифрових зображень стає все більш простим завданням. При цьому такі модифікації неможливо відстежити.

Іноді виникають ситуації, коли деякі зображення є важливими доказами під час розслідування злочинів. Спотворення таких доказів може призвести до невірного висновку судової експертизи.

У криптографії для автентифікації повідомлень використовується цифровий підпис. Також цифровий підпис використовується для автентифікації зображень. Проблемою в даному випадку є той факт, що цифровий підпис є метаданими, які можуть бути легко втрачені в результаті обробки зображення (наприклад, його стиснення).

Компанія Epson розробила для вирішення цієї проблеми систему, що використовує цифрові водяні знаки, стійкі до різноманітних спотворень зображення. Ця технологія була реалізована у цифрових фотокамерах, випущених компанією [1]. Навіть після незначної модифікації зображення вбудовані камерою мітки не верифікуються.

2.АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

2.1 Механізм вбудовування та виявлення цифрових водяних знаків

Елементами типової стегосистеми є: прекодер – пристрій, призначений для перетворення приховуваного повідомлення до виду, зручному для вбудовування сигнал-контейнер (контейнер - інформаційна послідовність, у яку вбудовується повідомлення); стегокодер - пристрій, призначений для здійснення вбудови прихованого повідомлення в інші дані з урахуванням їхньої моделі; пристрій виділення вбудованого повідомлення; стегодетектор – пристрій, призначений визначення наявності стегособщения; декодер – пристрій, який відновлює приховане повідомлення. Завдання вбудовування та виділення ЦВЗ виконує стеганографічна система з детектором ЦВЗ [4].

Типова схема використання ЦВЗ представлена на рис. 2.1.

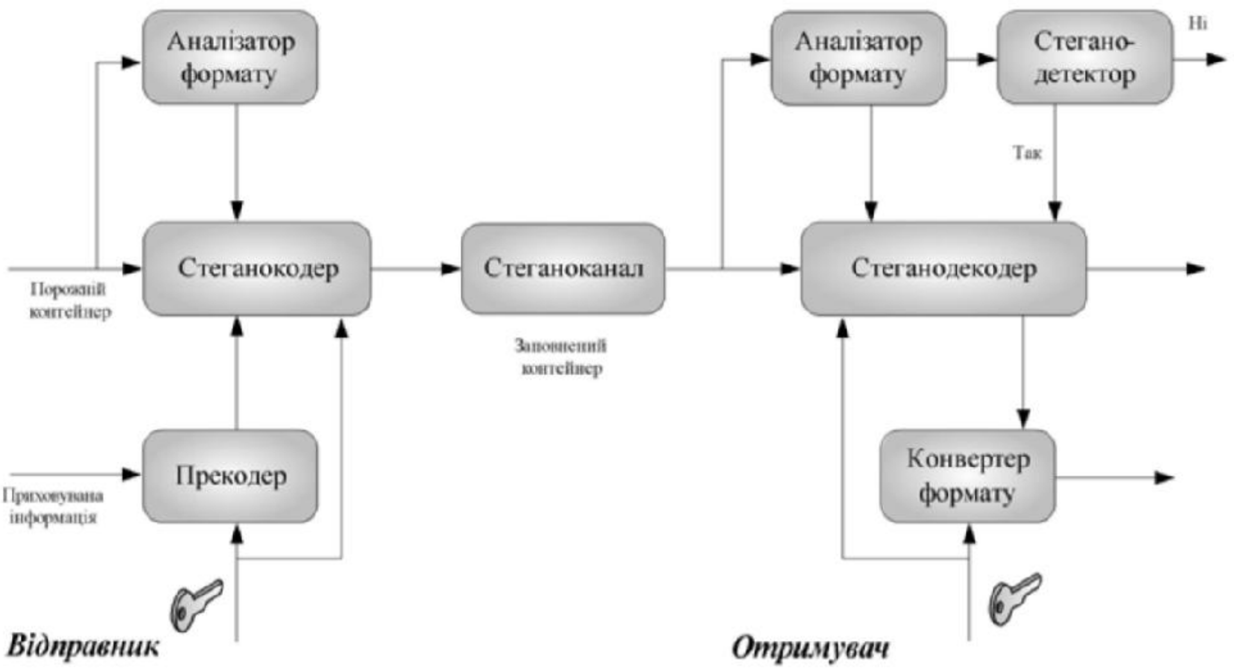

Рисунок 2.1 – Схема типової стегосистеми

Перш, ніж здійснювати вкладення ЦВЗ в контейнер, він має бути перетворений до певного виду. Наприклад, якщо як контейнер виступає зображення, то і вбудована послідовність часто представляється як двовимірний масив біт. Початкову обробку прихованого повідомлення виконує прекодер. Попередня обробка часто виконується з використанням ключа K підвищення секретності вбудови. Далі ЦВЗ «вкладається» у контейнер. Цей процес має враховувати особливості СЛЗ (система людського зору).

У стегодетекторі відбувається виявлення ЦВЗ в захищеному зображенні. Деякі спотворення можуть бути пов’язані з виникненням перешкод у каналі зв’язку, обробкою сигналу та атаками порушників. У багатьох моделях стегосистем сигнал-контейнер сприймається як адитивний шум. Однак такий підхід не враховує двох факторів: невипадкового характеру сигналу контейнера та вимог до збереження його якості. Для побудови ефективних стегосистем необхідно враховувати ці чинники.

Розрізняють стегодетектори, призначені для виявлення факту наявності ЦВЗ та пристрої, призначені для виділення цього ЦВЗ (стедокодери). У першому випадку можливі детектори з жорсткими або м’якими рішеннями. Для винесення рішення про наявність/відсутність ЦВЗ зручно використовувати взаємну кореляцію між наявним сигналом та оригіналом. За відсутності вихідного сигналу у справу вступають інші статистичні методи.

До стегокодера контейнер порожній, після нього заповнений, і називається стего. Стего має бути візуально невідмінним від оригінального контейнера.

Контейнери поділяються на вибрані, нав’язані та випадкові. Вибраний контейнер залежить від вбудованого повідомлення. Нав’язаний контейнер використовується, коли особа, яка надає контейнер, підозрює про можливе приховане листування і бажає запобігти цьому. Насправді ж найчастіше стикаються з випадковим контейнером.

Вбудовування повідомлення в контейнер може здійснюватися ключем, одним або кількома. Під ключом розуміється псевдовипадкова послідовність (ПВП) біт, що породжується генератором, що відповідає певним вимогам [4]. Числа, що породжуються генератором ПВП, можуть визначати позиції модифікованих пікселів у зображенні.

Інформація, що приховується, вбудовується відповідно до ключа в ті області контейнера, зміна яких не призводить до суттєвих спотворень. Ці біти утворюють стегошлях. Залежно від додатка, під суттєвим спотворенням можна розуміти спотворення, що призводить як до неприйнятності для адресата заповненого контейнера, так і можливості виявлення факту наявності прихованого повідомлення в результаті стегоаналізу.

2.2 Класифікація стеганографічних систем на основі цифрових водяних знаків

Розглянемо математичну модель стегосистеми. Алгоритм вбудовування ЦВЗ складається з трьох основних етапів:

- генерації ЦВЗ;
- вбудовування ЦВЗ у кодері;
- виявлення ЦВЗ у детекторі.

За $W^*, \mathrm{K}^*, I^*, B^*$ позначимо множину можливих ЦВЗ, ключів, контейнерів і повідомлень, що приховуються, відповідно. Тоді генерування ЦВЗ може бути представлено в наступному вигляді: $F: I^* \times K^* \times B^* \rightarrow W^*$, $W = F(I, K, B)$, де W, K, I, B – представники відповідних множин. Взагалі, функція F може бути довільною, але на практиці вимоги робастності ЦВЗ накладають на неї певні обмеження. Так, в більшості випадків, $F(I, K, B) \approx F(I + £, K, B)$ тобто незначна зміна контейнера не призводить до зміни ЦВЗ. Функція F зазвичай є складовою: $F = T \circ G$, де $G: K^* \times B^* \rightarrow C^*$ і $T: C^* \times I^* \rightarrow W^*$, тобто цифровий водяний знак залежить від властивостей контейнера.

Оператор Т модифікує кодові слова C^* , внаслідок чого виходить ЦВЗ W^*. На цю функцію не можна накладати обмеження.

Функція Т повинна бути обрана так, щоб вихідний контейнер I_0, заповнений контейнер I_W та незначно модифікований заповнений контейнер $I_W^{'}$ породжували б той самий ЦВЗ: $T(C, I_0) = T(C, I_W) = T(C, I_W^{'})$, тобто вона має бути робастною до малих змін контейнера.

Процес вбудовування ЦВЗ $W(i, j)$ у вихідне зображення $I_0(i, j)$ може бути описаний як суперпозиція двох сигналів: $£ : I^* \times W^* \times L^* \to I_W^*$, $I_W(i, j) = I_0(i, j) \oplus L(i, j) W(i, j) p(i, j)$, де $L(i, j)$ – маска вбудови ЦВЗ, що враховує характеристики СЛЗ, служить зменшення помітності ЦВЗ; $p(i, j)$ – функція проекції, яка залежить від ключа, та позначений $\oplus$ оператор суперпозиції.

Використання функції проекції може розглядатися як реалізація рознесення інформації по паралельних каналах. Крім того, ця функція має певну просторову структуру та кореляційні властивості, що використовуються для протидії геометричним атакам [4].

У стегосистемі головним компонетом є стегодетектор. Залежно від типу може видавати двійкові чи М-мірні рішення про наявність/відсутності ЦВЗ (у разі детектора з м’якими рішеннями).

Розглянемо спочатку простіший випадок «жорсткого» (бінарного) детектора стего. Позначимо операцію детектування через D . Тоді

$$D : I_W^* \times K^* \to \{0,1\},\ D(I_W, W) =$$

$$D(I_W, F(I_{W,} K)) = \begin{cases} 1, \text{ якщо міститься W} \\ 0, \text{якщо відсутннє W} \end{cases} \tag{2.1}$$

Нехай у половини пікселів зображення значення яскравості збільшено на 1, а решта – зменшено на 1. Тоді $I_W = I_0 + W$, де $F(I_0, K) = W$.

Спочатку детектор ЦВЗ обчислює величину

$I_W * W = (I_0 + W) * W = I_0 * W + W * W$. Оскільки W може приймати значення ±1, то $I_0 * W$ буде дуже мало, а $W * W$ буде завжди позитивно. Тому $I_W * W$ буде дуже близько до $W * W$. Тоді можна записати ймовірність невірного виявлення стего, як додаткову функцію помилок від квадратного кореня з відношення $W * W$ («Енергія сигналу») до дисперсії значень пікселів яскравості («Енергія шуму»).

Для випадку м’якого детектора та закритої стегосистеми маємо два основні заходи схожості:

$$¢ = \frac{I_0 I_W}{\| I_0 \| \| I_W \|}$$ – нормований коефіцієнт взаємної кореляції (2.2)

$$¢ = N - \sum_{i=1}^{N} i_0 i_w$$ – відстань по Хеммінгу [4] (2.3)

У детекторі можливе виникнення двох типів помилок. Існує ймовірність того, що детектор не виявить наявний ЦВЗ та вірогідність помилкового знаходження ЦВЗ у порожньому контейнері (імовірність помилкової тривоги). Зниження однієї ймовірності призводить до збільшення іншої. Надійність роботи детектора характеризують ймовірністю хибного виявлення.

Система ЦВЗ повинна бути побудована так, щоб мінімізувати ймовірність виникнення кожної з цих помилок, оскільки будь-яка з них може призвести до відмови від обслуговування.

3.ОГЛЯД ІСНУЮЧИХ АЛГОРИТМІВ ГЕНЕРУВАННЯ ТА ВБУДОВИ ЦИФРОВИХ ВОДЯНИХ ЗНАКІВ У ЗОБРАЖЕННЯ

У цьому розділі будуть розглянуті найбільш відомі алгоритми генерування та вбудови ЦВЗ у зображення. Але перед цим розглянемо деякі параметри, які мають враховуватись даними алгоритмами. До них відносяться особливості системи людського зору та алгоритми стиснення зображень. Нижче будуть розглянуті властивості людського зору, популярні растрові формати та методи стиснення, що використовуються в них.

3.1 Властивості людського зору

Властивості системи людського зору (СЛЗ) можна поділити на низькочастотні та високочастотні. Аж до середини 90-х дослідники брали до уваги, головним чином, низькочастотні властивості зору. Найновіші стегоалгоритми враховують основні високочастотні характеристики СЛЗ.

До найважливіших низькочастотних властивостей відносяться чутливість до зміни яскравості зображення, частотна чутливість і ефект маскування.

Чутливість до зміни яскравості визначається в такий спосіб. Випробуваному показують деяку однотонну картинку. Після того, як око адаптувалося до її освітленості I, поступово змінюють яскравість навколо центральної плями. Зміна освітленості ΔI продовжують доти, доки воно не буде виявлено [6]. Результати досліджень показали, що за малих значеннях яскравості СЛЗ поріг нерозрізненості зменшується.

Частотна чутливість СЛЗ проявляється у більшої сприйнятливості людини до низькочастотного (НЧ) шуму, що пов’язують із нерівномірністю амплітудно-частотної характеристики системи людського зору.

Елементи СЛЗ поділяють сигнал на окремі компоненти. Кожна складова збуджує нервові закінчення ока через ряд підканалів. Виділені оком компоненти мають різні просторові та частотні характеристики. У разі одночасного впливу на око двох компонентів зі подібними характеристиками збуджуються одні й ті самі канали. Це призводить до ефекту маскування, що полягає у збільшенні порога виявлення сигналу в присутності іншого сигналу, що має аналогічні характеристики. Тому адитивний шум набагато помітніше на гладких ділянках зображення, ніж на високочастотних, тобто в останньому випадку спостерігається маскування [4].

Але незважаючи на впровадження нових алгоритмів, високочастотні властивості СЛЗ досі досить рідко враховуються. До таких властивостей відносяться:

- чутливість до контрасту. Висококонтрастні ділянки зображення, перепади яскравості привертають значну увагу.
- чутливість до розміру. Великі ділянки зображення «помітніші» за менші розміри. Причому існує поріг насичення, коли подальше збільшення розміру не суттєво.
- чутливість до кольору. Деякі кольори «помітніші» за інші (особливо, якщо фон заднього плану відрізняється від кольору фігур на ньому).
- чутливість до розташування. Людина схильна насамперед розглядати центр зображення.
- люди уважніші до зображень переднього плану, ніж заднього.
- чутливість до зовнішніх подразників. Рух очей спостерігача залежить від конкретної обстановки, від отриманих перед переглядом або під час нього інструкцій, додаткової інформації.

3.2. Растрові графічні формати

3.2.1 Формат BMP (Bit MaP)

Цей формат є одним із найпростіших способів подання растру. Крім безпосередніх даних щодо пікселів формат BMP фактично нічого не містить.

BMP використовує колірну схему RGB. Максимальні розміри растру по горизонталі та вертикалі – до 2^{16} – 1 пікселів. RGB-значення надаються у безпосередньому вигляді, використання палітри не допускається (у розширених модифікаціях допускається використання 256-колірної палітри). Діапазон значень кожного з каналів – від 0 до 2^{8}-1. Загальна кількість кольорів – 2^{24} (режим TrueColor). Допустимі варіанти кодування кольору з 1, 4, 8, 16 біт на один піксель [5]. Базова версія цього формату не передбачає компресії зображення. Єдиним допустимим режимом виведення зображення є вивод рядків зліва направо, від верхніх рядків до нижніх. Формат BMP забезпечує максимальну якість зображення, тому що в ньому використовується повнокольорова (TrueColor) схема і зображення зберігається без втрат якості, але водночас факт повної відсутності компресії зумовив вкрай низьку ефективність даного формату з точки зору стиснення. Кодування та декодування формату виконується дуже швидко в порівнянні з іншими популярними форматами. Підтримується всіма відомими засобами роботи із графікою. Файли формату BMP мають розширення .BMP; .BM; .RLE. Кодування та декодування формату виконується дуже швидко в порівнянні з іншими популярними форматами. Підтримується всіма відомими засобами роботи із графікою.

3.2.2 Формат PNG (Portable Network Graphics)

Формат PNG розроблявся як засіб компактного представлення «штучної» та «природної» графіки в режимі середньої якості. Основною нішою застосування PNG є представлення графіки в Інтернеті. У PNG використовується колірна схема RGB. Максимальні розміри растру по горизонталі та вертикалі - до 2^{32} - 1 пікселів. RGB-значення можуть бути представлені або у вигляді 256-кольорової палітри (включаючи можливість представлення у вигляді 256 відтінків сірого), або у вигляді повнокольорового режиму TrueColor. Сучасні розширення PNG підтримують до 16 біт на кожен колірний канал, що дозволяє відображати до 2^{48} різних кольорів. При стисканні використовується алгоритм LZ77, який використовується у програмах-архіваторах [5]. Особливістю компресії PNG є наявність стиснення послідовності даних про пікселі як по горизонталі, так і по вертикалі. PNG при використанні повнокольорового режиму забезпечує гарну якість «природних» зображень, але через відсутність режиму стиснення з втратами йому все ж таки складно конкурувати з JPEG за ступенем їх стиснення. Швидкість декодування можна порівняти з відповідними показниками формату BMP, а кодування здійснюється помітно повільніше. Підтримується більшістю сучасних засобів роботи із графікою. Основним недоліком PNG є незручність його використання для дуже малих зображень (кнопок, іконок), оскільки палітра займає в цьому випадку близько 1 Кбайт, що можна порівняти з розміром самого зображення. Файли PNG зазвичай мають розширення .PNG.

3.2.3 Формат TIFF (Tagged Image File Format)

Формат TIFF є одним із найскладніших растрових форматів. Зображення формату TIFF складається із спеціалізованих блоків даних, які називаються тегами, кожен з яких визначається специфічним кодом тега. Теги зберігають інформацію про зображення. У специфікації формату файлів TIFF визначено понад 70 різних типів тегів. Зображення, закодоване у файлі TIFF, повністю визначається його тегами, і цей формат файлу легко розширюється, оскільки для додання додаткових властивостей файлу достатньо лише визначити додаткові типи тегів. У більшості програм для читання файлів TIFF реалізується лише підмножина тегів, саме тому створений однією програмою файл TIFF у ряді випадків не може бути коректно прочитаний іншою. TIFF використовує різні колірні схеми: RGB, LAB, CMYK та ін. Максимальні розміри растру визначаються з наступного міркування: добуток ширини на висоту (загальна кількість пікселів у зображенні) не повинна перевищувати 2^{32}. Значення RGB представляються як у безпосередньому вигляді, так і з використанням палітри. Діапазон значень кожного з колірних каналів – від 0 до 2^8 - 1. Загальна кількість кольорів – 2^{24} (режим TrueColor). Допустимі також варіанти кодування кольору з 1, 2, 4, 8, 16, 32 біт на один піксель. Допускається компресія з використанням ряду алгоритмів (найпоширеніші – RLE та LZW, хоча можуть бути й альтернативні – Deflate, JPEG тощо) [5]. Формат TIFF через свою відкритість та розширюваність є популярним способом представлення растрової графіки. Ефективність кодування та обчислювальна складність кодування/декодування залежить від вибору алгоритму компресії та приблизно збігається з форматами, що базуються на альтернативних алгоритмах компресії. У цьому форматі підтримуються такі професійні можливості, як альфа-канали і можливість зберігати кілька копій зображення з різною роздільною здатністю. Підтримується більшістю засобів роботи із графікою. Файли формату TIFF зазвичай мають розширення. TIF; .TIFF.

3.2.4 Формат JPEG (Joint Photographic Expert Group)

Цей формат, як і TIFF, також є досить складним способом представлення растру і складається з окремих блоків даних, які називаються маркерами, які є аналогами тегів TIFF. JPEG використовує колірну схему RGB. Максимальні розміри растру по горизонталі та вертикалі – до 216 – 1 пікселів. RGB-значення надаються лише у безпосередньому вигляді, використання палітри не допускається. Діапазон значень кожного з колірних каналів – від 0 до 2^8-1. Загальна кількість кольорів –2^{24} (режим TrueColor). Крім повнокольорового режиму 24 біт/піксел інших видів кодування не передбачено, за винятком зображення у відтінках сірого (до 28 відтінків сірого) [5]. Базова версія передбачає компресію із втратами у поєднанні із звичайною компресією. На етапі стиснення із втратами використовується квантування, що використовує дискретне косинусне перетворення, після якого застосовується арифметичне кодування чи кодування Хаффмана. Формат JPEG найчастіше використовується для збереження фотографій. У повнокольоровому режимі забезпечується максимальна ефективність при незначному погіршенні якості зображення, порівняно з оригіналом. Кодування та декодування JPEG проводиться повільніше від усіх інших форматів, що обумовлено високою складністю компресії/декомпресії з втратами. Стандарт стиснення JPEG є найпоширенішим та підтримується всіма відомими засобами роботи з графікою. Додатковою можливістю є збереження в тілі JPEG-файлу зменшеної копії зображення, яка може бути виведена до отримання основної частини файлу. При цьому зменшена копія сама по собі може мати практично будь-який формат (від BMP до JPEG). Стандарт JPEG добре підходить для стиснення зображень у 30-40 разів. При сильнішому стисканні якість різко падає. Файли формату JPEG мають розширення .JPG; .JPEG; .JFIF.

В даний час набуває поширення нова версія формату JPEG – JPEG-2000. Одним із суттєвих покращень JPEG-2000 є застосування більш ефективного алгоритму компресії із втратами – вейвлет-перетворення. Вейвлет-коефіцієнти піддаються квантуванню за алгоритмом, відомим як «ієрархічне кодування блоків з оптимізованим усіченням» (EBCOT). Перевагою EBCOT є те, що він працює з незалежними блоками, що не перекриваються, які кодуються ітеративно. Але дана версія ще не підтримується операційними системами сімейства Windows та популярними веб-браузерами, що значно зменшує її сферу застосування.

Процес застосування прихованої інформації у зображення можна порівняти з процесом їх стиснення. Вбудовування інформації найчастіше здійснюється в незначні області для збереження візуального представлення зображення. Оптимальний метод стиснення видалить цю інформацію, але сучасні алгоритми залишають достатньо можливостей для реалізації різних способів застосування даних. У стеганографії використовується багато ідей в галузі компресії зображень.

3.3 Алгоритми генерування та вбудови цифрових водяних знаків у зображення

За способом вбудови ЦВЗ стегоалгоритми поділяються на адитивні та нелінійні. В алгоритмах адитивного впровадження інформації вихідне зображення лінійно модифікується, а при вилученні декодер використовуються кореляційні методи. При цьому ЦВЗ або складається із зображенням-контейнером, або вплавляється в нього. Нелінійні методи в основному засновані на скалярному та векторному квантуванні.

Серед лінійних алгоритмів можна виділити алгоритми приховування даних у просторовій області та приховування даних у області перетворення. Нелінійні методи можна розділити на алгоритми розширення спектра, скалярного та векторного квантування та методи, що базуються на фрактальному кодуванні.

3.3.1 Метод заміни найменш значущого біта

Метод заміни найменш значущого біта (LSB – Least Significant Bit) – найпоширеніший серед методів заміни у просторовій області.

Молодший значний біт зображення практично несе в собі інформації, тому що людина в більшості випадків не здатна помітити змін у цьому биті. Фактично НЗБ – це шум, тому його можна використовувати для вбудовування інформації шляхом заміни менш значущих бітів пікселів зображення бітами секретного повідомлення. При цьому для зображення в градаціях сірого (кожен піксель зображення кодується одним байтом) обсяг вбудованих даних може становити 1/8 від загального обсягу контейнера.

Популярність даного методу обумовлена його простотою і те, що він має високу пропускну здатність.

Перед прихованням текстового файлу в контейнері його можна захистити криптографічним кодуванням (для цього можна використовувати, наприклад, шифр Віженера) [7].

Для того щоб при розпаковуванні контейнера з отриманої множини символів можна було чітко визначити початок і кінець саме прихованого повідомлення, доцільно ввести відповідні секретні мітки, які обмежували б це повідомлення [7]. Мітки повинні складатися з достатньої кількості символів, щоб не приймати за позначки символи випадкового характеру.

Усі графічні контейнери умовно поділяються на «чисті» та «зашумлені». У перших простежується зв’язок між молодшими та іншими бітами колірних компонентів, і навіть залежність між молодшими бітами. Вбудова повідомлення в чисте зображення руйнує існуючі залежності, що досить легко виявляється. Якщо зображення спочатку зашумлено (скановане зображення, цифрове фото тощо.), то визначити стороннє вкладення стає набагато складніше, хоча й можливе під час використання теорії ймовірностей і математичної статистики.

У цій атестаційній роботі було реалізовано модифікація даного алгоритму, у якій біти повідомлення вносяться в синю компоненту кожного другого пікселя зображення, що підвищує скритність повідомлення.

3.3.2 Метод Куттера-Джордана-Боссена

Куттер (M. Kutter), Джордан (F. Jordan) та Боссен (F. Bossen) [8] запропонували алгоритм вбудовування в канал синього кольору зображення, що є в RGB-кодуванні, так як до синього кольору СЧЗ є найменш чутливою. Розглянемо алгоритм передачі біта секретної інформації у цьому методі.

Нехай M_i - біт, який підлягає вбудові, $C = \{R, G, B\}$ - зображення-контейнер, $p(x, y)$ - псевдовипадковий піксель контейнера, в який виконується вбудова.

Секретний біт M_i вбудовується у канал синього кольору шляхом модифікації яскравості $ª_{x,y} = 0,2989 * R_{x,y} + 0,58662 * G_{x,y} + 0,11448 * B_{x,y}$:

$$B_{x,y}' = \begin{cases} B_{x,y} - ³ * ª_{x,y}, \text{ при } m_i = 0 \\ B_{x,y} + ³ * ª_{x,y}, \text{ при } m_i = 1 \end{cases} = B_{x,y} + (2 * m_i - 1) * ³ * ª_{x,y}, \qquad (3.1)$$

де ³ - Константа, що визначає енергію вбудованого сигналу, яка залежить від призначення стегосистеми. Чим більше ³ , тим вище стійкість вбудованої інформації до спотворень, проте тим сильніше її помітність.

Одержувач отримує біт, не маючи первинного зображення. Для цього виконується передбачення значення не модифікованого пікселя на основі значень сусідніх пікселів. Для оцінки пікселя запропоновано використовувати значення сусідніх пікселів. Автори використовували "хрест" пікселів розміром 7x7. Оцінка $\hat{B}^*_{x,y}$ має наступний вигляд:

$$\hat{B}^*_{x,y} = \frac{1}{4 * ±}\left[\sum_{i=-±}^{+±} B^*_{x+i,y} + \sum_{j=-±}^{+±} B^*_{x,y+j} - 2 * B^*_{x,y}\right], \qquad (3.2)$$

де ± — кількість пікселів зверху (знизу, ліворуч, праворуч) від пікселя, що оцінюється (у вигляді хреста 7x7 ± = 3).

При вилученні вбудованого біта обчислюється різниця ¢ між поточним ($B^*_{x,y}$ і передбачуваним $\hat{B}^*_{x,y}$ значеннями інтенсивності пікселя $p(x, y)$.

Знак ¢ означатиме вбудований біт: якщо ¢ < 0, то $M_i = 0$; якщо ¢ > 0, то M_i = 1.

Функції вбудови та вилучення в даному методі не симетричні, тобто функція вилучення не є зворотною функцією вбудовування. Хоча правильне розпізнання біта повідомлення далеко не стовідсоткове. Для зменшення ймовірності помилок вилучення у процесі вбудови кожен біт повторюється кілька разів (багаторазове вбудовування). Оскільки при цьому кожен біт був убудований ² раз, то виходить ² оцінки одного біта повідомлення. Біт повідомлення витягується як середня різниця реального та оціненого значення інтенсивності пікселя в отриманому контейнері:

$$\text{¢} = \text{²}^{-1}\sum_{i=1}^{\text{²}}\left[B^{*}_{x,y} - \hat{B}^{*}_{x,y}\right] . \tag{3.3}$$

Знак цієї усередненої різниці ¢ та визначає значення вбудованого біта.

Цей алгоритм було реалізовано у програмному комплексі атестаційної роботи з параметрами $\text{³} = 0{,}05$ і ² =200. Ці параметри були вибрані для ефективної роботи алгоритму з усіма зображеннями, але для деяких зображень кількість повторень ² може бути зменшено.

3.3.3 Алгоритм PatchWork

В основі алгоритму Patchwork лежить статистичний підхід [9]. Опишемо механізм роботи алгоритму.

Псевдовипадковим чином на основі ключа вибираються два пікселі зображення. Потім значення яскравості одного з них збільшується на деяке значення (від 1 до 5), значення яскравості іншого зменшується на те ж значення. Далі цей процес повторюється велике число разів (~10000) і є сума значень всіх різниць. За значенням цієї суми судять про наявність чи відсутність ЦВЗ у зображенні.

Введемо ряд позначень. Нехай значення пікселів, що вибираються на кожному кроці a_i і b_i , величина збільшення - δ. Тоді сума різниць значень пікселів буде мати наступний вигляд:

$$S_n = \sum_{i=1}^{n}[(a_i + \text{¢}) - (b_i - \text{¢})] = 2\text{¢}\, n + \sum_{i=1}^{n}(a_i - b_i) . \tag{3.4}$$

Математичне очікування величини $\sum_{i=1}^{n}(a_i - b_i)$ близько до нуля при досить великому n. Математичне очікування величини S_n буде більше 2¢. Таким чином, у стегодетекторі відповідно до ключа перевіряється значення S_n і в тому випадку, якщо значення значно відрізняється від нуля, виноситься рішення про наявність ЦВЗ [4].

Авторами було запропоновано поліпшення алгоритму підвищення його робастности. Замість окремих пікселів пропонується використовувати блоки (так звані patches, звідки і отримав назву алгоритм). Так як однією з найпоширеніших атак є стиск JPEG, то доцільно, щоб спектр ЦВЗ знаходився в області низьких частот. З іншого боку, якщо характер можливих модифікацій стего наперед невідомий, доцільне застосування сигналів із розширеним спектром. Від форми блоку залежить невидимість спотворень, що вносяться.

Алгоритм Patchwork є робастним до операцій стиснення зображення, його усічення, зміни контрастності. Основним недоліком алгоритму є його нестійкість до афінних перетворень, тобто поворотів, зрушення і масштабування. Також стегосистеми, засновані на застосуванні даного алгоритму, мають малу пропускну здатність. Так було в базової версії алгоритму передачі біта повідомлення потрібно 20000 пікселів [4].

3.3.4 Метод відносної заміни величин коефіцієнтів ДКП (метод Коха та Жао)

У більшості методів приховування даних зображення використовується, як декомпозиція контейнера. Серед усіх лінійних ортогональних перетворень найбільшого поширення у стеганографії набули вейвлет-перетворення та ДКП, що здебільшого пояснюється їх успішним застосуванням при стисканні зображень. Крім того, бажано застосовувати для приховання даних перетворення зображення, яке використовується при його стисненні, що сприятиме робастності до компресії зображення. У стандарті JPEG використовується ДКП, а JPEG2000 – вейвлет-перетворення.

Розглянемо схему роботи стегосистеми, що здійснює вкладення ЦВЗ у область перетворення [4]. Зображення поділяється на субсмуги. До кожної з них додається прихована інформація. В результаті зворотного перетворення виходить модифіковане зображення-контейнер. Після проведення компресії/декомпресії виходить зображення. Воно піддається прямому перетворенню, і з кожної субсмуги незалежно виймається приховане повідомлення.

Ефективність застосування вейвлет-перетворення та ДКП для стиснення зображень пояснюється тим, що вони добре моделюють процес обробки зображення у СЛЗ, відокремлюючи «значні» деталі від «незначних» [4]. Їх рекомендовано застосовувати у разі присутності в каналі передачі активного порушника. Дійсно, модифікація значних коефіцієнтів перетворень призведе до великого спотворення зображення.

Найбільш поширеним методом приховування інформації у частотній області зображення є метод відносної заміни величин коефіцієнтів ДКП, який свого часу описали Кох (E. Koch) та Жао (J. Zhao) [10].

Дискретне косинусне перетворення має вигляд: (3.5) – для прямого перетворення, (3.6) – для зворотного.

$$\Omega(³,¬) = \frac{¸(³)*¸(¬)}{\sqrt{2N}} * \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} C(x,y) * \cos\left[\frac{® * ³ * (2x+1)}{2N}\right] * \cos\left[\frac{® * ¬ * (2y+1)}{2N}\right], \quad (3.5)$$

$$S(\mathrm{x},\mathrm{y}) = \frac{1}{\sqrt{2N}} * \sum_{³=0}^{N-1}\sum_{¬=0}^{N-1} ¸(³)¸(¬) * \Omega(³,¬) * \cos\left[\frac{®³ * (2x+1)}{2N}\right] * \cos\left[\frac{®¬ * (2y+1)}{2N}\right] \quad (3.6)$$

Спочатку первинне зображення розбивається на блоки розмірністю 8х8 пікселів. ДКП застосовується до кожного блоку – формула (3.5), внаслідок чого виходять матриці розміром 8х8 коефіцієнтів ДКП, які найчастіше позначають $\Omega_b(³,\neg)$ де b – номер блоку контейнера C, а $(³,\neg)$ – позиція коефіцієнта у цьому блоці. Кожен блок призначений для приховування одного біта даних.

Розглянемо механізм роботи алгоритму Коха-Жао [10].

Під час організації секретного каналу кодеру та декодеру має бути відомо про два конкретні обрані коефіцієнти ДКП з кожного блоку, які будуть використовуватися для приховування даних. Задамо дані коефіцієнти їх координатами в масивах коефіцієнтів ДКП: $(³_1,\neg_1)$ і $(³_2,\neg_2)$. Крім цього, зазначені коефіцієнти повинні відповідати косинус-функціям із середніми частотами, що забезпечить приховування інформації у суттєвих для СЛЗ областях сигналу.

Сам процес приховування починається з випадкового вибору блоку C_b зображення, призначеного для кодування біта повідомлення з індексом b. Вбудова інформації здійснюється таким чином: для передачі біта "0" різницю абсолютних значень коефіцієнтів ДКП штучно роблять більшою за деяку позитивну величину, а для передачі біта "1" ця різниця робиться меншою порівняно з деякою негативною величиною:

$$|\Omega_b(³_1,\neg_1)|-|\Omega_b(³_2,\neg_2)|>P,\ \text{при}\ m_b=0; \qquad (3.7)$$

$$|\Omega_b(³_1,\neg_1)|-|\Omega_b(³_2,\neg_2)|<-P,\ \text{при}\ m_b=1; \qquad (3.8)$$

Ці зміни вносяться в коефіцієнти ДКП тільки, якщо їх різниця не відповідає біту, що приховується. Чим більше значення P, тим більш стеганосистема створена з урахуванням даного алгоритму є робастной до стиснення, проте якість зображення погіршується.

Після внесення змін до значень коефіцієнтів, які повинні задовольняти нерівності (3.7-3.8), проводиться зворотне ДКП.

Для отримання даних у декодері виконується аналогічна процедура вибору коефіцієнтів, а рішення про переданий біт приймається відповідно до наступного правила:

$$m_b^* = 0,\ \text{при}\ |\Omega_b^*(³_1,¬_1)| > |\Omega_b^*(³_2,¬_2)|; \quad (3.9)$$

$$m_b^* = 1,\ \text{при}\ |\Omega_b^*(³_1,¬_1)| < |\Omega_b^*(³_2,¬_2)|; \quad (3.10)$$

Враховуючи високорівневі особливості СЛЗ, секретне повідомлення вбудовується у канал синього кольору.

3.3.5 Алгоритм Барні (Barni)

У даному алгоритмі ЦВЗ є послідовністю бінарних псевдовипадкових чисел $w_i \in \{-1,1\}$. Довжина послідовності визначається розмірами вихідного зображення M та N, де:

$$i = 0,...,3 \times \frac{M}{2} \times \frac{N}{2} - 1 \quad (3.11)$$

При вбудовуванні інформації виконується чотирирівневе $(l = 4)$ вейвлет-перетворення з використанням фільтрів Добеші-6. До того ж для впровадження цифрового водяного знака використовуються детальні піддіапазони першого рівня розкладання. При цьому для модифікації вибираються коефіцієнти детальних піддіапазонів (LH, HL, HH), які змінюються з урахуванням локальної чутливості до шумів:

$$f'(m,n) = f(m,n) + \sim\ (m,n) w_i,$$
$$(m,n) = {}^{-}(l,\pm) \times A(l,m,n) \times \Xi(l,m,n). \quad (3.12)$$

Множник $\bar{}(l,\pm)$ у цьому виразі визначається піддіапазоном та рівнем роздільної здатності:

$$
\bar{}(l,\pm)=\begin{cases}\sqrt{2},\pm\in HH\\ 1,\pm\notin HH\end{cases}\times\begin{cases}1.00 & l=1\\ 0.32 & l=2\\ 0.16 & l=3\\ 0.1 & l=4\end{cases}, \tag{3.13}
$$

другий же множник визначається величиною, яка називається локальною яскравістю $A(l,m,n)$:

$$
A(l,m,n)=\frac{1}{256}f_4^{LL}\left(\frac{m}{2^{4l}}\times\frac{n}{2^{4l}}\right), \tag{3.14}
$$

а останній множник $\Xi(n,m,l)$ визначається локальною дисперсією [4].

У детекторі водяний знак виявляється при обчисленні значення кореляції w_i з коефіцієнтами вейвлет-перетворення (ВП)

Кожне бінарне значення водяного знака попередньо множиться на ваговий коефіцієнт, отриманий з урахуванням моделі чутливості СЛЗ до шуму. Це дозволяє досягти непомітності вбудовування.

3.3.6 Метод розширення спектра (метод Сміта-Коміски)

Спочатку методи розширення спектра (РС або SS Spread-Spectrum) використовувалися при розробці систем управління і зв’язку [7]. Останнім часом інтенсивний розвиток даних методів пояснюється бажанням створити ефективні системи зв’язку, що забезпечують високу стійкість до перешкод при передачі сигналу по зашумленному каналу.

При розширенні спектра прямою послідовністю (РСПП) сигнал модулюється функцією, що приймає псевдовипадкові значення у встановлених межах, і множиться на тимчасову константу - швидкість проходження елементів сигналу. Даний сигнал містить складові всіх частот, які, при їх розширенні, модулюють енергію сигналу в широкому діапазоні.

Найпопулярнішим спектральним методом, спрямованим на вбудову інформації у зображення, є метод РСПП, авторами якого є Сміт (JR Smith) та Коміскі (B.O. Comiskey) [11]. Алгоритм модуляції наступний: кожен біт повідомлення m_i є деякою базисною функцією φ_i розмірністю $X \times Y$, помноженої, залежно від значення біта (1 або 0), на +1 або -1:

$$E(x,y) = \sum_i m_i * \varphi_i(x,y). \tag{3.15}$$

Модульоване повідомлення E(x,y) попіксельно підсумовується із зображенням-контейнером C(x,y). Результатом є зображення, що містить S(x, y)= C(x, y) + E(x, y), при $x = 1..X$, $y = 1..Y$.

Щоб домогтися відсутності спотворень вже вбудованого біта повідомлення, базисні функції мають бути ортогональними:

$$\left\langle \varphi_i, \varphi_j \right\rangle = \sum_{x,y}^{X,Y} \varphi_i(x,y) * \varphi_j(x,y) = n_{\varphi} * G^2 * \delta_{i,j}, \tag{3.16}$$

де n_{φ} – кількість значущих пікселів у базовій функції; G^2 – середня потужність, що припадає на піксель;

$$\delta_{i,j} = \begin{cases} 1, \text{ при } i = j \\ 0, \text{ при } i \neq j \end{cases} \text{-дельта символ Кронекера.} \tag{3.17}$$

В ідеалі всі базові функції φ_i повинні бути ортогональними до зображення-контейнера: $\left\langle \tilde{N}, \varphi_i \right\rangle = 0, \forall i$. Однак насправді практично неможливо підібрати контейнер, який був би повністю ортогональним до всіх базисних функцій. $¨_i$. У такому разі має бути введена величина похибки $\left\langle \tilde{N}, ¨_i \right\rangle = \Delta \approx 0$, яка враховується збільшенням потужності G^2 .

Для того, щоб ефективно приховати інформацію, необхідно використовувати велику кількість базисних функцій, ортогональних до типових зображень. Кодування зображень висуває прямо протилежну вимогу: ідеальним вважається невелика кількість базисних функцій, які перекривають всю область зображення. Ці вимоги входять у конфлікт, коли стего піддається стиску: ідеальна схема компресії неспроможна повністю відобразити базиси, які використовувалися приховування.

При РСПП модулююча функція складається з постійного коефіцієнта посилення G (ціле число), помноженого на псевдовипадковий масив базисних функцій $¨_i$ значень ±1 [11]. Кожен масив $¨_i$ має індивідуальне розташування (x, y) -масиві. Крім того, масиви $¨_i$ є взаємно ортогональним і перекривають весь (x, y) - масив без проміжків.

Також вважатимемо, що всі базисні функції мають однакову кількість значущих елементів $(n_{..})$. Повна потужність виражається формулою:

$$P = \sum_{x,y}^{X,Y} \left(\sum_i G * m_i * ¨_i(x,y) \right)^2 = \sum_i \sum_{x,y}^{X,Y} \left(G * m_i * ¨_i(x,y) \right)^2 = G^2 * X * Y = N_{..} * n_{..} * G^2. \qquad (3.18)$$

На етапі отримання даних необов’язково володіти інформацією про вихідний контейнер C. Операція декодування полягає у відновленні прихованого повідомлення шляхом проектування отриманого зображення S на всі базисні функції $¨_i$ $:\pm_i = \left\langle S^*, ¨_i \right\rangle = m_i * n_{..} * G^2$.

Значення біт повідомлення m_i можуть бути легко відновлені за допомогою знакової функції:

$$m_i^* = sign(\pm_i) = \begin{cases} -1 \text{ при } \pm_i < 0; \\ 1, \text{ при } \pm_i > 0; \text{ при условии, что } G^2 >> 0. \\ ? \text{ при } \pm_i = 0, \end{cases} \quad (3.19)$$

При значенні $\pm_i$ =0 прихована інформація втрачається.

Даний алгоритм був реалізований у програмному комплексі та було проведено дослідження його візуальних характеристик вбудовування, стійкості до атак, пропускної спроможності.

3.3.7 Метод Хсу-Ву

У алгоритмі Хсу [12] у ролі ЦВЗ виступає бінарне зображення розміром удвічі менше вихідного. Обидва зображення попередньо розкладаються: контейнер декомпозується за допомогою вейвлет-перетворення (фільтр Добеші-6, два рівні), а ЦВЗ перетворюється за допомогою знижувальної роздільної здатності функції, описаної в стандарті JBIG (Joint Binary Image Group). ЦВЗ із зменшеним дозволом називається залишковим. Залишковий ЦВЗ інтерполюється (тобто між усіма пікселами вставляють нулі) і віднімається з початкового ЦВЗ. В результаті віднімання виходить різницевий ЦВЗ, що має малу енергію.

І різницевий, і залишковий ЦВЗ вбудовуються в вейвлет-коефіцієнти вихідного зображення. При цьому впровадження здійснюється у ВЧ(високочастотні)-НЧ(низькочастотні) та НЧ-ВЧ області. Для підвищення робастності використання ЦВЗ виконується «через стовпець» у кожну з областей: до однієї впроваджуються парні стовпці, а іншу – непарні [4].

Зазначено, що цей алгоритм не є стійким до стиснення та фільтрації зображення: оскільки вейвлет-перетворення концентрує енергію зображення в НЧ-областях, ВЧ-коефіцієнти будуть малі. Тому алгоритм стиснення видалить їх разом із вбудованою інформацією. Іншим недоліком алгоритму є обов’язкова наявність вихідного контейнера декодері.

3.3.8 Метод Чена

Алгоритм Че [13] є найпоширенішим алгоритмом векторного квантування. В даному алгоритмі ЦВЗ є послідовністю символів логотипу, розмір якого в чотири рази менше розмірів контейнера.

n коефіцієнтів вейвлет-перетворення групуються в n-мірний вектор Для кожного коефіцієнта логотипу здійснюється маніпуляції над вектором квантованих коефіцієнтів зображення-контейнера.

Вектор коефіцієнтів ДВП контейнера v_i модифікується відповідно до масштабованого кодового слова, що представляє w_i :

$$v' = v + \sim C(w_i) \tag{3.20}$$

Таким чином, при n = 4 для вбудовування одного коефіцієнта логотипу змінюються чотири коефіцієнти контейнера.

Для отримання інформації потрібна наявність вихідного зображення-контейнера. Вектор помилки обчислюється за формулою

$$e = \frac{v^* - v}{\sim} \tag{3.21}$$

і для відновлення вбудованого біта за кодовою книгою шукається найближче кодове слово

$$w_i = \min_{w_i} \| C(w_i - e) \| \tag{3.22}$$

При впорядкованості кодової книги пошук здійснюється дуже швидко. Автор алгоритму зазначає, що метод впровадження за допомогою векторного квантування є більш гнучким порівняно зі скалярним випадком і дозволяє краще контролювати робастність, рівень спотворень [13].

3.3.9 Алгоритм Басу

Особливістю алгоритму є факт відсутності як такого зовнішнього ЦВЗ. Інформація вбудовується за рахунок зміни зображення, що породжує самоподібність. Алгоритм дозволяє впровадити до 15 різних ЦВЗ.

Опишемо роботу алгоритму. Спочатку вибираються кілька «особливих» точок, причому використовується метод Стефана-Харріса. Кожна така точка визначає блок розміром 4х4 навколо неї та 16 блоків розміром 4х4, що утворюють доменний пул. Для кожної точки виконують зміну доменного блоку таким чином, щоб він був схожий на ранговий блок, ніж будь-який інший доменний блок. (Оскільки можна вибрати15 блоків, це дає можливість впровадити15 ЦВЗ). Доменний блок, що вийшов, визначається виразом

$$W_j = \sim \frac{D_j - \bar{D}_j}{\max(D_j - \bar{D}_j)}, \tag{3.23}$$

де $\bar{D}$ – середнє арифметичне значення пікселів у D. Він додається до R_j відповідно до виразу

$$R_j^{'} = W_j + \mathrm{int}(\frac{R_j}{s})s + \frac{s}{2} \tag{3.24}$$

де величина s називається коефіцієнтом квантування.

При вилученні ЦВЗ спочатку відновлюються значення спеціальних точок, D_j і W_j. Для кожного блоку R_j обчислюється

$$\widehat{W}_j^{'} = R_j - \mathrm{int}(\frac{R}{s})s - \frac{s}{2} \tag{3.25}$$

Далі знаходиться найбільш схожий блок, який має бути тим самим, що і в процесі вбудовування. Число збіглих блоків визначає ймовірність присутності ЦВЗ у зображенні.

Перевагами даного алгоритму є вбудовування до 15 ЦВЗ та відносна простота реалізації декодера.

4. ДОСЛІДЖЕННЯ АЛГОРИТМІВ ГЕНЕРУВАННЯ ТА ВБУДОВУВАННЯ ЦВЗ У ЗОБРАЖЕННЯ

Алгоритми генерування та вбудовування ЦВЗ почали розроблятися нещодавно, і досі у цій галузі є багато невирішених проблем.

Головними з них є проблема збереження якості контейнера-зображення при впровадженні ЦВЗ та стійкість (робастність) вбудовуваних міток до можливих перетворень контейнера (афінних трансформацій, обрізання, накладання фільтрів, зміни яскравості тощо) при забезпеченні достовірності подальшого відновлення ЦВЗ.

Іншими важливими питаннями є час роботи алгоритмів, їхня обчислювальна складність, пропускна здатність. Також для параметричних алгоритмів існує проблема грамотного підбору параметрів при вбудовуванні ЦВЗ зображення, оскільки грамотний підбір параметрів сприяє ефективній роботі алгоритму.

Метою даної кваліфікаційної роботи є вивчення даних питань, підбір оптимальних для кожного формату та типу зображень алгоритмів (як контейнери обрані різні зображення 4 популярних растрових форматів для представлення графічних да-них: JPEG, BMP, PNG та TIFF).

Для проведення дослідження було створено програмний комплекс, реалізований серед програмування Microsoft Visual Studio 2019 мовою програмування C++ при використанні бібліотеки для роботи із зображеннями OpenCV 2.

Для дослідження було обрано 4 алгоритми: вбудовування в найменш значний біт (LSB), алгоритм Куттера-Джордана-Боссена (Kutter), алгоритм Коха-Жао (Koch) та алгоритм Сміта-Коміски (Smith). Кожен з них є найпопулярнішим у своїй групі.

Алгоритм вбудовування найменш значний біт є найпопулярнішим представником серед алгоритмів вбудовування в просторову область. У роботі була досліджена його модифікація, яка передбачає вбудовування інформації в синю компоненту кожного другого пікселя зображення, починаючи з верхнього лівого його кута.

Алгоритм Куттера-Джордана-Боссена передбачає зміну значення синьої компоненти псевдовипадково вибраних пікселів на величину прямо пропор-ційну яскравості даного пікселя.

Алгоритм Коха-Жао є сімейством вбудовування ЦВЗ в область дискретного косинусного перетворення. Має на увазі зміну обраних коефіцієнтів ДКП на заздалегідь задану величину.

Алгоритм Сміта-Коміски - найбільш використовується з алгоритмів розширення спектра. Має на увазі модуляцію вихідного повідомлення базовими функціями. При вбудовуванні ЦВЗ значення пікселів змінюється на величину, яка залежить від похибки ортогональності базисних функцій та значень пікселів зображення-контейнера.

4.1 Дослідження стійкості алгоритмів до візуальних спотворень

Оцінювання візуальної стійкості стеганографічної системи до зовнішніх впливів є досить складним завданням.

Досить ефективними є методи оцінки рівня приховування виходячи з аналізу статистичних характеристик. Для оцінювання якості стеганографічних засобів розробляються показники, що дають кількісні оцінки. Найчастіше такі показни-ки оперують із зображеннями лише на рівні пікселів і навіть окремих RGB-компонент пікселів. Найбільш важливими показниками при аналізі рівня спо-творень, які вносяться до зображення під час приховування інформації, є співвідношення «сигнал/шум», обчислене в децибелах і якість зображення, що вимірюється у відсотках. Серед інших показників показників можна виділити середню абсолютну різницю значень пікселів, нормовану середню абсолютну різницю значень пікселів та максимальне відношення «сигнал-шум».

Далі наведено список вище зазначених різницевих показників спотворення та формули для їх обчислення.

Середня абсолютна різниця (Average Absolute Difference):

$$AD = \frac{1}{XY}\sum_{x,y}|C_{x,y} - S_{x,y}| \qquad (4.1)$$

Нормована середня абсолютна різниця (Normalized Average Absolute Difference):

$$NAD = \frac{\sum_{x,y}|C_{x,y} - S_{x,y}|}{\sum_{x,y}|C_{x,y}|} \qquad (4.2)$$

Відношення "сигнал/шум" (Signal to Noise Ratio):

$$SNR = \frac{\sum_{x,y}(C_{x,y})^2}{\sum_{x,y}(C_{x,y} - S_{x,y})^2} \qquad (4.3)$$

Максимальне відношення "сигнал/шум" (Peak Signal to Noise Ratio):

$$PSNR = XY * \frac{\max_{x,y}(C_{x,y})^2}{\sum_{x,y}(C_{x,y} - S_{x,y})^2} \qquad (4.4)$$

Якість зображення (Image Fidelity)

$$IF = 1 - \frac{\sum_{x,y}(C_{x,y} - S_{x,y})^2}{\sum_{x,y}(C_{x,y})^2} \qquad (4.5)$$

У рамках дослідження було обчислено наведені вище показники тестових зображень.

Як тестові зображення були обрані зображення представлені на рис. 4.1-4.7

Рисунок 4.1 – Зображення пейзажа (Nure.jpg)

Рисунок 4.2 – Зображення з багатою палітрою кольорів (Tulips.jpg)

Рисунок 4.3 – Фотографія з невисокою чіткістю, зроблена з аматорської фотокамери (photo.jpg)

Рисунок 4.4 – Точкове зображення-портрет (Lena.bmp)

Рисунок 4.5 – репродукція картини (city.png)

Рисунок 4.6 – Логотип, створений у графічному редакторі (Logo.png)

Рисунок 4.7 – Космічний знімок, представлений у форматі TIFF (sky.tiff)

Результати вбудови кожного зображення-контейнера будуть представлені в таблицях і проаналізовані.

Як повідомлення було обрано тестова фраза “ХНУРЕ кращій серед кращіх”, подана у вигляді рядка символів ASCII (8 біт однією символ). Довжина рядка – 25 символів, розмір повідомлення – 200 біт.

Стегосистеми з контейнером Nure.jpg, дане зображення є фотографією-пейзажем розміром 1024x768 пікселів. Зображення чітке, плавні переходи (різниця значень суміжних пікселів невелика – особливо це стосується синьої компоненти пікселів). У цьому зображенні домінуючою є синя компонента. Червона компонента практично відсутня, зелена – як і синя, серйозно виражена, але представлена дуже нерівномірно.

Отримані значення параметрів візуального спотворення для зображення наведено в таблиці 4.1.

Таблиця 4.1 – Значення параметрів візуального спотворення при вбудовуванні зображення Nure.jpg

	AD	NAD	SNR	PSNR	IF
LSB	0,480371	0,003992	34762,4	121153	99,997%
Kutter	4,14126	0,034373	27,0141	87,912	96,556%
Cox	0,938733	0,009512	23456,9	66099,6	99,995%
Smith	0,873914	0,009255	18567,1	50406,7	99,994%

За всіма наведеними параметрами найкращі показники продемонстрував алгоритм LSB. Цей алгоритм не дублює інформацію, що забезпечує дуже високе відношення "сигнал-шум" і вбудовує ЦВЗ в незначні біти пікселів, що практично не змінює статистику контейнера (мале значення середньої абсолютної різниці, а отже, і ідеальне значення якості зображення).

Дуже хороші показники, лише трохи поступаються LSB, показують алгоритми Кох і Smith. Високі результати Кох пов’язані в основному з тим, що це зображення стисло у форматі JPEG. При стиску використовувалося те саме перетворення (дискретне косинусне), що і при вбудовуванні ЦВЗ. Що стосується Smith, його низькі показники спотворення пов’язані з малою похибкою ортогональності та великою потужністю сигналу.

Найменш стійким є алгоритм Kutter. Основною проблемою є слабкість його декодера, через що стало важким завданням вірне розпізнавання ЦВЗ. Яскравість зображення виявилася недостатньою (найвагомішою компонентою для яскравості є зелена, а вона слабко виражена в цій фотографії), в результаті довелося збільшувати коефіцієнт енергії сигналу[3] до 0,15. А через насиченість синьої компоненти проблематичним завданням стало визначення значення пікселя в середині «хреста», як середнє арифметичне його оточуючих пікселів. Через це довелося дублювати вбудовування 250 разів ($^2 = 250$), у результаті знизився показник “сигнал-шум”. В результаті, при якості 96% присутні видимі спотворення, що ускладнює використання алгоритму для даного зображення. Дещо зменшити помітність застосування можна вбудувавши ЦВЗ в червону компоненту пікселів. Коректне розпізнавання тексту відбувається без підвищення коефіцієнта потужності сигналу (^{3} =0,05), що знижує видимі візуальні спотворення.

Стегосистеми з контейнером Tulips.jpg, дане зображення є фотографією з яскравими та насиченими кольорами, роздільна здатність зображення 1024x768 пікселів. На зображенні досить багато деталей, задній план відчутно розмитий. Синя компонента розподілена вкрай нерівномірно, червона досить слабо виражена; домінує зелена компонента, що сприяє збільшенню яскравості. Переходи між пікселями (різниця кольорів сусідніх пікселів) щодо плавні. Отримані в результаті експерименту показники наведено у таблиці 4.2.

Таблиця 4.2 – Показники візуального спотворення при вбудовуванні ЦВЗ зображення Tulips.jpg

	AD	NAD	SNR	PSNR	IF
LSB	0,50	0,0072	30098	130021	99,99%
Kutter	0,64	0,0095	96,96	404,761	98,96%
Cox	0,61	0,0090	24078,8	104635	99,99%
Smith	0,93	0,0140	9989,6	44986,9	99,99%

Високу візуальну стійкість показав алгоритм Кох. Внаслідок нерівномірності розподілу синьої компоненти деякі вибрані коефіцієнти ДКП відрізняються один від одного більше ніж на одиницю, через це значення параметра різниці було збільшено (P=2). Незначне збільшення параметра P не призвело до серйозних спотворень зображення.

Хороші результати показав алгоритм Smith, поступившись Кох незначною мірою як зображення і в середній абсолютній різниці значень пікселів.

Загалом непогані результати показав алгоритм Kutter. Високий показник яс-кравості дозволив знизити коефіцієнт енергії сигналу в 5 разів ( ), а відносно високий ступінь передбачуваності значення пікселів зображення доз-волило знизити кількість вбудовувань до 50. У результаті отримано прий-нятне відношення "сигнал-шум" і відносно невелика зміна значень пікселів, а отже, і слабка візуальна помітність вбудовування (коефіцієнт якості близько 99%).

Стегосистеми з контейнером photo.jpg це зображення є фотографією, зроблена на аматорську фотокамеру. Рух камери спричинив деяку змазаність зображення. Через технічну недосконалість камери кольору досить тьмяні, зображення нечітке. Розмір зображення 1600×1200 пікселів. Червона, синя та зелена компоненти мають приблизно однакові нерівномірні розподіли. Отримані значення параметрів візуального спотворення для зображення наведено в таблиці 4.3.

Таблиця 4.3 – Показники візуального спотворення при вбудовуванні зображення photo.jpg

	AD	NAD	SNR	PSNR	IF
LSB	0,500186	0,005061	29710,2	130002	99,9966%
Kutter	0,207647	0,002101	2441,74	10684,2	99,959%
Cox	0,994167	0,01006	14947,8	65406,5	99,9933%
Smith	0,871591	0,00882	13141,3	57501,7	99,9924%

За значенням середньої абсолютної різниці лідером є алгоритм Kutter, що з низькими показниками яскравості контейнера. Але через нечіткість зображення статистичний декодер працює неефективно, тому для правильного вилучення довелося збільшувати кількість інтеграцій. до 250. Проте цей факт негаразд позначився на спотворенні внаслідок більшого проти іншими зображеннями розміру.

Близькі один до одного результати показали стегосистеми, що використовують алгоритми Cox та Smith. Найкращі показники першої з них пов’язані з тим, що це зображення стисло у форматі JPEG.

Стегосистеми з контейнером Lena.bmp, дане зображення є портретом у форматі BMP (крапковий малюнок) з невеликою роздільною здатністю 512x512. Формат BMP забезпечує максимальну якість зображення, тому що не передбачає стиснення.

Будь-який портрет містить досить велику кількість дрібних деталей. Кольо-ри приглушені (знімок зроблений у середині XX століття). Домінує червона компонента, зелена та синя виражені середньо. Отримані значення параметрів візуального спотворення для зображення наведено в таблиці 4.4.

Таблиця 4.4 – Показники візуального спотворення при вбудові зображення Lena.bmp

	AD	NAD	SNR	PSNR	IF
LSB	0,50032	0,004746	24526,8	101185	99,9959%
Kutter	0,9866	0,00936	1723,36	7109,71	99,942%
Cox	18,2043	0,1727	35,8981	148,097	97,2143%
Smith	0,859631	0,008155	11852,6	48897,7	99,9916%

Найкращі показники візуального спотворення мають алгоритми LSB та Smith. Хороші показники має алгоритм Kutter. Це пов’язано з не найвищою яскравістю зображення та високою кореляцією між сусідніми пікселями.

Найгірші показники має алгоритм Кох, який використовує ДКП, тоді як формат зображення його не використовує. Але головною проблемою для Кох став невеликий розмір зображення, що різко знизило ставлення сигнал-шум і призвело до візуальної помітності вбудовування. Від використання цього алгоритму рекомендується відмовитись.

City є цифровою репродукцією картини у форматі PNG розміром 1200x801 пікселів. Репродукції картин характеризуються невисокою яскравістю та наявністю великої кількості деталей. Зображення затемнене, кольори не насичені. Домінуючої колірної компоненти немає. Отримані параметри візуального спотворення для цього зображення наведені в таблиці 4.5.

Таблиця 4.5 – Показники візуального спотворення при вбудовуванні зображення City.png

	AD	NAD	SNR	PSNR	IF
LSB	0,499378	0,007036	13332,3	88309,9	99.9925%
Kutter	0,188796	0,00266	6548,79	43377,6	99,9847%
Cox	27,9807	0,394223	8,25231	54,6611	87,8821%
Smith	0,827543	0,011659	7248,92	48015,1	99,9862%

Алгоритми LSB, Kutter та Smith показують практично ідеальні результати. При цьому Kutter показує найнижчий показник середньої абсолютної зміни значень пікселів (внаслідок низької яскравості зображення). Будь-який з цих алгоритмів може бути застосований для вбудовування ЦВЗ у це тестове зображення.

А ось алгоритм Кох застосовувати не варто. Вибрані середньочастотні коефіцієнти ДКП близькі один до одного за значенням і їх необхідно змінювати, що призводить до спотворень зображення. Це пов’язано з особливостями фор-мату та низькою яскравістю зображення.

Зображення Logo є логотипом, створеним у графічному редакторі та збереженим у картинці формату PNG. Зображення є повністю штучним, згладжування відсутнє (спостерігається ефект «драбинки»). Роздільна здатність зображення-контейнера 468x508. Отримані значення параметрів візуального спотворення для зображення наведено в таблиці 4.6.

Таблиця 4.6 – Показники візуального спотворення під час вбудови у зображення Logo.png

	AD	NAD	SNR	PSNR	IF
LSB	0,499449	0,002513	95744,2	130193	99,999%
Kutter	10,9237	0,054136	19,6862	26,6672	94,9203%
Cox	0,883	0,004376	54150,1	73352,4	99,9982%
Smith	0,996071	0,004936	27691,8	37511,7	99,9964%

Алгоритми LSB, Кох та Smith показують результати, близькі до еталонних. Високі результати алгоритму Кох пов’язані з високою контрастністю зображення, а тому і досить великим розкидом коефіцієнтів ДКП. Для алгоритму Кох показник різниці коефіцієнтів ДКП P було збільшено до 2 для правильного визначення повідомлення. Для зменшення візуальних спотворень алгоритму Kutter кількість вбудовувань[2] було зменшено до 170 , але навіть із зменшеною кількістю вбудовувань цей метод не слід застосовувати для вбудовування ЦВЗ у дане зображення. Високі показники спотворення пов’язані з великим (практично максимальним) значенням яскравості для деяких пікселів і білим тлом зображення, на якому помітні навіть малі спотворення, а також з малим розміром зображення, яке здійснюється вбудовування.

Sky є зображенням, отриманим з одного з космічних телескопів, та збереженим у форматі TIFF без стиснення з роздільною здатністю 650x487 пікселів. Зображення містить велику кількість контрастних деталей. Домінуюча компонента-червона, синя представлена досить слабо і нерівномірно, зелена компонента насичена, але розподілена нерівномірно. Деякі області майже безбарвні (чорні області). Отримані значення параметрів візуального спотворення для зображення наведено в таблиці 4.7.

Таблиця 4.8 – Показники візуального спотворення при вбудовуванні зображення MilkyWay.tiff

	AD	NAD	SNR	PSNR	IF
LSB	0,499346	0,008378	14982,6	130220	99,9933%
Kutter	0,383127	0,006428	885,517	7696,4	99,8871%
Cox	0,982208	0,016479	7609,06	66133,5	99,9869%
Smith	18,6553	0,312991	14,4246	125,37	93,0674%

Результати, близькі до ідеальних, показують алгоритми LSB і Кох. Завдяки темному загальному фону зображення, а отже, і більшій помітності деталей вдалося знизити кількість вбудовувань[2] в алгоритмі Kutter до 180. Відставання від перших двох алгоритмів викликане невеликою пропускною здатністю при велику кількість повторних вбудов для даного малого зображення.

Слабка потужність ЦВЗ призвела до проблем із вірним декодуванням повідомлення у Smith. Цю проблему вдалося вирішити лише посиленням потужності сигналу K_G у 20 разів, що спричинило значне погіршення якості зображення. Вбудовування в інші колірні компоненти не змінює ситуації. Від використання алгоритму Smith у разі слід відмовитися.

Для підбиття підсумків введемо деяку шкалу оцінок результатів експериментів. За найкращий результат за критерієм якості зображення алгоритму нараховується 4 бали, за другий результат -3 бали, за третій -2, за четвертий -1 і якщо алгоритм не рекомендовано застосовувати -0 балів.

Результати дослідження наведемо у таблиці 4.8.

Таблиця 4.8 – Результати дослідження стійкості алгоритмів до візуальних спотворень

	Nure	Tulips	photo	Lena	city	Logo	sky	Підсумок
LSB	4	4	4	4	4	4	4	28
Kutter	1	1	1	2	2	0	2	9

Продовження таблиці 4.8

Kox	3	3	3	0	0	3	3	15
Smith	2	2	2	3	3	2	0	14

Найкращий результат для всіх зображень показав алгоритм LSB. Це пов'язано з його високою пропускною здатністю та вбудовуванням у найменш значущий біт, який практично не змінює статистику зображення.

Друге місце зайняв алгоритм Kox, що показав хороші результати здебільшого зображень. Але, головним чином, область його застосування-це зображення у форматах, що використовують ДКП (найвідоміший -JPEG). Для інших зображень необхідно оцінювати розмір зображення та його яскравість. Третє місце зайняв алгоритм Smith, який показав середні результати для всіх зображень. Цей алгоритм можна назвати універсальним і досить надійним з погляду візуальних спотворень. Недоліком є те, що у всіх експериментах є алгоритми, що показують кращі результати.

Останнє місце зайняв алгоритм Kutter. Область застосування сильно обмежена через слабкого статистичного декодера. Доцільно його використання при вбудовуванні зображення із середньою насиченістю колірної компоненти, в якій здійснюється вбудовування, за наявності помітної кореляції між значеннями сусідніх пікселів у зображенні і при невисокому значенні яскравості.

4.2 Дослідження пропускної спроможності стегосистем, що використовують обрані алгоритми

Для стегосистем важливо визначити пропускну спроможність каналів передачі повідомлень, що приховуються, та її залежність від інших характеристик стегосистем та умов їх використання. Пропускна здатність стегосистеми визначається як максимальна кількість інформації, яку можна передати без помилок за її використання.

Під прихованою пропускною здатністю будемо розуміти максимальну кількість інформації, яка може бути вкладена в один елемент контейнера, при цьому повідомлення, що приховуються, повинні бути безпомилково передані одержувачу і захищені від атак порушника, таких як спроби виявлення факту наявності каналу прихованого зв’язку, читання повідомлень, що приховуються, введення помилкових повідомлень чи руйнування вбудованої інформації [4].

Логічно припустити, що прихована пропускна здатність повинна бути меншою за пропускну здатність системи.

Існують два підходи до визначення кількості інформації, що захищається від різних атак порушника стеганографічними методами.

Перший з них [15] застосуємо до стегосистем, в яких повідомлення, що захищаються, повинні бути безпомилково передані в умовах активної протидії порушника. Цей підхід описує приховування повідомлення в контейнерних даних, і враховує, що крім спотворень при впровадженні в контейнер можливі і навмисні спотворення з боку порушника, а також спотворення випадкового характеру, спричинені ненавмисними перешкодами каналу зв’язку або спотворення при стисненні контейнера. Порушник, що розглядається, крім методів стегоаналізу, може використовувати і активні дії. Метою порушника може бути руйнування інформації, що приховується.

У такому випадку завдання інформаційного приховування формулюється як завдання безпомилкової передачі прихованої інформації при впливі випадкових і навмисних перешкод. Цей підхід визначає лише теоретично досяжну швидкість достовірної передачі повідомлень, що приховуються. При цьому враховані умови, за яких інформація, що приховується, гарантовано передається в умовах довільних спроб порушника з її руйнування.

Другий підхід, що розвивається в роботі [16], дає оцінки прихованої пропускної спроможності при вкладенні повідомлень, що приховуються, в надлишкові контейнерні дані. У цій задачі оцінки пропускної здатності залежать від характеристик візуальної непомітності прихованого каналу.

Даний підхід орієнтований на стегосистеми, в яких реалізується прихована передача невідомій одержувачу інформації, при цьому порушник намагається виявити факт наявності прихованого зв’язку і, при встановленні цього факту, прочитати інформацію, що приховується. Відомо велика кількість алгоритмів, які передбачають різні способи вкладення контейнери (деякі їх вже було розглянуто у розділі 3). Автори цих робіт оцінюють максимально можливу кількість інформації, яку можна вкласти непомітно з урахуванням критеріїв невиявлення, що використовуються.

Розглянемо пропускну здатність, використовуючи другий підхід, для стегосистем, заснованих на використанні алгоритмів: вбудовування у найменш значний біт, Куттера-Джордана-Боссена, Коха-Жао та Сміта-Коміски. Під пропускною спроможністю С розумітимемо максимальну кількість символів ASCII (1 символ=8 біт), яку можна передати за один такт роботи стегосистеми. Приховану пропускну спроможність $\tilde{C}$ визначимо як максимальну кількість символів, яку можна вбудувати у зображення-контейнер без істотних втрат як зображення (візуальної непомітності).

Результати кожного алгоритму представлені як графіків.

Пропускна здатність стегосистеми з обраною для дослідження модифікації алгоритму LSB (вбудовування у кожен другий біт синій компоненти зображення) $C = M * N / 16$, де M і N – довжина та ширина контейнера. При використанні базової версії алгоритму пропускна спроможність стегосистеми у 6 разів більша. Отримані експериментально значення пропускної та прихованої пропускної спроможності стегосистем, що використовують алгоритм вбудовування у найменш значний біт, представлені на рис. 4.8.

Рисунок 4.8 – Пропускна спроможність стегосистем, що використовують алгоритм LSB

Теоретично пропускна здатність стегосистеми, що використовує даний алгоритм, дорівнює $M*N/8$. Але, враховуючи слабкість статистичного детектора, за такого обсягу повідомлення годі й говорити про правильну інтерпретацію повідомлення. Тож фактична пропускна спроможність алгоритму значно менша.

Розглянемо пропускну спроможність при параметрах, підібраних кожного зображення щодо візуальних спотворень. Для зображення Nure.jpg $^{2} = 200;^{3} = 0,05$ при вбудовуванні у червону компоненту; для зображення Lena.bmp $^{2} = 200;^{3} = 0,05$; для City.png $^{2} = 200;^{3} = 0,05$; для Sky.tiff $^{2} = 180;^{3} = 0,05$. Отримані значення пропускної та прихованої пропускної спроможності стегосистем, що використовують при вбудовуванні алгоритм Куттера-Джордана-Боссена, представлені на рис. 4.9

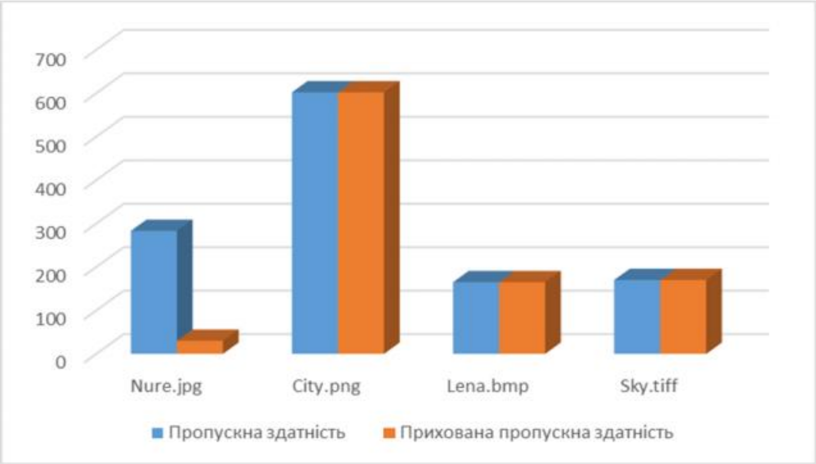

Рисунок 4.9 – Пропускна спроможність стегосистем, що використовують алгоритм Куттера

Оскільки алгоритм Коха-Жао оперує блоками 8х8 пікселів, теоретично пропускна здатність C дорівнює $M * N / 512$. Насправді при такому обсязі вбудованої інформації контейнер іноді буває настільки спотворений, що неможливо правильно визначити вбудоване повідомлення.

Отримані експериментально значення пропускної та прихованої пропускної спроможності стегосистем, що використовують при вбудові алгоритм Коха-Жао, представлені на рис. 4.10.

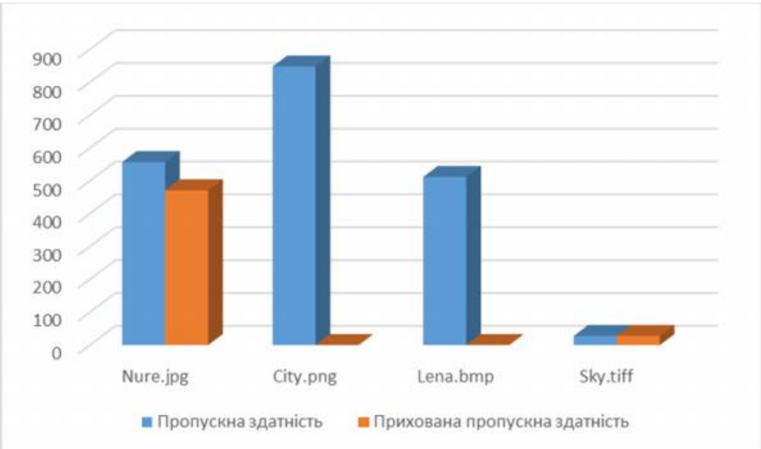

Рисунок 4.10 – Пропускна спроможність стегосистем, що використовують алгоритм Коха

У методі Сміта-Комиски біти повідомлення при вбудовуванні модулюються базисними функціями, так що пропускна здатність стегосистеми, заснованої на даному алгоритмі, залежить від числа базисних функцій і в ідеалі дорівнює значенню, близькому до $M * N / 8$.

Вивчимо роботу декодера та отримані в результаті експериментів візуальні спотворення, і отримаємо значення пропускної здатності та прихованої пропускної здатності.

На рис. 4.11 представлені отримані в результаті експериментів значення пропускної та прихованої пропускної спроможності стегосистем, які використовують при вбудові алгоритм Сміта-Коміски.

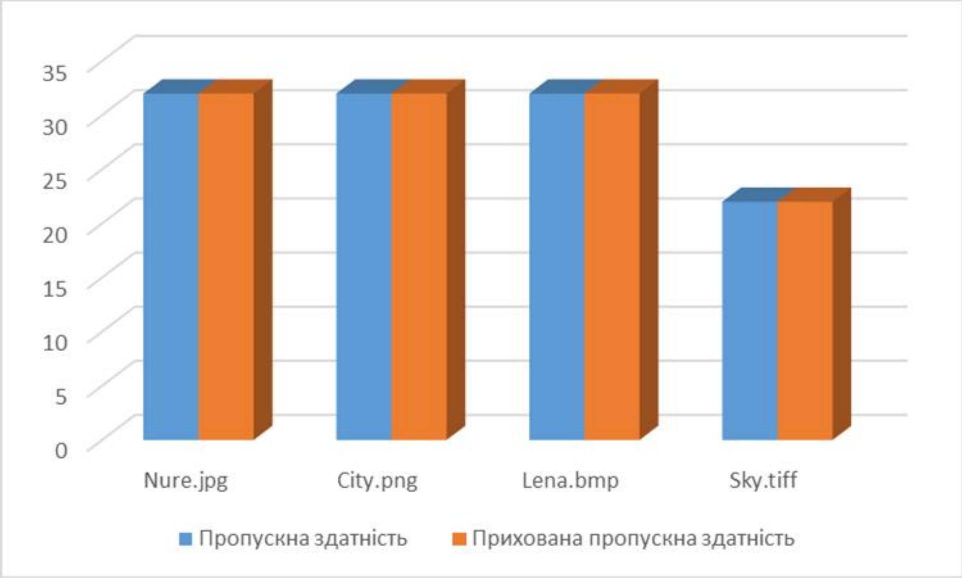

Рисунок 4.11 – Пропускна спроможність стегосистем, що використовують алгоритм Сміта-Коміски

Як видно з результатів експериментів, в більшості випадків прихована пропускна здатність дорівнює пропускній здатності алгоритмів. Цей факт підтверджує, що обрані для дослідження алгоритми враховують особливості людського зору та мінімізують спотворення навіть при вбудовуванні максимально можливої кількості символів у зображення.

Найвищу пропускну здатність (а, отже, і приховану пропускну здатність) має алгоритм вбудовування в найменш значний біт. Це з тим, що вбудовування йде попиксельно, а чи не поблочно (що піднімає теоретичну пропускну спроможність) і з екстремально низькими показниками візуального спотворення.

Алгоритм Куттера-Джордана-Боссена має на увазі багаторазове дублювання інформації. Це веде до значного зменшення пропускної спроможності. Серйозна різниця між ПС та прихованою ПС для зображення Desert.jpg викликана тим, що вбудовування ведеться в червону компоненту, чия зміна помітніша для СЛЗ, ніж у синю.

Вивчення алгоритму Коха-Жао дало дуже суперечливі результати. Незважаючи на те, що алгоритм оперує блоками пікселів, а не окремими пікселями, для більшості зображень пропускна здатність виявилася вищою, ніж при використанні алгоритму Куттера-Джордана-Боссена. Але головною проблемою стала обмежена сфера його застосування. Як результат, алгоритм Коха-Жао не може застосовуватися для непомітного вбудовування ЦВЗ у два зображення із чотирьох тестових.

Найнижчу пропускну спроможність має алгоритм Сміта-Коміски. Для збільшення пропускної спроможності цього алгоритму потрібно різко збільшити кількість базисних функцій, що вимагає виділення величезної кількості пам’яті і призводить до роботи з безліччю величезних числових масивів, а, отже, і серйозним витратам часу, що робить даний алгоритм незручним для застосування. А зворотна залежність між якістю роботи декодера та кількістю базисних функцій робить недоцільним такий спосіб збільшення пропускної спроможності.

4.3 Дослідження робастності алгоритмів до різних видів атак

Усі атаки на системи вбудовування ЦВЗ можуть бути поділені на три групи: атаки, створені задля видалення ЦВЗ; геометричні атаки, створені задля спотворення контейнера; криптографічні атаки.

Атаки, спрямовані на видалення ЦВЗ. До цієї групи належать фільтрація контейнерів, перемодуляція, стиск із втратами, усереднення та колізії [5]. Ці атаки засновані на припущенні про те, що ЦВЗ має шумовий характер.

Фільтри можуть використовувати критерії максимальної правдоподібності чи максимальної апостеріорної ймовірності. Як фільтр також може використовуватися медіанний або середній фільтр.

Стиснення з втратами та очищення сигналів від шумів значно знижують пропускну здатність стегоканалу, особливо для гладких областей зображення, коефіцієнти ДКП яких можуть бути видалені без помітного зниження якості зображення.

Перемодуляція – порівняно новий метод, який використовується для атак на ЦВЗ. Нехай ЦВЗ впроваджено зображення із застосуванням широкосмугових сигналів і розмножено попри всі зображення. Оскільки оцінюваний декодером ЦВЗ корелюється з істинним, з'являється можливість обману декодера. Розглянемо механізм атаки[5].

Спочатку ЦВЗ "передбачається" шляхом віднімання фільтрованої версії зображення з вихідного контейнера. «Предказаний» ЦВЗ піддається високочастотної фільтрації, усікається, множиться на два і віднімається з вихідного зображення. Крім того, якщо відомо, що при впровадженні ЦВЗ використовувалася деяка маска для зменшення помітності вбудовування, атакуючий оцінює цю маску і примножує на неї ЦВЗ.

Однак така атака ефективна тільки проти високочастотного ЦВЗ, тому реальні ЦВЗ будуються так, щоб їхній спектр був максимально корелюваний зі спектром вихідного зображення. Також для протидії цій атаці використовується попередня низькочастотна фільтрація зображення.

До інших атак цієї групи відносяться атака усереднення та атака змови. У разі наявності великої кількості копій стиг з різними ЦВЗ або з різними ключами впровадження можна виконати їх усереднення.

Атака шляхом статистичного усереднення представлена у [5]. Порушник може спробувати оцінити ЦВЗ і відняти його із зображення. Такий вид атак особливо небезпечний у разі, коли атакуючий може отримати деякий узагальнений ЦВЗ, наприклад, деякий $W = F(S_0, W)$, що практично не залежить від вихідного зображення S_0.

Атакуючий може виявити ЦВЗ шляхом усереднення кількох зображень. Наприклад, у нього є $S_0 + W, S_1 + W, ..., S_N + W$.

Тоді їхня сума буде досить близька до NW, якщо N велике, а зображення статистично незалежні.

Проти цій атаці можна випадкове використання одного з двох ЦВЗ з ймовірностями p_1 і $p_2 = 1 - p_1$. Тоді вищенаведена атака дасть лише $p_1 W_1 + (1 - p_1) W_2$.

При атаці змови є кілька однакових копій, що містять різні ЦВЗ, а для атаки з кожної копії вибираються якісь частини, які в сукупності і утворюють безліч, що атакується. Чим більше копій, що містять стего, є у порушника, тим вище ймовірність того, що близьке до вихідного відновлене зображення не міститиме стего. Робастність до цієї атаки можна різко збільшити за рахунок спеціальної побудови стего та використання стегосистем із закритим ключем.

Геометричні атаки змінюють ЦВЗ, вносячи просторові чи тимчасові спотворення. Геометричні атаки являють собою афінні перетворення з невідомим декодеру параметром. Усього є шість афінних перетворень: масштабування, зміна пропорцій, повороти, зсув та усічення. Ці атаки призводять до порушення детекції ЦВЗ. При цьому можливе вирізування окремих пікселів, а також перестановка їх місцями.

Деякі методи вбудовування ЦВЗ робастні до афінних перетворень. У цьому робастность досягається з допомогою використання інваріантних до зсуву областей, застосування опорного ЦВЗ, обчислення автокореляційної функції ЦВЗ.

Криптографічні атаки названі так тому, що вони є аналогами атак у криптографії. До них відносяться атаки з використанням оракула, а також злому за допомогою «грубою сили».

Атака з використанням оракула дозволяє створити незахищене ЦВЗ зображення за наявності у порушника детектора. Проведення цієї атаки полягає у експериментальному вивченні поведінки детектора. Наприклад, якщо детектор виносить «м’які» рішення, тобто показує ймовірність наявності стего в сигналі, то атакуючий може з’ясувати, як невеликі зміни зображення впливають на поведінку детектора. Модифікуючи зображення пікселя за пікселем, він може з’ясувати, який алгоритм використовує детектор. У разі детектора з «жорстким» рішенням атака здійснюється біля кордону, де детектор змінює своє рішення з «присутньою» на «відсутньою» [5].

На даний момент немає стегоалгоритмів, стійких проти подібних атак.

Вибрано атаки, які вносять такі зміни, які не сильно спотворюють зображення-стего. Ці атаки є найефективнішими, оскільки одержувач, які мають вихідного зображення, приймає спотворення особливості зйомки (у разі фотографій) чи зовсім їх помічає. У програмному комплексі реалізовані поворот контейнера-зображення на 90 градусів вліво (такий поворот часто зустрічається при збереженні фото), стиснення в 2 рази (одержувачу не відомий вихідний розмір зображення), згладжування за допомогою функції Гауса, збільшення контрастності, збільшення та зменшення яскравості, ерозія зображення, обрізання 20% зображення з різних боків, стиснення JPEG з різними показниками якості. Як контейнери будуть використані тестові зображення, як повідомлення-тестове повідомлення (рядок) "ХНУРЕ кращі серед кращіх".

Робастність стегосистем залежить від вибору алгоритму, підбору параметрів для конкретного зображення та довжини вбудованого повідомлення.

Розглянемо робастність стегосистем, використовують у вбудовуванні досліджувані у роботі алгоритми, до найпоширенішим видам атак. Отримані результати зведемо до таблиць для стегосистем з різними контейнерами. Знаком "+" позначимо робастність стегосистеми до цієї атаки, знаком "-" - нестійкість до атаки.

Після кожної таблиці залишимо примітки, що стосуються зниження прихованої пропускної спроможності та зміни параметрів алгоритмів.

У таблиці 4.9 представлені результати експерименту для стегосистем із контейнером Nure.jpg.

Таблиця 4.9 – Робастність стегосистем, що використовують контейнер Nure.jpg до атак

	LSB	Kutter	Kox	Smith
Масштабування	-	-	-	-
Поворот	-	-	+	+
Фільтри Гауса	-	-	+	+
Контрастність	-	+	+	+
Підвищення яскравості	-	+	+	+
Зниження яскравості	-	+	+	+
Ерозія	-	-	+	+
Обрізання праворуч	+	+	-	-
Обрізання зліва	-	+	-	-
Обрізання знизу	+	+	+	-
Обрізання зверху	-	+	-	-
Стиснення JPEG з коефіцієнтом якості 0,7	-	-	+	+
Стиснення JPEG з коефіцієнтом якості 0,5	-	-	+	+

Для стійкості стегосистеми, що використовує при вбудовуванні алгоритм Куттера до обрізання країв (20% з краю), необхідно збільшити параметр числа повторень ^{2} до 450, для робастності до зміни яскравості-до

550. Дана система дозволяє вбудовувати не більше 11 символів робастно до фільтрації за допомогою функції Гауса.

Для забезпечення робастності до зміни яскравості стегосистеми, побудованої на використанні алгоритму Коха, необхідно збільшити різницю коефіцієнтів P до 10. Така стегосистема дозволяє вбудувати лише 7 символів (наприклад, ініціали автора зображення) робастно до масштабування та 12 символів до спотворення ЦВЗ за допомогою обрізання .

У таблиці 4.10 наведено результати експерименту для стегосистем з контейнером Lena.bmp.

Таблиця 4.10 – Робастність стегосистем, що використовують контейнер Lena.bmp до атак

	LSB	Kutter	Kox	Smith
Масштабування	-	-	-	-
Поворот	-	-	+	+
Контрастність	-	+	+	+
Підвищення яскравості	-	+	+	+
Зниження яскравості	-	+	+	+
Ерозія	-	-	+	+
Обрізання праворуч	-	+	-	-
Обрізання зліва	-	+	-	-
Обрізання знизу	+	+	+	-
Обрізання зверху	-	+	-	-
Стиснення JPEG з коефіцієнтом якості 0,7	-	-	+	+
Стиснення JPEG з коефіцієнтом якості 0,5	-	-	+	+

Для стійкості стегосистеми, що використовує алгоритм вбудовування найменш значний біт до обрізання частини зображення (трохи більше 20% з кожної із сторін) праворуч, необхідно зменшити кількість символів до 10 (внаслідок невеликого розміру зображення).

Стегосистема, заснована на застосуванні алгоритму Коха, стійка до масштабування зображення та видалення частини зображення. При цьому катастрофічно падає пропускна здатність: у першому випадку – до 4 символів (досить лише для вбудовування ініціалів), у другому – до 6 символів.

У таблиці 4.11 наведено результати експерименту для стегосистем з контейнером City.png.

Таблиця 4.11 – Робастність стегосистем, що використовують контейнер City.png до атак

	LSB	Kutter	Kox	Smith
Масштабування	-	-	-	-
Поворот	-	-	+	+
Фільтри Гауса	-	+	+	+
Контрастність	-	+	+	+
Підвищення яскравості	-	+	+	+
Зниження яскравості	-	+	+	+
Ерозія	-	-	+	+
Обрізання праворуч	+	+	-	-
Обрізання зліва	-	+	-	-
Обрізання знизу	+	+	+	-
Обрізання зверху	-	+	-	-

Продовження таблиці 4.11

Стиснення JPEG з коефіцієнтом якості 0,7	-	-	+	+
Стиснення JPEG з коефіцієнтом якості 0,5	-	-	+	+

Для робастності стегосистеми, що використовує алгоритм Куттера до фільтрації за допомогою функції Гауса, необхідно збільшити енергію сигналу ^{3} до 0,1.

Для стійкості стегосистеми, що ґрунтується на використанні алгоритму Коха до обрізання зображення праворуч, довжина повідомлення не повинна перевищувати 15 символів.

У таблиці 4.12 наведено результати експерименту для стегосистем з контейнером Sky.tiff.

Таблиця 4.12 – Робастність стегосистем, що використовують контейнер City.png до атак

	LSB	Kutter	Kox	Smith
Масштабування	-	-	-	-
Поворот	-	-	+	+
Фільтри Гауса	-	+	+	+
Контрастність	-	+	+	+
Підвищення яскравості	-	+	+	+
Зниження яскравості	-	+	+	+
Ерозія	-	-	+	+
Обрізання праворуч	-	+	-	-
Обрізання зліва	-	+	-	-

Продовження таблиці 4.12

Обрізання знизу	+	+	+	-
Обрізання зверху	-	+	-	-
Стиснення JPEG з коефіцієнтом якості 0,7	-	-	+	+
Стиснення JPEG з коефіцієнтом якості 0,5	-	-	+	+

Стегосистема, що використовує алгоритм вбудовування в найменш значний біт робастна для видалення частини зображення праворуч при довжині повідомлення не більше 15 символів.

Для стійкості стегосистеми, заснованої на використанні алгоритму Куттера, до фільтрації за допомогою функції Гауса та зменшення яскравості необхідно збільшити енергію сигналу ^{3} до 0,15. А для робастності до збільшення яскравості необхідно ще й збільшити кількість вбудовувань^{2} до 500.

Стегосистема, що використовує алгоритм Коха, стійка до масштабування зображення з довжиною повідомлення не більше 4 символів, до обрізання зображення праворуч – не більше 7 символів.

Підіб’ємо підсумки дослідження.

Найгірші результати показали стегоситеми, що використовують вбудовування у найменш значний біт. Ці системи стійкі лише до видалення областей зображення, у яких ведеться вбудовування (вбудовування ведеться згори донизу, ліворуч у кожен другий біт). Отже, алгоритм вбудовування в найменш значний біт не застосовується для побудови стегосистем при високій ймовірності проведення атак у стегоканалі. Даний алгоритм може бути успішно використаний для побудови крихких стегосистем.

Стегосистеми, засновані на використанні алгоритму Куттера, стійкі до застосування деяких фільтрів, зміни яскравості, обрізання зображення (за рахунок багаторазового дублювання інформації). Серйозними недоліками є руйнування ЦВЗ при стисканні JPEG (що дуже поширеною операцією) і поворотів зображення. Такі стегосистеми є напівкрихкими.

Стегосистеми, що використовують алгоритм Коха, робастні до повороту зображення (внаслідок блочності алгоритму), стиску JPEG, фільтрації зображення. Недоліком є послідовне вбудовування бітів повідомлення у кожен блок, що призводить до низької робастності до видалення частин зображень. Проте такі стегосистеми мають досить високу робастність і можуть застосовуватися для захисту авторських прав на зображення.

Стегосистеми, що використовують при вбудовуванні ЦВЗ алгоритм Сміта стійкі до поворотів зображення, стиску JPEG, використання фільтрів. Але такі стегосистеми зовсім не стійкі до обрізання зображення (ЦВЗ рознесений по всій площі зображення).

В результаті дослідження з’ясувалося, що найскладнішими для стегосистем атаками є масштабування та обрізання зображення. Розглянуті стегосистеми або зовсім не робастні до цих атак, або робастні при різкому зниженні пропускної здатності.

4.4 Дослідження часу роботи алгоритмів генерування та вбудови ЦВЗ у зображення

При дослідженні роботи будь-якого програмного алгоритму важливим критерієм його ефективності є час, що витрачається на роботу. Час є важливим обчислювальним ресурсом, а мінімізація витрат цього – важливим завданням для розробника алгоритму.

Виміряємо середній час роботи (в секундах) алгоритмів генерування та вбудови цифрових водяних знаків, реалізованих у програмному комплексі. Середній час роботи алгоритму визначатиметься як середнє арифметичне часів, що витрачаються на роботу при вбудовуванні у конкретне зображення.

Для дослідження вибрано 7 тестових зображень: Nure.jpg (фотографія-пейзаж), Tulips.jpg (фотографія-зображення кольорів), photo.jpg (фотографія невисокої чіткості, зроблена з аматорської фотокамери), Lena.bmp (точкове зображення-портрет), city.png (репродукція картини), logo.png (логотип, створений у графічному редакторі), Sky.tiff (космічний знімок, представлений у форматі TIFF). Причому час, що витрачається на генерування, вбудовування та вилучення ЦВЗ, розглядаються окремо.

Для об’єктивності дослідження вимірюватимемо час роботи кожного алгоритму для кожного контейнера по 10 разів і заноситимемо в таблицю середнє значення. Так як час роботи будь-якого алгоритму залежить від характеристик ПК, мови програмування та середовища розробки, в якому проводиться дослідження, необхідно вказати дані характеристики. У таблиці 4.13 дано відомості про процесор, оперативну пам’ять і середовище програмування.

Таблиця 4.13 – Характеристики ПК та середовище програмування

Процесор	Intel Core i5 660 3.33 GHz
Кількість оперативної пам’яті	8 Гб
Мова програмування	C++
Середовище виконання	Microsoft Visual Studio 2019

Для вимірювання часу використовуємо функцію clock бібліотеки time. Результати дослідження наведено у таблицях 4.14 – 4.17.

Таблиця 4.14 – Час роботи алгоритму вбудови у найменш значний біт

	Шифрування+ Генерування	Вбудовування	Вилучення+ Дешифрування	Загальне
Nure.jpg	0,427	1,362	1,036	2,825
Tulips.jpg	0,423	1,312	1,033	2,768
Photo.jpg	0,984	3,254	2,594	6,832
Lena.bmp	0,151	0,467	0,362	0,98
City.png	0,517	1,75	1,29	3,557
Logo.png	0,137	0,508	0,341	0,986
MilkyWay.tiff	0,185	0,546	0,415	1,146

Таблиця 4.15 – Час роботи алгоритму Куттера-Джордана-Боссена

	Шифрування+ Генерування	Вбудовування	Вилучення+ Дешифрування	Загальне
Nure.jpg	0,002	0,477	0,375	0,854
Tulips.jpg	0,002	0,186	0,083	0,271
Photo.jpg	0,001	0,558	0,386	0,945
Lena.bmp	0,002	0,28	0,283	0,565
City.png	0,002	0,499	0,302	0,803
Logo.png	0,002	0,276	0,243	0,521
MilkyWay.tiff	0,002	0,301	0,257	0,56
Середній час	0,002	0,368	0,275	0,645

Таблиця 4.16 – Час роботи алгоритму Коха-Жао

	Шифрування+ Генерування	Вбудовування	Вилучення+ Дешифрування	Загальне
Nure.jpg	8,093	6,53	7,843	22,466
Tulips.jpg	7,885	6,35	7,899	22,134
Photo.jpg	18,764	15,643	18,76	53,167
Lena.bmp	2,485	2,038	2,451	6,974
City.png	9,947	8,293	9,868	28,108
Logo.png	2,389	1,97	2,375	6,734
MilkyWay.tiff	3,166	2,677	3,12	8,963
Середній час	7,533	6,214	7,474	21,221

Таблиця 4.17 – Час роботи алгоритму Сміта-Коміська

	Шифрування+ Генерування	Вбудовування	Вилучення+ Дешифрування	Загальне
Nure.jpg	29,448	0,206	12,254	41,908
Tulips.jpg	29,6	0,19	12,345	42,135
Photo.jpg	71,225	0,455	29,943	101,623
Lena.bmp	10,204	0,056	4,137	14,397
City.png	38,178	0,336	15,069	53,583
Logo.png	8,882	0,094	3,696	12,672
MilkyWay.tiff	11,834	0,096	4,951	16,881
Середній час	28,481	0,205	11,771	40,457

За результатами дослідження найшвидшим алгоритмом став алгоритм Куттера-Джордана-Боссена (середній час – 0,645 секунд). Для жодного зображення час виконання не перевищив позначки в 1 секунду. Найкращий час досягається при використанні як контейнер зображення Tulips.jpg (0,271 секунди). Таке швидке виконання пов’язане з найменшою серед усіх зображень кількістю вбудованих. У цьому методі генерування ЦВЗ, що включає ще й генерування ключів, відбувається практично миттєво. Куди більше часу займає генерування координат та вбудовування у контейнер, а також вилучення цифрового водяного знака із контейнера-зображення.

Хороший результат був показаний алгоритмом вбудовування у найменш значний біт (середній час – 2,728 секунди). Час виконання цього алгоритму залежить лише від розміру зображення. Відставання за досліджуваним критерієм від алгоритму Куттера пов’язані з перебором всіх пікселів перед вбудовуванням і вилученням, і навіть з необхідністю декомпозиції пікселів для вбудовування найменш значний біт кожного пікселя.

Алгоритми Коха-Жао та Сміта-Коміски виявилися надто повільними. Середній час виконання алгоритму Коха-Жао перевищив 20 секунд (21,221 секунд). Найкращі часи були досягнуті при вбудовуванні у найменші за розміром зображення – Logo.png (6,734 секунд) та Lena.bmp (6,974 секунд). Така повільна робота алгоритму пов’язана з необхідністю двічі проводити дискретне косинусне перетворення і зворотне дискретне косинусне перетворення для всіх блоків зображення. Для прискорення роботи алгоритму при відомій довжині повідомлення можна здійснювати пряме і зворотне дискретне косинусне перетворення тільки для блоків, в які здійснюватиметься вбудовування інформації.

Алгоритм Сміта - Коміскі виявився найповільнішим (середній час - 40,457 секунд). На генерування ЦВЗ і вбудовування його в зображення Hockey.jpg, яке є найбільшим із тестових зображень, було витрачено більше 100 секунд (витяг ЦВЗ-близько 30 секунд), що породжує серйозні незручності при вилученні ЦВЗ із контейнера, особливо при вилученні повідомлення-посилання з ілюстрації у друкованому виданні.

ВИСНОВКИ

Проведене дослідження найбільш популярних алгоритмів генерування та вбудовування цифрових водяних знаків у зображення, проведене в рамках даної кваліфікаційної роботи, спрямоване на популяризацію технології ЦВЗ та об’єктивне детальне порівняння обраних алгоритмів. Як критерії порівняння були обрані: візуальна помітність вбудовування (включаючи залежність візуальної помітності та параметрів алгоритмів), пропускна та прихована пропускна здатність стегосистем, робастність до різних видів атак (було розглянуто 13 різних атак) та час роботи алгоритмів. Було створено програмний комплекс, що реалізує досліджувані алгоритми.

Алгоритм вбудовування в найменш значний біт має низьку обчислювальну складність, витрачає для роботи мінімальну кількість часу, має дуже велику приховану пропускну здатність і вносить мінімальні візуальні спотворення при вбудовуванні. Головним і дуже серйозним недоліком є його нестійкість практично всіх видів атак. Ця обставина ускладнює його використання у стегоканалах із високою ймовірністю проведення атак (наприклад, у сховищах в Інтернеті). Доцільно застосування цього алгоритму при вбудовуванні інтерактивної інформації зображення, розташовані в друкованої продукції. Такі зображення найчастіше виходять за межі друкарні тільки у вигляді ілюстрації на папері і надалі ЦВЗ може бути лише лічений, а чи не змінений.

Алгоритм Куттера-Джордана-Боссена є найшвидшим алгоритмом з усіх розглянутих, має прийнятну пропускну здатність, робастен до більшості атак. Його головний недолік - необхідність грамотного підбору контейнера для візуальної непомітності вбудовування. Даний алгоритм може ефективно застосовуватися для вирішення різних завдань, але перед публікацією захищеного зображення необхідно переконатися у відсутності помітних спотворень зображення.

Алгоритм Коха-Жао боязкий до багатьох атак і має непогану пропускну здатність. Алгоритм використовує дискретне косинусне перетворення, що робить ефективним його використання при вбудовуванні ЦВЗ зображення, стиснуті в JPEG. Саме JPEG є найпопулярнішим графічним форматом для представлення фотографій. Таким чином, алгоритм Коха-Жао добре застосовується для захисту фотографій та електронних репродукцій картин у форматі JPEG. Внаслідок обчислювальної складності ДКП алгоритм споживає для роботи серйозні тимчасові ресурси, що робить практично неможливим його використання в системах детектування ЦВЗ реального часу (наприклад, при вбудовуванні в друкарську продукцію).

Алгоритм Сміта-Коміски також має високу робастність, трохи спотворює вихідний контейнер, ефективність його роботи не залежить від формату зображення. Але висока обчислювальна складність, споживання великої кількості пам’яті, низька пропускна спроможність та тривалий час роботи значно звужує сферу його застосування. Використання даного алгоритму доцільно при вбудовуванні дуже невеликого ЦВЗ у зображення (наприклад, прізвища та ініціалів автора зображення) у системах, що не потребують швидкого виявлення та декодування ЦВЗ.

У найближчому майбутньому все більш популярними будуть ставати методи стиснення зображень, що базуються на дискретному вейвлет-перетворенні, з’являться ефективні фрактальні методи. І на зміну алгоритмам генерування і вбудовування цифрових водяних знаків, що активно використовуються сьогодні, прийдуть нові алгоритми, що використовують ці перетворення. Також створено достатню кількість нелінійних алгоритмів, що використовують квантування. Вже зараз вони демонструють високу приховану пропускну здатність і робастність до різних атак, проте поки що активно не застосовуються. Але це питання часу. Також, безсумнівно, з’являться абсолютно нові технології створення ЦВЗ, які мають низьку обчислювальну складність, підвищену робастність, універсальність, низьку візуальну помітність

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. КОРОЛЬОВ, В. Ю., et al. Планування досліджень методів стеганографії та стегоаналізу. ВІСНИК, 2011, 2011187.
2. Рева, А. К. (2019). Використання цифрового водяного знаку для захисту фотознімків з місця надзвичайної ситуації. наука про цивільний захист як шлях становлення молодих вчених SCIENCE ON CIVIL PROTECTION AS A WAY OF BECOMING YOUNG SCIENTISTS, 9, 169.
3. McMahon, Kris. "The current state of digitized images necessitates congressional action to protect authors and content providers from online infringement." Suffolk UL Rev. 49 (2016): 469.
4. Кузнецов, О. О., Євсеєв, С. П., Король, О. Г., Кузнецов, А. А., & Евсеев, С. П. (2011). Стеганографія: навчальний посібник.
5. Стасюк, О. І., Гнатюк, С. О., Довгич, Н. І., & Літош, М. С. (2011). Сучасні стеганографічні методи захисту інформації. Захист інформації, 13(1 (50)).
6. Marcellin, M. W., Gormish, M. J., Bilgin, A., & Boliek, M. P. (2000, March). An overview of JPEG-2000. In Proceedings DCC 2000. Data Compression Conference (pp. 523-541). IEEE.
7. Шелест, М. Є., and В. І. Андреєв. "Комп’ютерна стеганографія та її можливості." Сучасна спеціальна техніка 1 (2011): 97-104.
8. Kutter, Martin, Frederic D. Jordan, and Frank Bossen. "Digital watermarking of color images using amplitude modulation." Journal of Electronic imaging 7.2 (1998): 326-332.
9. Bender, Walter R., Daniel Gruhl, and Norishige Morimoto. "Techniques for data hiding." Storage and Retrieval for Image and Video Databases III. Vol. 2420. International Society for Optics and Photonics, 1995.
10. Tirkel, A. Z., van Schyndel, R. G., & Osborne, C. F. (1995, December). A two-dimensional digital watermark. In Dicta (Vol. 95, No. 7, pp. 5-8).

11. Chandramouli, Rajarathnam. "A mathematical framework for active steganalysis." Multimedia systems 9.3 (2003): 303-311.

12. Qianli, Yang, and Cai Yanhong. "A digital image watermarking algorithm based on discrete wavelet transform and discrete cosine transform." 2012 International Symposium on Information Technologies in Medicine and Education. Vol. 2. IEEE, 2012.

13. Chae J. Robust Techniques for Data Hiding in Images and Video / J. Chae / PhD thesis, Department for Electrical and Computer Engineering, University of California, Santa Barbara, CA, USA, 1999. - P.202.

14. Bas P., Chassery J.-M., Davoine F. Geometrical and frequential watermarking scheme using similarities // In SPIE Conference on Security and Watermarking of Multimedia Contents. 1999. №3657. P. 264-272

15. Moulin P., O'Sullivan J. Information-theoretic analysis of information hiding // Computer engineering / 1999. - № 43. - P.3.

16. Бологова Н. Н., Дячкин Н. А., Костенков П. Ю. Аутентификация изображения с помощью цифровых подписей на основе содержимого изображения V Всеукраїнська мультидисциплінарна науково-практична Інтернет-конференція 26 листопада 2021, Україна, Одеса с. 34-37