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

Касаяні Андрій Васильович,
студент групи РФ-181

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

Підвищення ефективності стеганографічного алгоритму, заснованого на сингулярному розкладі блоків матриці цифрового зображення

Спеціальність:
122 Комп’ютерні науки

Спеціалізація, освітня програма:
Програмне забезпечення систем захисту інформації

Керівник:
Трифонова Катерина Олексіївна,
ст. викладач

Одеса – 2022

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

Рівень вищої освіти: перший (бакалаврський)
Спеціальність 122 – Комп'ютерні науки
Освітня програма – Програмне забезпечення систем захисту інформації

ЗАТВЕРДЖУЮ
Завідувач кафедри КБПЗ
д.т.н.,проф. А.А. Кобозєва
_____
«____»_____20__р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ
Касаяні Андрію Васильовичу

1. Тема роботи: *Підвищення ефективності стеганографічного алгоритму, заснованого на сингулярному розкладі блоків матриці цифрового зображення*,
керівник роботи: *ст. викл. Трифонова Катерина Олексіївна,*
затверджені наказом ректора № 168-в від 17.05.2022р.
2. Зміст роботи: *огляд стеганографічних методів; теоретичні основи підвищення ефективності стеганографічного алгоритму, заснованому на сингулярному розкладі блоків зображення; практична розробка стеганографічного алгоритму*
3. Перелік ілюстративного матеріалу: *структурна схема стеганографічної системи; схема організації роботи програмного застосування*

4. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці	доц. Ярова І.А.		

5. Дата видачі завдання «____»_____ 20___ р.

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів кваліфікаційної роботи	Строк виконання	Примітка
1	Аналіз стеганоалгоритму, на основі сингулярного розкладу матриці	21.03.22	виконано
2	Розробка стеганографічного алгоритму	04.04.22	виконано
3	Реалізація стеганографічного алгоритму	18.04.22	виконано
4	Тестування програмного продукту	02.05.22	виконано
5	Підготовка пояснювальної записки	16.05.22	виконано
6	Підготовка презентації та доповіді	30.05.22	виконано
7	Попередній захист	03.06.22	виконано
8	Нормоконтроль, рецензування	13.06.22	виконано

Здобувач вищої освіти _____ Касаяні А.В.

Керівник роботи _____ Трифонова К.О.

ЗАВДАННЯ

на розробку розділу «Охорона праці»

Касаяні Андрію Васильовичу, група РФ181

Інститут інформаційної безпеки, радіоелектроніки та телекомунікацій

Кафедра кібербезпеки та програмного забезпечення

Тема роботи: *Підвищення ефективності стеганографічного алгоритму, заснованого на сингулярному розкладі блоків матриці цифрового зображення*

Зміст розділу:

1. Аналіз умов праці і вибір основних заходів виробничої безпеки.
2. Аналіз пожежної безпеки та вибір заходів і засобів пожежної безпеки.

Керівник роботи

_____ _____
(підпис) (прізвище та ініціали)

« __ » _____20 __ р.

Консультант з охорони праці

_____ _____
(підпис) (прізвище та ініціали)

« __ » _____20 __ р.

АНОТАЦІЯ

Кваліфікаційна робота на тему «Підвищення ефективності стеганографічного алгоритму, заснованого на сингулярному розкладі блоків матриці цифрового зображення» на здобуття першого (бакалаврського) рівня вищої освіти за спеціальністю 122 – Комп'ютерні науки, спеціалізація, освітня програма: Програмне забезпечення систем захисту інформації, містить: 15 рисунків, 2 таблиці, 1 додаток, 24 літературних джерела за переліком посилань. Робота виконана на 58 сторінках загального тексту і 43 сторінках основного тексту.

Метою даної кваліфікаційної роботи є підвищення ефективності стеганографічного алгоритму, заснованому на сингулярному розкладі блоків матриці цифрового зображення.

Теоретичною основою стеганографічного методу, який реалізований у роботі, є сингулярний розклад матриці зображення. Тестування початкового алгоритму та його версії із модифікацією, для визначення ефективності, відбувалося за допомогою середовища розробки MATLAB. Повна реалізація розробленої модифікації та функцій її аналізу виконана засобами обраного середовища MATLAB.

Результуючим продуктом виконання кваліфікаційної роботи є програмне забезпечення для приховування додаткової інформації у цифрових зображеннях, з можливістю подальшої передачі цього зображення та відновлення секретної інформації на стороні одержувача, для підвищення захисту інформації, що передається по відкритих каналах зв'язку.

ЦИФРОВЕ ЗОБРАЖЕННЯ, СИНГУЛЯРНИЙ РОЗКЛАД, СИНГУЛЯРНІ ЧИСЛА, СТЕГАНОГРАФІЯ, ЗАХИСТ ІНФОРМАЦІЇ

ABSTRACT

Qualification work «Efficiency improvement of the steganography algorithm based on the singular value decomposition of digital image matrix blocks» for the first level of higher education (bachelor) in the specialty 122 – Computer Science, specialization, educational program: Software for information security systems, contains: 15 figures, 2 tables, 1 appendix, 24 references according to the list of references. Work carried out on 58 total pages of text and 43 pages of main text.

The aim of this qualification work is to increase the efficiency of the steganography algorithm, based on the singular decomposition of digital image matrix blocks.

Theoretical basis of the steganography method, which is implemented in the work, is the singular decomposition of the image matrix. Testing of the initial algorithm and its version with modification, to determine efficiency, was performed using the MATLAB development environment. The full implementation of the developed modification and the functions of its analysis is performed by means of the selected MATLAB environment.

The result of the qualification work is software for hiding additional information in digital images, with the possibility of further transmission of this image and recovery of secret information on the part of the recipient, to increase the protection of information transmitted through open communication channels.

DIGITAL IMAGE, SINGULAR VALUE DECOMPOSITION, SINGULAR VALUES, STEGANOGRAPHY, INFORMATION SECURITY

ЗМІСТ

ВСТУП ……..……..……..……8
1 ПРИХОВУВАННЯ СЕКРЕТНОЇ ІНФОРМАЦІЇ У ЦИФРОВИХ ЗОБРАЖЕННЯХ У ФОРМАТІ ЗІ СТИСКОМ 11
1.1 Організація роботи стеганографічної системи 11
1.2 Стеганографічні методи для цифрових зображень у форматі зі стиском.... 14
1.3 Програмні реалізації стеганографічних методів для цифрових зображень у форматі зі стиском 16
2 ТЕОРЕТИЧНІ ОСНОВИ СТЕГАНОГРАФІЧНОГО АЛГОРИТМУ, ЗАСНОВАНОГО НА СИНГУЛЯРНОМУ РОЗКЛАДІ БЛОКІВ МАТРИЦІ ЦИФРОВОГО ЗОБРАЖЕННЯ 21
2.1 Стеганографічний алгоритм, заснований на сингулярному розкладі блоків матриці цифрового зображення..... 21
2.2 Підвищення ефективності стеганографічного алгоритму, заснованого на сингулярному розкладі блоків матриці цифрового зображення..... 23
2.3 Ефективність модифікованого стеганографічного алгоритму..... 28
3 ПРАКТИЧНА РЕАЛІЗАЦІЯ СТЕГАНОГРАФІЧНОГО АЛГОРИТМУ, ЗАСНОВАНОГО НА СИНГУЛЯРНОМУ РОЗКЛАДІ БЛОКІВ МАТРИЦІ ЦИФРОВОГО ЗОБРАЖЕННЯ 32
3.1 Обґрунтування вибору середовища розробки програмного застосування.. 32
3.2 Організація роботи програмного застосування 34
3.3 Інструкція з використання програмного застосування..... 36
4 ОХОРОНА ПРАЦІ 41
ВИСНОВКИ..... 48
ПЕРЕЛІК ПОСИЛАНЬ..... 50
Додаток А Лістинг програмного коду 52

ВСТУП

Стрімкий розвиток інформаційних технологій у наш час призвів до того, що більша частина інформації створюється, зберігається та обробляється у цифровому вигляді. Питання обмеження доступу до такого роду інформації стає дедалі важливим. Захист конференційної та приватної інформації відбувається за допомогою різноманітних засобів, серед яких варто виділити стеганографію. Задачею стеганографії є збереження факту секретної передачі інформації, що може бути застосовано у багатьох, а може й в усіх, сферах людської діяльності.

Актуальність вдосконалення існуючих стеганографічних методів на ряду з розробкою нових є надзвичайно великою, бо у всі часи захист певного роду приватної інформації від несанкціонованого доступу, був передовою задачею.

Ще одним засобом захисту сучасної цифрової інформації є криптографія. Криптографія є досить надійною, з точки зору захищеності зашифрованої за допомогою неї інформації. Але вона, все ж таки, не скриває факт захисту певної інформації, що у деякому ступені є її вразливістю. Стеганографія, у свою чергу, вирішує проблему секретності захисту інформації, що робить її незамінною для вирішення певного роду задач.

Метою даної кваліфікаційної роботи є підвищення ефективності стеганографічного алгоритму, заснованому на сингулярному розкладі блоків матриці цифрового зображення. Незважаючи на те, що поданий алгоритм має високу надійність сприйняття, відновлення додаткової інформації цим алгоритмом супроводжується великою кількістю помилок. Вирішення саме цієї проблеми мається на увазі, під підвищенням ефективності стеганоалгоритму, що розглядається.

Для реалізації такого роду модифікації, з підвищення ефективності, було поставлено ряд наступних задач:

а) провести аналіз предметної області – розглянути існуючі стеганографічні методи приховування інформації у цифрових зображеннях;

б) визначити причину низького показника ефективності стеганоалгоритму;

в) виходячи з визначеної причини розробити модифікацію, яка суттєво підвищить ефективність стеганоалгоритму;

г) розробити програмний продукт на основі модифікованого алгоритму, заснованому на сингулярному розкладі блоків матриці.

У якості об’єкта дослідження виступає процес приховування та відновлення інформації у цифровому зображені.

Предметом дослідження є вплив занурення додаткової інформації на сингулярні числа блоків матриці зображення в результаті роботи стеганографічного алгоритму, що розглядається у цій кваліфікаційній роботі.

Теоретичною основою стеганографічного методу, який реалізований у роботі є сингулярний розклад матриці зображення. Тестування початкового алгоритму та його версії із модифікацією, для визначення ефективності, відбувалося за допомогою середовища розробки MATLAB. Повна реалізація розробленої модифікації та функцій її аналізу виконана засобами обраного середовища MATLAB.

Результуючим продуктом виконання кваліфікаційної роботи є програмне забезпечення для приховування додаткової інформації у цифрових зображеннях, з можливістю подальшої передачі цього зображення та відновлення секретної інформації на стороні одержувача, для підвищення захисту інформації, що передається по відкритих каналах зв’язку.

Дана робота має таку структуру: вступ, чотири розділи, перелік посилань, висновки та додаток.

У вступі розкривається сутність і стан обраної предметної області та приводяться обґрунтування вибору теми роботи та її актуальності. Також у вступі поставлена основа мета роботи та задачі, які потрібні для досягнення цієї мети.

У першому розділі розглядаються основні положення, які стосуються стеганосистем та їх побудови. Надано визначення поняття стеганосистеми, приведено вимоги, щодо її побудови, та визначено сфери використання стеганосистем у сучасному світі. Представлена загальна структурна схема

стеганосистеми з детальним описом її складових. Також виконаний аналіз існуючих стеганографічних методів та побудована характеристика відносно їх недоліків та переваг. Окрім стеганографічних методів у розділі розглядаються конкретні реалізації стеганоалгоритмів, у вигляді програмних застосувань, що здатні працювати з зображенням у форматі з втратами.

Другий розділ присвячений розгляду алгоритму, заснованому на сингулярному розкладі блоків матриці цифрового зображення. Було розроблено модифікацію, яка підвищує ефективність вище зазначеного алгоритму. Для підтвердження більшої ефективності алгоритму із модифікацією у порівнянні з початковим алгоритмом, наведене їх тестування з визначенням основних параметрів ефективності, серед яких: відсоток відновлення та пікове відношення сигнал-шум.

Третій розділ надає інформацію про засоби реалізації програмного продукту, які використовувалися для побудови кінцевого програмного забезпечення. У розділі обґрунтований вибір MATLAB, у якості такого засобу, шляхом виявлення його переваг серед конкурентів. Також описаний головний інструмент середовища MATLAB з точки зору роботи із зображеннями – Image Processing ToolBox, який був використаний для реалізації стеганоалгоритму, що розглядається.

У четвертому розділі був проведений аналіз робочого місця програміста, у результаті якого виявлений ряд шкідливих виробничих факторів. До кожного фактору, згідно до відповідних державних документів, надані нормативні показники. На основі порівняння цих показників з їх фактичними значеннями, надані поради та рекомендації, щодо покращення умов праці на робочому місці програміста. Також у розділі проведений аналіз пожежної безпеки та наведені засоби підвищення пожежної безпеки робочого приміщення, яке розглядається.

Результати роботи опубліковані в збірнику наукових праць «Актуальні наукові дослідження в сучасному світі».

1 ПРИХОВУВАННЯ СЕКРЕТНОЇ ІНФОРМАЦІЇ У ЦИФРОВИХ ЗОБРАЖЕННЯХ

1.1 Організація роботи стеганографічної системи

У сучасному світі інформація є одним з найбільш цінних ресурсів. У зв’язку із стрімким розвитком технологій більша частина інформації набула цифрового вигляду, та разом з тим її кількість і доступність стала набагато більшою. Тому питання захисту інформації від несанкціонованого доступу стає надзвичайно актуальним.

Для того щоб захистити інформацію від стороннього небажаного зору та впливу, вчені усього світу вдалися до розробки різних методів та засобів боротьби із несанкціонованим доступом, серед яких – криптографія та стеганографія. Засобом захисту інформації у криптографії є шифрування, тобто перетворення інформації у деякий вигляд, у якому вона не представляє колишньої цінності. На відміну від криптографії, стеганографія спрямована на приховування факту наявності та передачі секретної інформації.

Стеганосистема (скорочено від стеганографічна система) – це система, яка є основою для створення секретного каналу зв’язку та передачі інформації через цей канал [1-3]. Криптографічне шифрування бере за мету обмежити доступ до власне контейнеру, коли у стеганографічній системі метою є забезпечення секретності факту передачі вбудованої інформації, що захищається, та можливості її відновлення з та без наявності будь-яких спотворень.

Для побудови надійної стеганосистеми слід дотримуватися таких вимог [1]:

- має бути забезпечена необхідна пропускна спроможність;
- безпека системи повинна повністю визначатися таємністю ключа: порушник може повністю знати всі алгоритми роботи стеганосистеми й статистичні характеристики множин повідомлень і контейнерів, але це не дасть йому ніякої додаткової інформації про наявність або відсутність повідомлення в даному контейнері;
- знання порушником факту наявності повідомлення в якому-небудь

контейнері не повинно допомогти йому при виявленні повідомлень в інших контейнерах;

– занурення повідомлення у контейнер не має призводити до його видимих спотворень;
– обчислювальна складність стеганоалгоритму має бути прийнятою;
– при відсутності повідомлення у більшості випадків стеганосистема не має його відновлювати.

У наш час стеганосистеми активно використовуються у багатьох сферах людської діяльності для вирішення таких задач [4-6]:

– прихована передача даних;
– приховане зберігання інформації;
– захист авторського права;
– захист справжності документів;
– підтвердження достовірності переданої інформації;
– стеганографічне відстежування.

На рисунку 1.1 відображено схематичне представлення стеганосистеми [1].

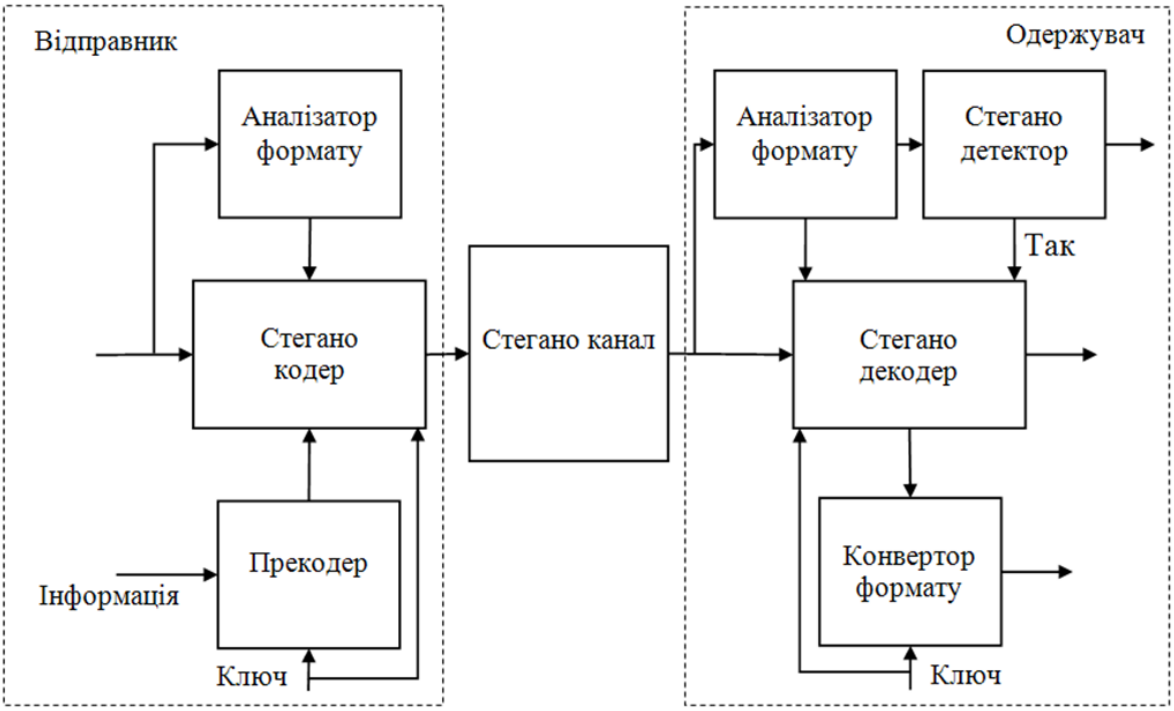


Рисунок 1.1 – Структурна схема стеганосистеми

Основними поняттями, якими оперують у стеганографії є повідомлення та контейнер. Повідомленням називають інформацію у певному представленні (наприклад текст, зображення або аудіо), яка має бути таємно передана. Контейнер також представляє собою деяку інформацію, але, на відміну від повідомлення, ця інформація не є секретною і використовується як об’єкт для впровадження повідомлення.

Стеганосистема, як і будь-яка інша система, має свою структуру, яка є фундаментом її побудови. Структурна схема стеганосистеми складається з таких основних елементів:

- прекодер;
- стеганокодер;
- аналізатор формату;
- стеганоканал;
- стеганодетектор;
- стеганодекодер;
- конвертер формату.

Довільний вигляд додаткової інформації (повідомлення) не завжди підходить для впровадження у контейнер, тому перед впровадженням, прекодер здійснює деяке перетворення додаткової інформації у певний вигляд, який є зручним та підходящим для впровадження.

Після перетворення, яке здійснене прекодером, додаткова інформація власне поміщається стеганокодером у пустий контейнер (контейнер, який ще не вмістить у собі ніякої додаткової інформації), який враховує особливості контейнеру та його моделі.

У більшості стеганосистем для підвищення надійності стеганоалгоритму на етапі впровадження та відновлення додаткової інформації використовують секретний ключ.

Стеганоканал – це канал через який передається заповнений додатковою інформацією контейнер. Стеганоканал може піддаватися сторонньому впливу, атакам або поміхам.

Завданням з’ясування наявності додаткової інформації у контейнері займається стеганодетектор. Саме стеганодетектор каже про наявність або відсутність повідомлення у контейнері.

Наступним кроком після виявлення наявності додаткової інформації є її відновлення. Відновлення інформації відбувається стеганодекодером. Слід зазначити, що у деяких стеганосистемах стеганодекодер може бути відсутнім.

1.2 Стеганографічні методи для цифрових зображень у форматі зі стиском

Застосовуючи стеганографію для приховування секретної інформації у цифрових зображеннях, неможливо не зіткнутися із ситуацією, коли в якості контейнера виступає зображення у форматі з втратами – JPEG. Розповсюдженість даного формату із втратами обумовлена його перевагами у вигляді економії місця у пам’яті електронної техніки, за рахунок стиску зображення та відповідно зменшенням його розмірів. Одним з найпоширеніших видів атак на стеганографічні системи є атака стиском. Атака стиском призводить до труднощів у відновленні зануреної секретної інформації. Для вирішення проблеми ефективності стеганографічних методів з використанням зображень, у найбільш розповсюдженому форматі JPEG, люди з усього світу почали створювати нові стеганографічні алгоритми та вдосконалювати вже існуючі.

Розглянемо більш детально існуючі алгоритми, які здатні працювати із зображеннями у форматі з втратами.

Перший у черзі алгоритм на основі дискретного косинусного перетворення [7]. Цей метод використовує коефіцієнти дискретно косинусного перетворення (скорочено ДКП) алгоритму JPEG для впровадження додаткової інформації. Ліва верхня частина матриці коефіцієнтів ДКП містить низькочастотні коефіцієнти, які зменшуються у напрямку руху до правої нижньої частини. Виходячи з того, що низькочастотна область ДКП може з високою точністю представляти інформацію про корельоване зображення, у даному алгоритмі секретна інформація вбудовується у середню та високочастотну область з мінімальним впливом на

надійність сприйняття стеганоповідомлення. Також в цьому методі розроблено методологію Global-Adaptive-Region (GAR), яка адаптує розмір області ДКП до кореляційних властивостей області зображення представленої конкретним блоком ДКП.

Стеганоалгоритм приховування повідомлень у JPEG зображеннях, заснований на новому методі ДКП-М3 [8]. Розроблений алгоритм пропонує новий спосіб вбудовування секретного повідомлення, який націлений на зменшення спотворення властивостей зображення, для збільшення надійності збереження секретності передачі. Перед вбудовуванням повідомлення стискається, а далі, шляхом зміни різниці модулів коефіцієнтів ДКП, здійснюється впровадження цього повідомлення. Даний алгоритм може впроваджувати до 54 біт додаткової інформації в один блок ДКП.

Наступним розглянемо алгоритм з високою пропускною спроможністю, заснований на вейвлет-перетворені [9]. Власне вейвлет-перетворення – це інтегральне перетворення, яке переводить сигнал із часового представлення у частотно-часове. Для застосування вейвлет-перетворення у цьому алгоритмі використовуються вейвлет-фільтри, серед яких: Daubechies, Coiflets, Biorthogonal, Reverse Biorthogonal. Робота стеганографічного методу в області вейвлет-перетворення обумовлена рядом переваг, серед яких: отримання високої пропускної спроможності та високої надійності функцій системи приховування.

Ще один алгоритм, який застосовує вейвлет-перетворення – це стеганоалгоритм, заснований на цілочисельному вейвлет-перетворені [10]. Цей алгоритм використовує коефіцієнти вейвлет-перетворення для впровадження додаткової інформації у чотири підгрупи двовимірного вейвлет-перетворення. Цілочисельним цей різновид вейвлет-перетворення називається через те, що він вирішує проблеми, які виникають при роботі з числами із плаваючою крапкою. Також, цей метод використовує генетичний алгоритм для того щоб знайти функцію відображення для кожного блоку зображення. Хромосоми генетичного алгоритму кодуються як масив з 64 генів, які вказують на номера пікселів у кожному з блоків.

Для підведення підсумків, у таблиці 1.1. були приведені методи, які використовуються вищеперерахованими стеганографічними алгоритмами та наведена стисла характеристика, щодо переваг та недоліків кожного з них.

Таблиця 1.1 – Характеристика методів, які використовуються стеганоалгоритмами за їх перевагами та недоліками

Автори	Метод	Переваги	Недоліки
Rabie T., Kamel I.	Дискретне косинусне перетворення у глобальній адаптивній області	висока пропускна спроможність при збережені прийнятної надійності сприйняття;	низький рівень безпеки.
Attaby A., Ahmed M., Alsammak A.	Різниця модулів коефіцієнтів дискретного косинусного перетворення	висока пропускна спроможність;	низька надійність збереження секретності.
Divya V., Sasirekha N.	Вейвлет-перетворення	високий рівень безпеки; висока надійність сприйняття	низька якість зображення.
Miri A., Faez K.	Цілочисельне вейвлет-перетворення з використанням генетичного алгоритму	висока надійність сприйняття;	низький рівень помилок при відновлені; вразливість до атак стиском та геометричних атак.

1.3 Програмні реалізації стеганографічних методів для цифрових зображень у форматі зі стиском

На сьогоднішній день існує багато реалізацій стеганографічних алгоритмів у вигляді десктопних та веб-застосувань. У мережі Інтернет у відкритому доступі вдалося знайти багато різноманітних програм, які реалізують стеганографічні методи занурення та відновлення додаткової інформації. У результаті проведення

аналізу знайдених реалізацій, були обрані такі десктопні застосування та веб-сервіси: Steg, JPHS, Steganography Online Codec. Вибір саме цих програм обумовлений їх схожістю за своїм функціоналом з програмним застосуванням, який був розроблений у цій кваліфікаційній роботі.

Десктопне застосування Steg [11]. Інтерфейс десктопного застосування Steg представлено на рисунку 1.2.

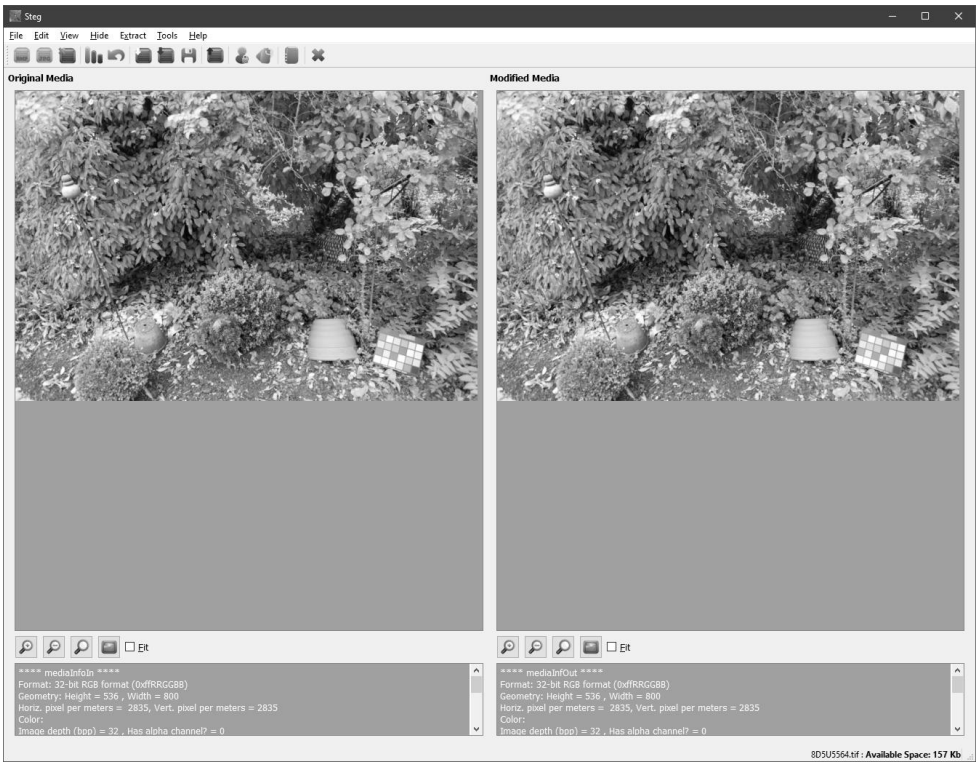


Рисунок 1.2 – Інтерфейс застосування Steg

Після завантаження вхідного зображення, Steg проводить його аналіз та надає інформацію про потенційний простір, доступний для приховування додаткові інформації. Цікавою особливістю програмного забезпечення Steg є використання криптографії у наступних режимах роботи програми: автоматичний, симетричний, асиметричний непідписаний, асиметричний підписаний. Автоматичний режим являє собою занурення та відновлення додаткової інформації без використання ключів та паролів. Режим у якому використовується один пароль, як для вбудови так і для відновлення інформації – симетричний. При роботі з асиметричним не підписаним режимом, при

відправлені використовується лише відкритий ключ одержувача, а для отримання – власний закритий ключ. Останній асиметричний підписаний режим – при вбудовуванні потребує відкритий ключ одержувача та власний закритий ключ, а для відновлення – відкритий ключ відправника та власний закритий ключ.

Десктопне застосування JPHS [12]. Інтерфейс десктопного застосування JPHS представлено на рисунку 1.3.

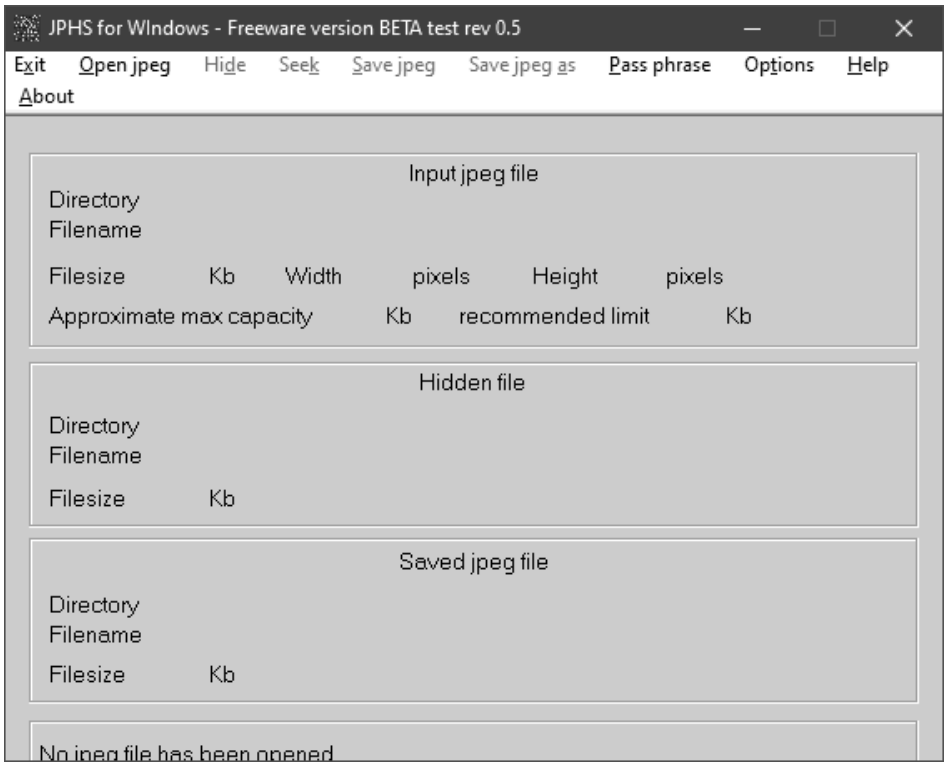


Рисунок 1.3 – Інтерфейс застосування JPHS

У якості стеганографічного методу, JPHS використовується перезапис найменших біт коефіцієнтів дискретного косинусного перетворення алгоритму JPEG. Модуль занурення програми, який має назву JPHIDE, за допомогою криптоалгоритму Blowfish виконує шифрування додаткової інформації, та обирає, де саме зберігати біти додаткової інформації випадковим чином. Таке розподілення дозволяє уникнути ситуації, коли стеганоповідомлення сильно статистично відрізняється від початкового зображення, через що прихований файл можна легко відновити. Важливо відмітити, що програма JPHS дозволяє зберігати зображення у форматі з втратами.

Steganography Online Codec – веб-застосування для реалізації стеганографічного вбудовування та вилучення секретної інформації для цифрового зображення, з додатковим забезпеченням захисту секретного повідомлення [13].

Перед вбудовуванням секретне повідомлення стискається для зменшення його розмірів за допомогою алгоритму DEFLATE. Виконується генерування криптографічного ключа для заданого пароля на основі алгоритму PBKDF2. Стисле секретне повідомлення шифрується за допомогою отриманого ключа на основі алгоритму AES. Отримані дані вбудовуються в обране цифрове зображення у найменші біти RGB компонент. Зберігання отриманого зображення відбувається лише у форматі без втрат.

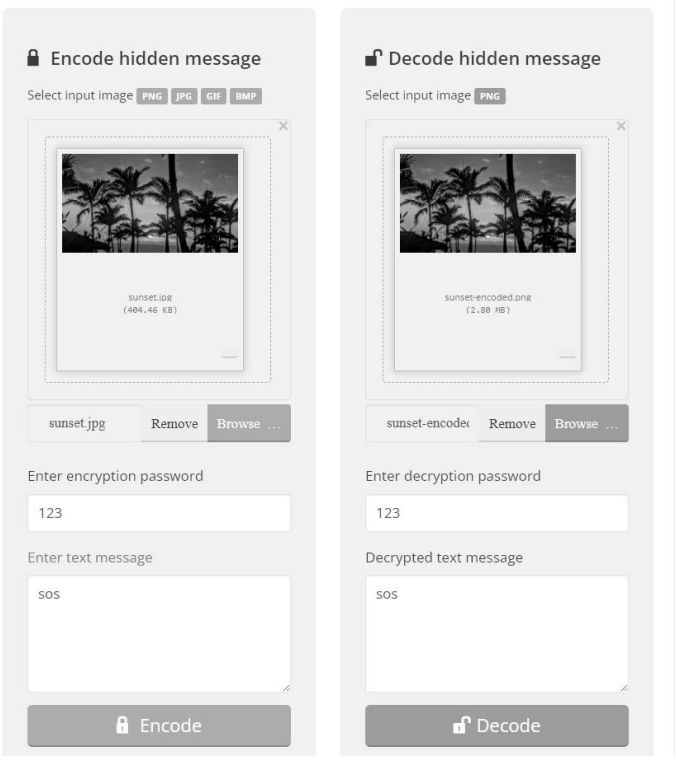


Рисунок 1.4 – Інтерфейс веб-сервісу Steganography Online Codec

В даному розділі описано приховування секретної інформації у цифрових зображеннях, розглянута мета стеганографії та її актуальність на теперішній час. Було надано визначення поняттю стеганографічної системи, докладно розібрано її характерні риси та основні структурні елементи з описанням кожного з них.

Представлено аналіз існуючих стеганографічних методів, таких як: дискретне косинусне перетворення у глобальній адаптивній області, різниця модулів коефіцієнтів дискретного косинусного перетворення, вейвлет-перетворення та цілочисельне вейвлет-перетворення з використанням генетичного алгоритму. У результаті такого аналізу представлена характеристика цих методів, щодо їх сильних та слабких сторін. Було проведено огляд практичних реалізацій стеганоалгоритмів, що працюють з зображеннями у форматі з втратами. Розглянуті програмні продукти: Steg, JPHS, Steganography Online Codec.

2 ТЕОРЕТИЧНІ ОСНОВИ СТЕГАНОГРАФІЧНОГО АЛГОРИТМУ, ЗАСНОВАНОГО НА СИНГУЛЯРНОМУ РОЗКЛАДІ БЛОКІВ МАТРИЦІ ЦИФРОВОГО ЗОБРАЖЕННЯ

2.1 Стеганографічний алгоритм, заснований на сингулярному розкладі блоків матриці цифрового зображення

В роботі [14] представлена розробка нового стеганографічного алгоритму, який заснований на сингулярному розкладі блоків матриці зображення. Запропонований алгоритм, у якості переваги над іншими стеганоалгоритмами, має високу надійність сприйняття вихідного стеганоповідомлення. Збереження надійності сприйняття є дуже важливою характеристикою для будь-якої стеганосистеми, бо вона безпосередньо впливає на головну задачу стеганографії – збереження факту секретності передачі інформації.

Теоретичну основу даного алгоритму складає сингулярний розклад матриці, який має такий вигляд [14]:

$$A=U\Sigma V^T, \tag{2.1}$$

де A – матриця контейнер;

U – матриця лівих сингулярних векторів;

V – матриця правих сингулярних векторів;

Σ – діагональна матриця сингулярних чисел.

Одною з найважливіших характеристик стеганографічного алгоритму є пропускна спроможність. Під пропускною спроможністю розуміють кількість інформації, яку можна передати через канал за одне його використання.

Авторами роботи [14], для підвищення пропускної спроможності, запропоновано використовувати наступну стратегію: матриця цифрового зображення представляється у вигляді блоків. Кожен такий блок призначений для вбудовування одного біту інформації.

Вбудовування секретної інформації відбувається безпосередньо у матрицю сингулярних чисел, шляхом збурення сингулярних чисел в залежності від значення біту. Такий процес вбудовування додаткової інформації, у загальному

випадку, не порушує надійність сприйняття отриманого стеганоповідомлення, як зазначається авторами роботи [14].

Одним з найпоширеніших видів атак на стеганографічні системи, робота над якими продовжується і на даний час, є атака стиском. В роботі [14] зазначається, що запропонований стеганографічний алгоритм, заснований на модифікації сингулярних чисел, є стійким до атаки стиском. Завдяки цьому, цей метод набуває переваги перед іншими методами.

Алгоритмічна реалізація такого методу складається з декількох основних кроків. Цифрове зображення зчитується та розбивається на три матриці основних кольорів схеми RGB. Для вбудовування додаткової інформації обирається складова B, тобто складова, яка відповідає за синій колір у зображенні.

Додатковою інформацією для вбудовування в даному алгоритмі є бітова послідовність, тобто набір нулів та одиниць. Таким чином, інформація у будь-якому вигляді завжди може бути представлена у бінарному форматі, що надає можливість занурення цієї інформації у зображення стеганографічним методом, що розглядається.

Для кожного отриманого блоку 8 на 8 будується сингулярний розклад. В залежності від значення біту додаткової інформації, обирається відповідна формула для вбудовування у найбільше сингулярне число.

Наступний крок – повернення у просторову область. Для цього використовується формула сингулярного розкладу для відновлення блоку матриці, яка вже зазнала збурень.

Процес вилучення додаткової інформації починається з розбиття матриці синього кольору на блоки 8 на 8. Для кожного блоку виконується побудова сингулярного розкладу, та за встановленою формулою, виконується вилучення додаткової інформації.

Узагальнені кроки стеганоалгоритму, що використовує сингулярний розклад матриці контейнера, представлені нижче [14].

Вбудовування додаткової інформації:

а) розбиття матриці на блоки 8 на 8; кожен блок використовується для

вбудовування одного біту додаткової інформації;

б) для кожного отриманого блоку будується сингулярний розклад (2.1);

в) якщо біт додаткової інформації дорівнює нулю, то:

$$\bar{\sigma}_1 = roundn(\sigma_1, K) + \frac{1}{4} \cdot \sigma_2, \quad (2.2)$$

де σ_1 – найбільше (перше) сингулярне число;

σ_2 – друге сингулярне число;

$\bar{\sigma}_1$ – найбільше (перше) сингулярне число після збурення;

K – розряд округлення;

roundn() – функція округлення до розряду K.

г) якщо біт додаткової інформації дорівнює одиниці, то:

$$\bar{\sigma}_1 = roundn(\sigma_1, K) + \frac{3}{4} \cdot \sigma_2. \quad (2.3)$$

д) використання сингулярного розкладу для повернення в просторову область:

$$\bar{A} = U\bar{\Sigma}V^T, \quad (2.4)$$

де $\bar{A}$ – блок матриці зображення зі збуренням;

$\bar{\Sigma}$ – діагональна матриця сингулярних чисел із збуренням.

Вилучення додаткової інформації:

а) розбиття матриці стеганоповідомлення на блоки 8 на 8;

б) для кожного отриманого блоку будується сингулярний розклад (2.1);

в) якщо вірною є нерівність:

$$\overline{\overline{(\sigma_1}} - roundn(\bar{\bar{\sigma}}_1, K)) < \frac{1}{2} \cdot \bar{\sigma}_2, \quad (2.5)$$

тоді відновлюємо 0, в іншому випадку – 1.

2.2 Підвищення ефективності стеганографічного алгоритму, заснованого на сингулярному розкладі блоків матриці цифрового зображення

Алгоритм, який розглядається у цій кваліфікаційній роботі, має серйозний недолік, який полягає у достатньо низькому відсотку правильно відновленого секретного повідомлення з цифрового зображення. Для того щоб розібратися з

причиною такого низького відсотку, розглянемо роботу стеганографічного алгоритму на конкретному прикладі.

У якості зображення, в яке буде занурюватися додаткова інформація, візьмемо звичайне цифрове зображення (рис. 2.1).

Рисунок 2.1 – Тестове цифрове зображення

Обране зображення, як можна побачити, має велику кількість деталей та достатньо велику контрастність, що відіграє важливу роль у тестуванні алгоритму.

Почнемо тестування з впровадження додаткової інформації. Отримане, після впровадження, стеганоповідомлення використовуємо для відновлення додаткової інформації. Порівняємо початковий вигляд додаткової інформації із тим, що вдалося відновити (рис. 2.2).

Початкова секретна інформація:

1	1	1	0	0	1	1	1	1	0	1	1	1	1	1	1	1	0	0	1	1	…

Відновлена секретна інформація

0	0	1	0	0	1	1	0	0	0	1	0	1	0	1	0	0	0	0	1	1	…

Рисунок 2.2 – Порівняння зануреної та відновленої бітових послідовностей

В результаті порівняння початкової та відновленої секретної інформації, можна зробити висновок про те, що інформація відновилась не повністю вірно. Підрахувавши кількість правильно відновлених бітів інформації та їхню загальну кількість, отримуємо відсоток відновлення, який складає *57%*.

Розглянемо приклад успішного відновлення біту додаткової інформації. Коли $\sigma_1 = 168.57$, а $\sigma_2 = 114.2$, отримуємо таке збурення для σ_1 $\bar{\sigma}_1 = 100 + 114.2 * 0.75 = 185.65$ при впровадженні одиниці.

При відновленні цього біту за формулою маємо: $185.65 - 100 > 0.5 * 114.2$, тобто $85.65 > 57.1$– тому відновлюється одиниця.

У цьому прикладі відновлення зрозуміло, що алгоритм працює саме так, як повинен працювати. Інформація відновлюється коректно. Але ж відсоток відновлення свідчить про те, що алгоритм має проблему, яка потребує вирішення.

Для того щоб зрозуміти причину не вірного відновлення бітів, розглянемо приклад саме такого біту.

Коли $\sigma_1 = 954$, а $\sigma_2 = 147$, отримаємо таке збурення для σ_1: $\bar{\sigma}_1 = 900 + 147 * 0.75 = 1010.25$ при впровадженні одиниці.

У результаті відновлення за формулою маємо: $1010.25 - 1000 < 0.5 * 147$, тобто $10.25 < 73.5$ – тому відновлюється нуль, хоча занурювали одиницю**.**

У цьому прикладі наочно видно, як при зануренні біту додаткової інформації, відбувається збільшення числа у розряді до якого проводилося округлення, що у результаті призводить до помилки при відновлені. Саме через таку особливість алгоритму, відсоток відновлення у ньому має такий низький показник, а разом з тим й алгоритм має малу ефективність.

Для вирішення такого роду помилки та, власне, для підвищення ефективності стеганоалгоритму, заснованому на сингулярному розкладі блоків матриці зображення, пропонується його модифікація.

Модифікація алгоритму полягає в тому, що перше сингулярне число буде збурюватися не за рахунок саме другого сингулярного числа, а з використанням замість нього одного з наступних сингулярних чисел, в залежності від того, чи відбувається збільшення числа у розряді до якого проводиться округлення із

використанням кожного наступного сингулярного числа.

Підбір сингулярного числа, за рахунок якого буде збурюватися перше сингулярне число, буде відбуватися за наступними формулами.

При зануренні одиниці, коли $\sigma_1 \geq 100$:

$$\overline{\sigma_1} = [\sigma_1] + 0.75 \cdot \sigma_2, \quad (2.6)$$
$$[\sigma_1] + 0.75 \cdot \sigma_2 < [\sigma_1] + 100,$$
$$\sigma_2 < 100/0.75,$$

де σ_1 – найбільше (перше) сингулярне число;

σ_2 – друге сингулярне число;

$\bar{\sigma}_1$ – найбільше (перше) сингулярне число після збурення.

При зануренні одиниці, коли $\sigma_1 < 100$:

$$\overline{\sigma_1} = [\sigma_1] + 0.75 \cdot \sigma_2, \quad (2.7)$$
$$[\sigma_1] + 0.75 \cdot \sigma_2 < [\sigma_1] + 10,$$
$$\sigma_2 < 10/0.75.$$

При зануренні нуля, коли $\sigma_1 \geq 100$:

$$\overline{\sigma_1} = [\sigma_1] + 0.25 \cdot \sigma_2, \quad (2.8)$$
$$[\sigma_1] + 0.25 \cdot \sigma_2 < [\sigma_1] + 100,$$
$$\sigma_2 < 100/0.25.$$

При зануренні нуля, коли $\sigma_1 < 100$:

$$\overline{\sigma_1} = [\sigma_1] + 0.25 \cdot \sigma_2, \quad (2.9)$$
$$[\sigma_1] + 0.25 \cdot \sigma_2 < [\sigma_1] + 10,$$
$$\sigma_2 < 10/0.25.$$

Завдяки запропонованій модифікації, ефективність алгоритму зростає, тобто зростає показник відсотку відновлення. Тому, для роботи алгоритму з контрастними зображеннями, які містять у собі велику кількість деталей, розроблена модифікація, яка є незамінною, бо дозволяє гарантовано правильно відновлювати вбудовану інформацію.

Узагальнені кроки модифікованого стеганоалгоритму, що використовує сингулярний розклад матриці контейнера, представлені нижче.

Вбудовування додаткової інформації:

а) розбиття матриці на блоки 8 на 8; кожен блок використовується для вбудовування одного біту додаткової інформації;

б) для кожного отриманого блоку будується сингулярний розклад (2.1);

в) для кожного набору сингулярних чисел встановлюється коефіцієнт підбору сингулярного числа λ:

$$\lambda = \begin{cases} 100, & \sigma_1 \geq 100 \\ 10, & \sigma_1 < 100 \end{cases}, \tag{2.10}$$

де σ_1 – найбільше (перше) сингулярне число.

г) для кожного набору сингулярних чисел обирається таке σ_n, що виконується умова:

$$\sigma_n = \lambda / 0.75, \tag{2.11}$$

де λ – коефіцієнт підбору сингулярного числа.

д) якщо біт додаткової інформації дорівнює нулю, то:

$$\bar{\sigma}_1 = roundn(\sigma_1, K) + \frac{1}{4} \cdot \sigma_n, \tag{2.12}$$

де σ_1 – найбільше (перше) сингулярне число;

σ_n – n-е сингулярне число;

$\bar{\sigma}_1$ – найбільше (перше) сингулярне число після збурення;

K – розряд округлення;

roundn() – функція округлення до розряду K.

е) якщо біт додаткової інформації дорівнює одиниці, то:

$$\bar{\sigma}_1 = roundn(\sigma_1, K) + \frac{3}{4} \cdot \sigma_n. \tag{2.13}$$

ж) використання сингулярного розкладу для повернення в просторову область:

$$\bar{A} = U \bar{\Sigma} V^T, \tag{2.14}$$

де $\bar{A}$ – блок матриці зображення зі збуренням;

$\bar{\Sigma}$ – діагональна матриця сингулярних чисел із збуренням.

Вилучення додаткової інформації:

а) розбиття матриці стеганоповідомлення на блоки 8 на 8;

б) для кожного отриманого блоку будується сингулярний розклад (2.1);

в) для кожного набору сингулярних чисел встановлюється коефіцієнт підбору сингулярного числа λ:

$$\lambda = \begin{cases} 100, & \sigma_1 \geq 100 \\ 10, & \sigma_1 < 100 \end{cases}, \tag{2.15}$$

де σ_1 – найбільше (перше) сингулярне число.

г) для кожного набору сингулярних чисел обирається таке σ_n, що виконується умова:

$$\sigma_n = \lambda / 0.75, \tag{2.16}$$

де λ – коефіцієнт підбору сингулярного числа.

д) якщо вірною є нерівність:

$$\overline{\overline{(\sigma_1}} - roundn(\bar{\bar{\sigma}}_1, K)) < \frac{1}{2} \cdot \bar{\sigma}_n , \tag{2.17}$$

тоді відновлюємо 0, в іншому випадку – 1.

2.3 Ефективність модифікованого стеганографічного алгоритму

Головною метою розробки модифікованого стеганографічного алгоритму, заснованого на сингулярному розкладі блоків матриці цифрового зображення, було підвищення ефективності початкового алгоритму, тобто підвищення відсотку правильно відновленого секретного повідомлення. Для оцінки ефективності стеганографічного алгоритму використано показник об’єму відновленої інформації, у відповідності до наступної формули:

$$D = \frac{\sum_{i=1}^{N} \overline{s_i \otimes s'_i}}{N} \cdot 100\%, \tag{2.18}$$

де $S = (s_i),\ i = \overline{[1, N]}$ – додаткова інформація вбудована в контейнер в бінарному форматі;

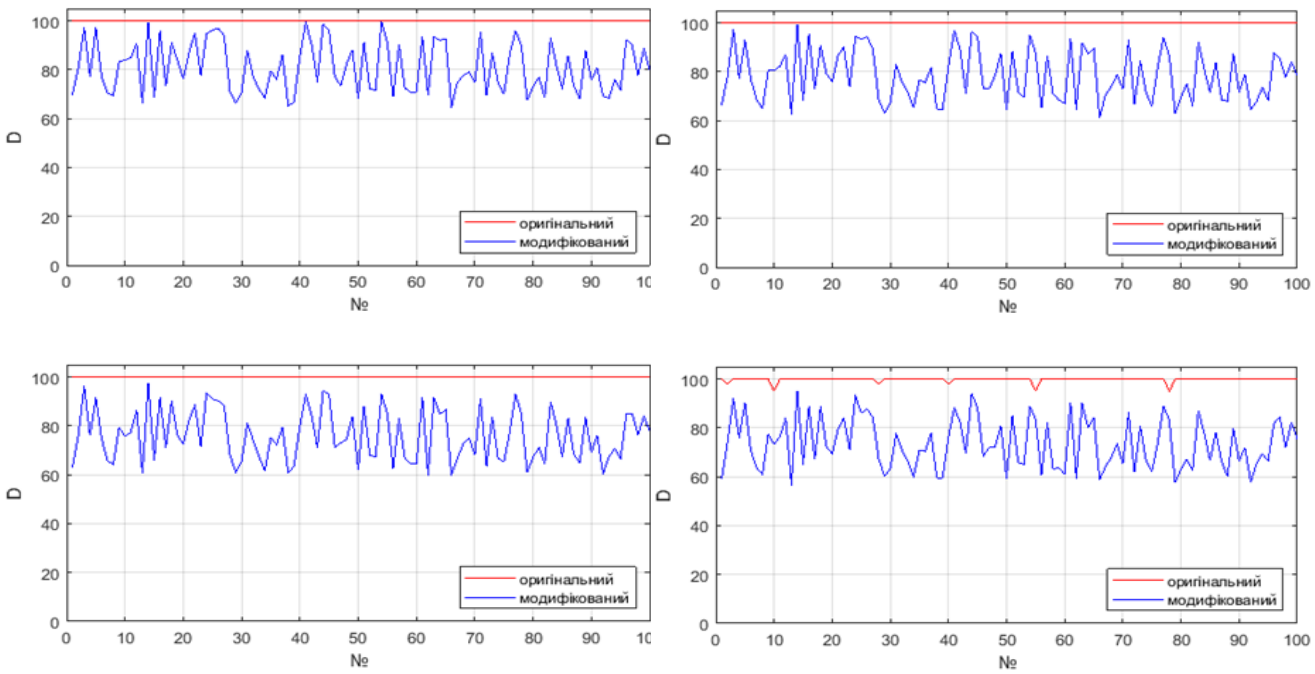
$S' = (s'_i),\ i = \overline{[1, N]}$ – додаткова інформація вилучена з стеганоповідомлення в бінарному форматі;

N – розмір додаткової інформації;

$\otimes$ – операція виключне або;

$\overline{\cdot}$ – операція інверсія.

В середовищі розробки Matlab було проведено обчислювальний експеримент, в якому було задіяно сто цифрових зображень, що початково були збережені в форматі без втрат. В кожне зображення за допомогою початкового алгоритму та модифікованого алгоритму було виконано занурення додаткової інформації. Отримані стеганоповідомлення були збережені без втрат та з втратами, з різними коефіцієнтами стиску Q. Для отриманих таким чином стеганоповідомлень було виконано вилучення секретних повідомлень та підраховано показник об'єму відновленої інформації (2.18). Результати експерименту представлені на рисунку 2.3.



в) г)

Рисунок 2.3 – Оцінка об'єму відновленої інформації

а) – без стиску; б) – Q=95; в) – Q=85; г) – Q=75;

Збереження надійності сприйняття є дуже важливою характеристикою для будь-якої стеганосистеми, бо вона безпосередньо впливає на головну задачу стеганографії – збереження факту секретності передачі інформації. Для оцінки надійності сприйняття стеганоповідомлення використано показник пікового

відношення «сигнал-шум» PSNR, у відповідності до наступної формули:

$$PSNR = 10 \cdot \log_{10}\left(\frac{255^2}{MSE}\right), \tag{2.19}$$

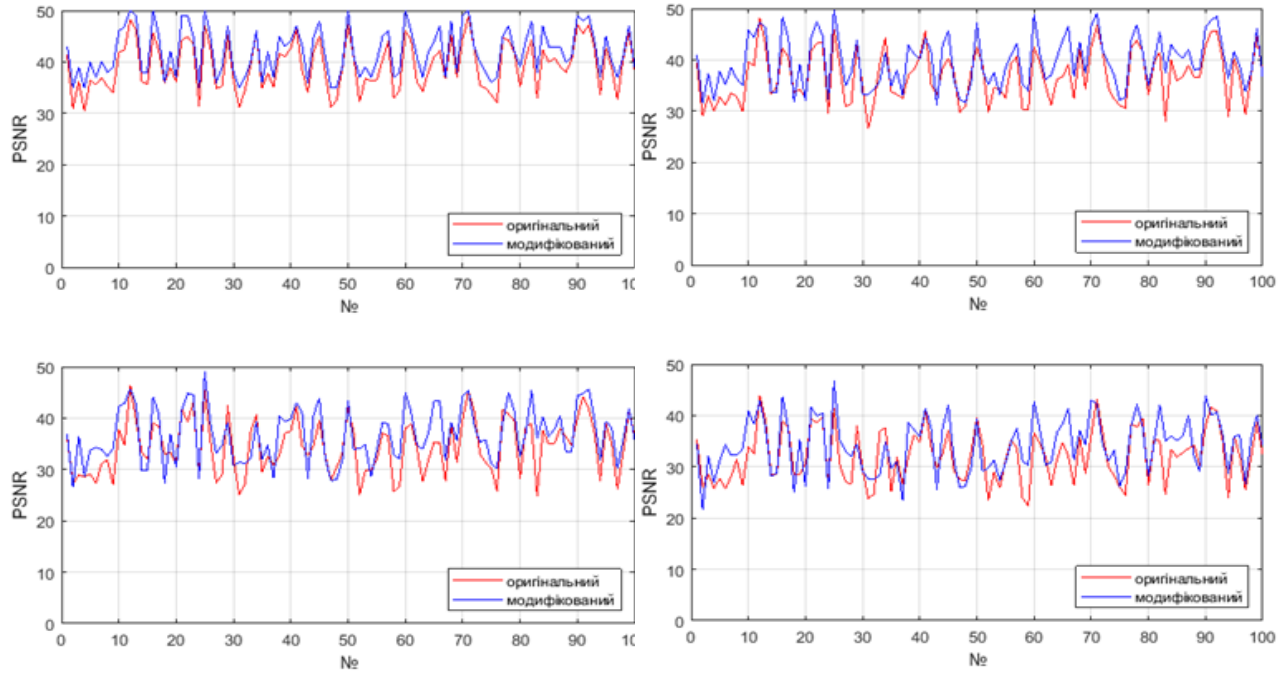
$$MSE = \frac{1}{M \cdot N}\sum_{i=1}^{N}\sum_{j=1}^{M}(I_{i,j} - \bar{I}_{i,j})^2,$$

де M, N – розмір стеганоповідомлення;

$I_{i,j}$ – значення яскравості контейнера;

$\bar{I}_{i,j}$ – значення яскравості стеганоповідомлення.

Для використаних у попередньому експерименті контейнерів та отриманих стеганоповідомлень, підраховано PSNR (2.19). Результати експерименту представлені на рисунку 2.4.



в) г)

Рисунок 2.4 – Оцінка надійності сприйняття

а) – без стиску; б) – Q=95; в) – Q=85; г) – Q=75;

В розділі представлені теоретичні основи стеганографічного алгоритму, заснованого на сингулярному розкладі блоків матриці цифрового зображення.

Детально досліджено послідовність вбудовування та вилучення секретного повідомлення для початкового стеганографічного алгоритму. Після проведення аналізу ефективності, було виявлено причину низького показника відсотку правильного відновлення секретної інформації для даного алгоритму.

Задля вирішення проблеми було розроблено модифікацію, яка суттєво підвищує його ефективність. У розділі наведена покрокова робота алгоритму з модифікацією.

Для оцінки ефективності модифікованого стеганографічного алгоритму, проведено обчислювальний експеримент з встановлення надійності сприйняття та відсотка правильно відновленого секретного повідомлення. Практично доведено, що модифікація алгоритму збільшує його ефективність.

3 ПРАКТИЧНА РЕАЛІЗАЦІЯ СТЕГАНОГРАФІЧНОГО АЛГОРИТМУ, ЗАСНОВАНОГО НА СИНГУЛЯРНОМУ РОЗКЛАДІ БЛОКІВ МАТРИЦІ ЦИФРОВОГО ЗОБРАЖЕННЯ

3.1 Обґрунтування вибору середовища розробки програмного застосування

Для того щоб реалізувати стеганографічний алгоритм, заснований на сингулярному розкладі блоків матриці цифрового зображення, було обрано середовище розробки MATLAB.

MATLAB – це середовище розробки для програмування та числових обчислень, яке призначено для розробки алгоритмів та створення моделей. Програма працює на більшості існуючих операційних системах та має у своєму арсеналі потужні інструменти для розв’язання математичних та інженерних задач.

Для середовища розробки MATLAB представляє високорівневу інтерпретовану мову програмування. Мова програмування MATLAB заснована на матричній структурі даних. Вибір такої структури даних обумовлений зручністю математичних обчислень. Завдяки інтегрованості середовища розробки, у MATLAB є можливість написання програм з використанням інших високорівневих мов програмування. MATLAB дає можливість для використання об’єктно-орієнтованого підходу розробки програмних рішень та побудови інтерфейсів до програм [15].

Розробка коду для програмного продукту розділяється на два типи: написання функцій та написання скриптів. Скрипти представляють собою набір команд, які використовують загальний простір для зберігання значень змінних. Функції, у свою чергу, зберігають змінні у власному просторі, а також відрізняються наявністю вхідних і вихідних аргументів [16].

Для візуалізації у MATLAB є своя графічна система, яка представляє собою керовану графіку. В ній наявні команди високого рівня для візуалізації даних у двовимірному або тривимірному вигляді та для обробки зображень. Окрім команд високого рівня, у керованій графіці MATLAB є й команди низького рівня, які використовуються для редагування зовнішнього вигляду графіки.

Багатофункціональність середовища розробки MATLAB розкривається у вигляді створених розробниками наборів інструментів. Набір інструментів або toolbox – це комплекс функцій та об’єктів, які написані мовою MATLAB, та покликані вирішувати задачі певної області [17-18].

Серед таких наборів інструментів слід виділити набір цифрової обробки зображень – Image Processing Toolbox, який є дуже актуальним у питанні реалізації стеганографічних систем та алгоритмів. Набір інструментів Image Processing Toolbox створений для візуалізації та моделювання, а також містить у собі багато чисельні алгоритми та програми для аналізу та обробки зображень.

Задачі, які можна вирішувати із застосуванням Image Processing Toolbox]:

- запис та читання графічних файлів;
- побудова фільтрів, фільтрація та відновлення зображень;
- маніпуляції із розмірами зображень;
- аналіз та статистична обробка зображень;
- маніпуляції із кольором;
- двовимірні перетворення;
- візуалізація даних.

Саме за допомогою цього набору інструментів була реалізована робота із зображеннями в алгоритмі, заснованому на сингулярному розкладі блоків матриці цифрового зображення.

Серед функціоналу середовища MATLAB важливою є можливість побудови графічних інтерфейсів, реалізація якої представлена у середовищі під назвою GUIDE [19]. Програмування інтерфейсів за допомогою цього середовища відбувається швидким та зручним чином. В ньому представлена велика кількість елементів керування: кнопки, поля вводу, списки, що розгортаються, прапорці, перемикачі, які можуть бути розташовані на формі за допомогою миші. Кожен встановлений елемент керування для інтерфейсу має свій набір параметрів в залежності від свого типу, які можна легко змінювати. Для елементів керування інтерфейсу у програмному коді можна створювати обробники подій, які будуть виконувати вказаний набір команд, коли відбудеться певна подія. Програмне

застосування, що розробляється, може складатися з одного або декількох вікон. Застосування може легко реалізовувати можливість вводу та виводу різного виду інформації, наприклад, текстової, графічної або звукової.

Для взаємодії з користувачем у GUIDE передбачено можливість створення діалогових вікон, з використанням яких можна відкривати та зберігати файли або передати файл для друку. За допомогою зручного інструментарію у середовищі GUIDE було побудовано інтерфейс користувача програмного застосування, яке реалізує стеганографічний алгоритм.

Виходячи з вищеперерахованого, можна зробити висновки про наявність великої кількості переваг програмного продукту MATLAB над його конкурентами у вигляді інших мов програмування та середовищ розробки.

Переваги, які обумовлюють вибір MATLAB для розробки програмного рішення цієї роботи:

- інтегрованість;
- наявність великої кількості вбудованих бібліотек;
- зручний та інтуїтивно зрозумілий інтерфейс;
- наявність зручного конструктору для побудови програмного інтерфейсу;
- зручність математичних обчислень;
- можливість відладки програмного коду.

3.2 Організація роботи програмного застосування

Теоретичні дослідження стеганографічного алгоритму, заснованого на сингулярному розкладі блоків матриці цифрового зображення, практично реалізовані у вигляді програмного застосування, організація роботи якого є модульною.

Модульність, за визначенням, означає наявність у програмній реалізації функціональних частин (підпрограм), які називаються модулями.

Для демонстрації взаємодії окремих модулів, при організації роботи програмного застосування, спроектована діаграма послідовності, що відображає

динамічну взаємодію системи (рис. 3.1).

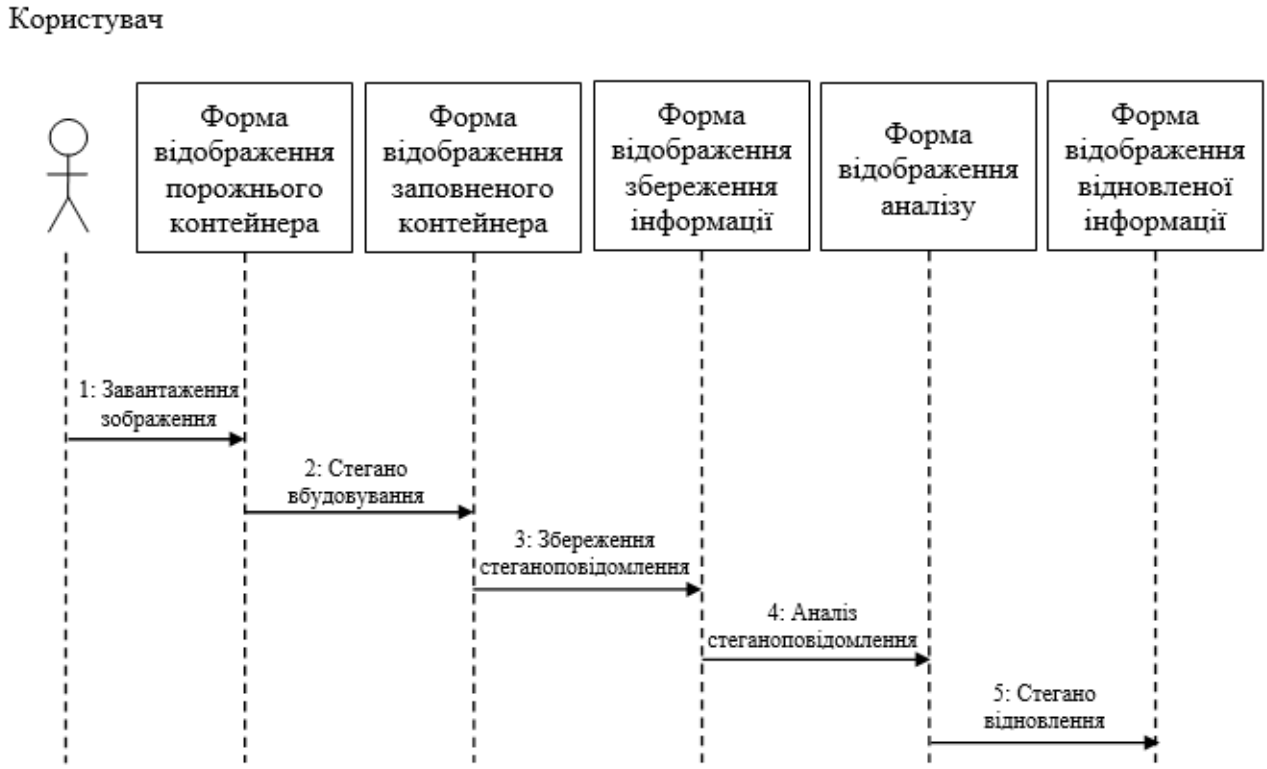


Рисунок 3.1 – Діаграма послідовності стеганографічного застосування

Розроблене програмне застосування для розв’язку стеганографічної задачі складається з п’ятьох модулів.

Модуль завантаження зображення – це модуль, який відповідає за збереження цифрового зображення у середовищі програми та за відображення інформації, що стосується розмірів та шляху до завантаженого зображення у комп’ютерній файловій системі.

Модуль занурення додаткової інформації – це модуль, який виконує одну з ключових функцій стеганографічного алгоритму – занурення додаткової інформації у зображення-контейнер. Занурення відбувається за алгоритмом, заснованим на сингулярному розкладі матриці контейнера та його версією із модифікацією, яка запропонована у цій роботі.

Модуль збереження – це модуль який відповідає за збереження збуреного контейнеру (стеганоповідомлення) у файловій системі комп’ютера.

Модуль аналізу – це модуль, який реалізує функції аналізу, передбачені для порівняння ефективності алгоритму без модифікації та його версії з модифікацією. Цими функціями є: відображення проценту відновлення додаткової інформації; відображення пікового відношення сигнал-шум; відображення графіку значень сингулярних чисел зображення-контейнера та стеганоповідомлення.

Модуль відновлення додаткової інформації – це модуль, який, відповідно до алгоритму занурення, реалізує другу важливу функцію стеганографічного алгоритму – відновлення інформації з стеганоповідомлення.

3.3 Інструкція з використання програмного застосування

Для реалізації стеганографічного алгоритму, заснованого на сингулярному розкладі блоків матриці цифрового зображення, ефективність якого була підвищена у кваліфікаційній роботі, спроектовано та розроблено програмне застосування, яке має інтерфейс приведений на рисунку 3.2.

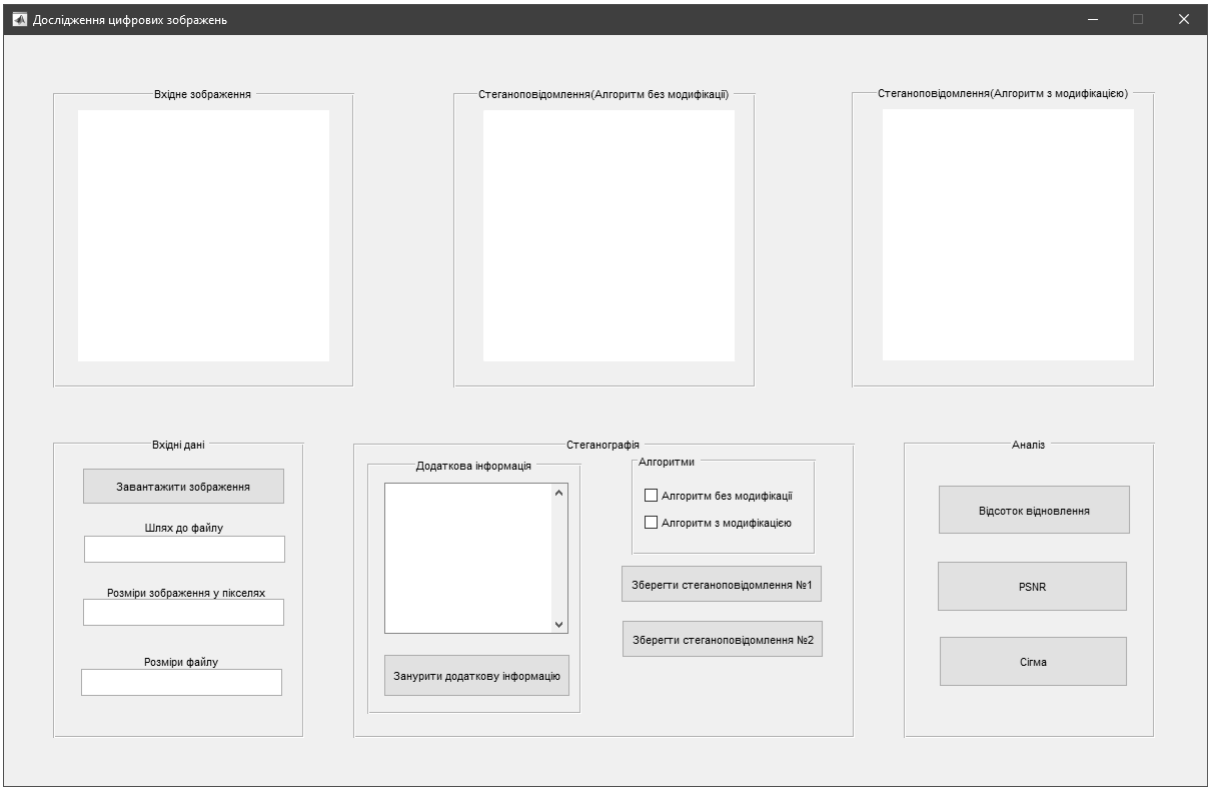


Рисунок 3.2 – Інтерфейс стеганографічного програмного застосування

Першим, невід’ємним кроком, при експлуатації поданого програмного застосування є завантаження зображення-контейнера.

Для цього потрібно натиснути на кнопку з відповідною назвою «Завантажити зображення». При натисканні на цю кнопку відкривається діалогове вікно, у якому користувачеві потрібно обрати зображення для завантаження у застосування. Під кнопкою завантаження, після вибору файлу, з’являється інформація щодо шляху до нього та його розмірів у пікселях та у байтах (рис. 3.3).

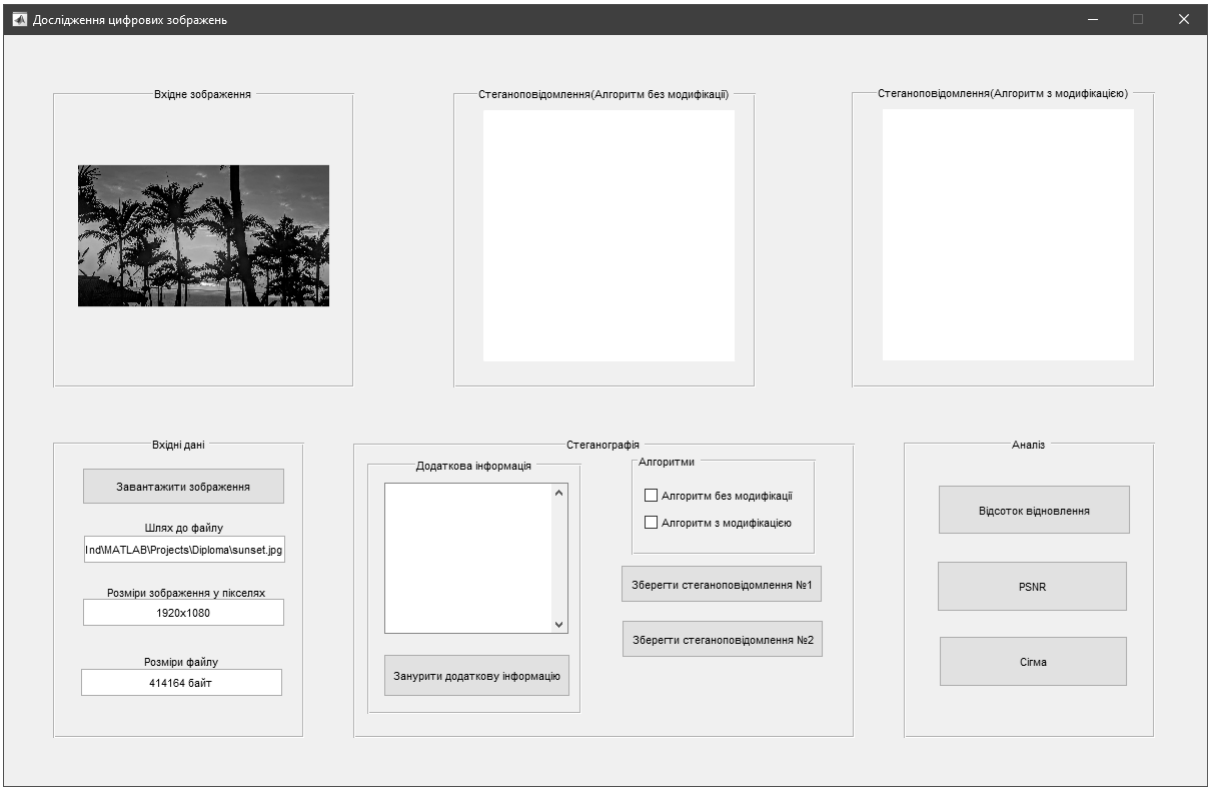


Рисунок 3.3 – Демонстрація стеганографічного контейнеру

Головна частина стеганографічного застосування, а саме занурення додаткової інформації за допомогою алгоритму заснованому на сингулярному розкладі блоків матриці зображення, в інтерфейсі реалізована у вигляді поля для вводу додаткової інформації та кнопки для виконання занурення. Для подальшого порівняння ефективності алгоритму з модифікацією та без неї, була реалізована можливість обирати між цими двома варіантами алгоритму. Також передбачена можливість обрання обох алгоритмів одночасно (рис. 3.4).

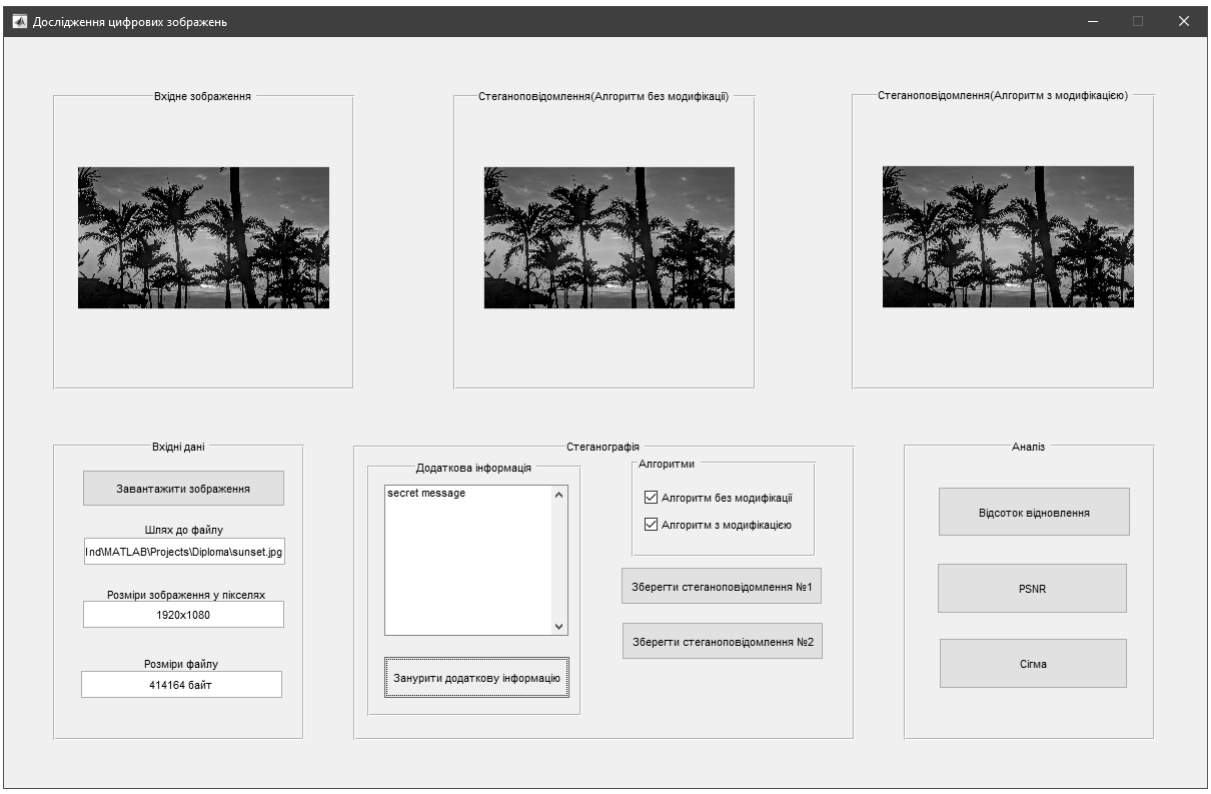


Рисунок 3.4 – Демонстрація стеганоповідомлень, отриманих в результаті занурення додаткової інформації початковим та модифікованим алгоритмами

Після занурення додаткової інформації у цифрове зображення, з’являється можливість збереження одного з двох варіантів стеганоповідомлень, або обох одразу.

Наступна частина функціоналу програми – це аналіз роботи стеганографічних алгоритмів. Аналіз реалізований у вигляді таких можливостей: підрахунок відсотку відновлення додаткової інформації зануреної за допомогою початкового та модифікованого алгоритму; підрахунок пікового відношення сигнал-шум для стеганоповідомлень, отриманих за допомогою початкового та модифікованого алгоритму; відображення графіку значень сингулярних чисел обраного блоку матриці цифрового зображення.

Метою даної роботи є саме підвищення ефективності стеганоалгоритму, яка полягає в успішності відновлення бітів, що занурюються у цифрове зображення. Тому для порівняння ефективності алгоритму у його початковому вигляді та алгоритму з модифікацією, була створена функція, яка відображає відсоток відновлення бітів при використанні кожного з алгоритмів. На рисунку 3.5 можна

побачити наглядну різницю у відсотках відновлення, що свідчить про більш високу ефективність алгоритму з модифікацією.

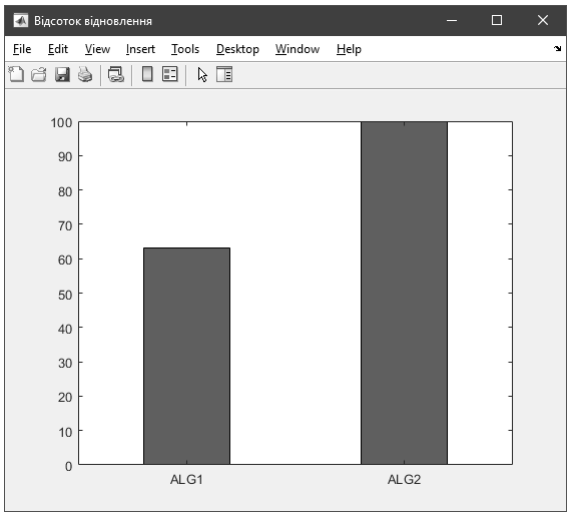


Рисунок 3.5 – Графік порівняння відсотків відновлення додаткової інформації зануреної за допомогою початкового та модифікованого алгоритму

Для того, щоб підрахувати пікове відношення сигнал шум треба натиснути на кнопку «PSNR». У вікні, що з’явилося (рис. 3.6), користувачеві буде представлений графік, на якому будуть відображені значення PSNR для кожного з двох алгоритмів, для порівняння ефективності алгоритмів.

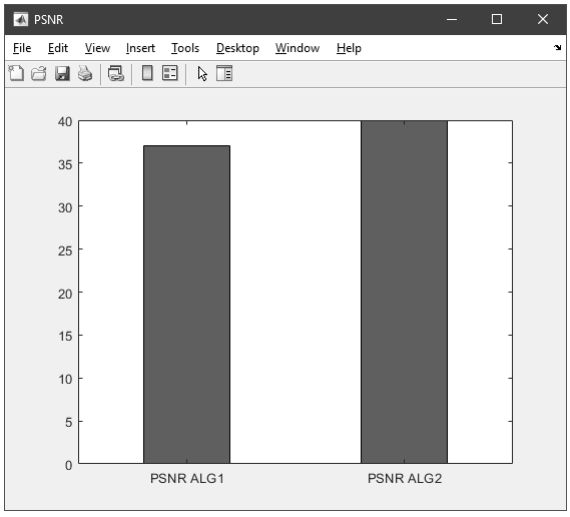


Рисунок 3.6 – Графік порівняння пікового відношення сигнал-шум для стеганоповідомлень, отриманих за допомогою початкового та модифікованого алгоритму

Додатковою можливістю для проведення аналізу, реалізованою у програмному застосуванні, є перегляд графіку значень сингулярних чисел блоку матриці цифрового зображення. Для цього користувачеві треба натиснути на кнопку «Сигма» та у вікні, що відкриється, можна буде побачити побудований графік (рис. 3.7).

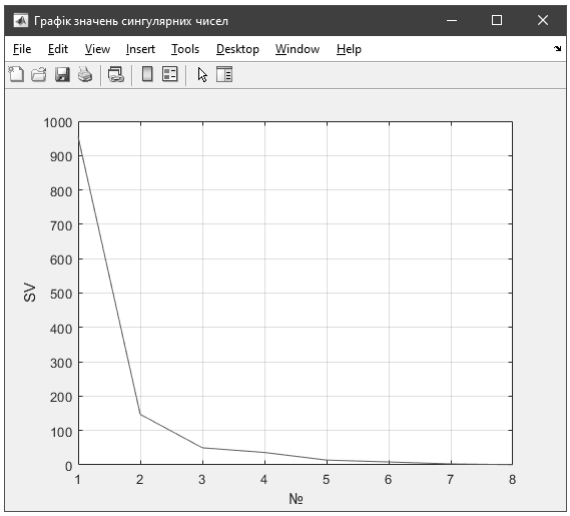


Рисунок 3.7 – Графік значень сингулярних чисел блоку зображення

В даному розділі розглянута практична реалізація стеганографічного алгоритму, заснованого на сингулярному розкладі блоків матриці цифрового зображення. Виконано обґрунтування вибору середовища розробки програмного застосування. MATLAB містить необхідний функціонал для реалізації стеганоалгоритму, який розглядається. Також цей функціонал є зручним та зрозумілим для використання, що є безумовно великою перевагою MATLAB, як засобу реалізації програмних застосувань для стеганографічних досліджень, у порівнянні з його конкурентами. Представлена організація роботи програмного застосування. Описано функціонування модулів програмного застосування та їх взаємодія за допомогою діаграми послідовності. Приведена докладна інструкція з використання програмного застосування Описано функціонал та можливості застосування, яке реалізує початковий стеганографічний алгоритм та його версію із модифікацією, що підвищує ефективність. Програмний код реалізованого програмного застосування представлений у додатку А.

4 ОХОРОНА ПРАЦІ

Одними з найважливіших складових людського життя є здоров’я та безпека людини. Будь-яка робоча діяльність супроводжується шкідливими факторами, які негативним чином впливають на людське здоров’я. Тому завданням охорони праці є захист людини від негативних факторів у будь-якому напрямку, в якому бере участь людина. Захист в охороні праці полягає у поліпшенні умов праці, підвищенні безпеки цих умов, зниженні ризиків професійних захворювань та травматизму [21].

Під час інтенсивної роботи за комп’ютером людина може попасти під вплив таких шкідливих факторів: підвищений рівень шуму; недостатня освітленість робочої поверхні; вимушена робоча поза; підвищена напруга в електричній мережі з можливістю замикання крізь тіло людини; підвищений рівень електромагнітного випромінювання.

Шум – це звуки, які мають різну інтенсивність та гучність та у такій сукупності можуть завдавати шкоди слуховому апарату людини. Джерела шуму на робочому місці: прилади для друку та копіювання, пристрої для контролю вологості і кондиціонування.

Згідно до документу ДСН 43.3.6 037-99 [22] допустимий рівень звукового тиску у октавних смугах частот на робочому місці програміста повинен становити не більше 50дБА. За відсутністю приладів друкування та при непотрібності використання систем кондиціонування, фактичне зашумлення робочого місця виконання кваліфікаційної роботи 45дБА.

Фізичне здоров’я також може постраждати в результаті не правильного положення під час сидіння за комп’ютером. Упродовж тривалої та інтенсивної роботи хребет, колінні суглоби, кісті рук та інші частини людського тіла піддаються великому навантаженню, що може призвести до відповідних захворювань.

На зорову систему людини на такому робочому місці негативно впливають погана освітленість та не дотримування правильної відстані від очей до монітору

комп’ютера. У зв’язку з тим, що робота відбувається у приміщенні, освітленість має бути наближеною до умов сонячного освітлення за допомогою штучних джерел освітлення.

Освітлення на робочому місці представляє собою комбінацію природного та штучного освітлення. Вечірнє природне освітлення приміщення складає 400лк. Такий показник не відповідає нормі, вказаній у документі ДБН В.2.5-28-2006 [23]. Освітленість поверхні під час роботи за комп’ютером або роботи з документами повинна складати не менше ніж 500лк. Тому у такий час доби до природного освітлення надане загальне штучне освітлення.

Для збереження здоров’я зорової системи окрім освітлення треба також дотримуватися норм, що стосуються положення екрану відносно очей людини, які описані у пунктах 4.14-4.15 ДСанПіН 3.3.2.007-98 [24]. Згідно документу, відстань від очей користувача до екрану має складати 600-700мм. Також розташування екрану має забезпечувати зручність зорового спостереження у вертикальній площині під кутом 30 градусів до лінії погляду.

Рівень напруженості електростатичного поля, під вплив якого може потрапити людина у робочому місці становить 15кВ/м протягом восьмигодинного робочого дня. Нормою, що вказана у документі ДСанПІН 3.3.2.007-98 [21] є 15кВ/м. Для іонізуючих електромагнітних випромінювань нормативне значення визначено як $7,74 * 10^{-12}$А/кг, а фактичне значення $7 * 10^{-12}$А/кг.

Параметри мікроклімату мають значний вплив на якість умов праці. До цих параметрів належать: температура, вологість та швидкість повітря. Для теплого періоду року, згідно до документу ДСанПіН 3.3.2.007-98 [21] оптимальні значення температури у приміщенні складають 23-25°С, вологості 40-60%, а швидкості повітря менше ніж 0.1м/с. Для холодного періоду року – температура 22-24°С, така ж сама вологість та швидкість повітря, як і у теплий період року.

Для усіх вищеперерахованих шкідливих факторів побудовано таблицю, у якій наведено фактичні та нормативні значення факторів виробничого середовища для робочого місця, що представлені у таблиці 4.1.

Таблиця 4.1 – Порівняння норм шкідливих чинників з фактичним значенням

Документ	Фактор виробничого середовища і трудового процесу	Нормативне значення	Фактичне значення
ДСН 43.3.6 037-99	Шум: – рівень фонового шуму; – рівень звукового тиску.	 50дБА 60дБ	 45дБА 50дБ
ДБН В.2.5-28-2006	Освітленість	500лк	400лк
ДСанПІН 3.3.2.007-98	Електромагнітні випромінювання: – напруженість електростатичного поля; – іонізуючі електромагнітні випромінювання.	 20кВ/м $7{,}74 * 10^{-12}$ А/кг	 15кВ/м $7 * 10^{-12}$ А/кг
ДСанПіН 3.3.2.007-98	Робоча поза: – відстань від екрану до очей; – кут, який забезпечує зручність спостереження.	 600-700мм 30°	 600мм 35°
ДСанПіН 3.3.2.007-98	Параметри мікроклімату (теплий період року): – температура повітря; – вологість повітря; – швидкість руху повітря.	 23-25°С 40-60% 0,1м/с	 22°С 59% 0,09м/с

У результаті виконання аналізу безпеки робочого місця, було виявлено деяку кількість шкідливих чинників, що можуть погано впливати на здоров’я працюючої людини. Серед виявлених слід виділити такі шкідливі чинники: зашумленість робочого місця, недостатня освітленість робочої поверхні, не правильне положення тіла при сидінні, електромагнітне випромінювання, не оптимальні показники мікроклімату.

Шум. Професійна діяльність людей, працюючих у сфері комп’ютерних наук, робочі місця яких знаходяться у спеціально обладнаних офісах може супроводжуватися великою кількістю шуму. Задля вирішення цієї проблеми треба вдатися до таких заходів, як зменшення кількості приладів, які є джерелом надмірного шуму, їх правильне розташування, контроль якості обладнання (у випадку коли надмірний шум є результатом не правильного функціонування приладу), використання звукоізоляційних та звукопоглинаючих засобів. Також рівень шуму може залежати від розташування будівлі, в якій знаходиться офіс. Тому, важливо, щоб розташування було на значній відстані від місць з великою кількістю розповсюджуваного гучного шуму. Прикладом такого місця є аеропорт

або будівництво.

Освітленість. Для забезпечення безпечних умов праці програміста, його робоче місце має бути достатньо освітленим. Задля того щоб досягти достатньої освітленості, треба використовувати сумісне освітлення. Природного освітлення може вистачати в світлий час доби, але в іншому випадку дуже важлива наявність додатково штучного освітлення. Окрім наявності, важливим є правильне розташування та розподіл джерел світла у приміщенні. Сукупність цих засобів має надавати задовільний показник освітленості у будь-який робочий час доби.

Вологість та температура повітря. Для підвищення вологості повітря слід використовувати зволожувачі повітря. Використання цього приладу повинно сприяти тому, щоб вологість не знижувалась нижче норми, яка становить 40-60%. Низькі показники вологості при тривалому впливі на людину можуть призводити до серйозних захворювань. Високий рівень вологості, у свою чергу, збільшує віддачу тепла тілом людини, що також може призвести до захворювань. Для зменшення рівня вологості за необхідністю треба використовувати вологопоглиначі, та встановлювати в офіси якісну систему вентиляції та обігріву. Нормований рівень температури у приміщенні має бути встановлений шляхом використання приладів кондиціонування, обігріву та охолодження.

Робоча поза програміста під час роботи – це дуже важливий фактор впливу на здоров’я працюючого. Окрім того, що потрібно дотримуватися норм щодо положення тіла при роботі за комп’ютером, описаних у документі ДСанПіН 3.3.2.007-98 [24], також можна скористатися приведеним у цьому же документі комплексом вправ для хребта, рук та вправами для поліпшення мозкового кровообігу. У додаток до перерахованих вправ, слід зазначити наявність рекомендацій щодо психофізіологічного розвантаження.

Для будь-якого виробничого процесу чи іншого виду діяльності людини, пов’язаної з роботою у приміщенні характерним ризиком є пожежа. Пожежі можуть призводити як до великих економічних та матеріальних втрат, так і до загибелі людей. Тому пожежна безпека займає особливо важливе місце у питані охорони праці.

Будь-яка пожежа безумовно є небезпекою для здоров’я людини. До небезпечних факторів пожежі можна віднести:

- власне відкрите полум’я та його іскри – відкритий вогонь може завдати великої шкоди тілу людини при контакті;
- велика температура повітря – може призводити до опіків дихальних шляхів людини;
- токсичні продукти горіння – викликають отруєння з подальшими наслідками;
- недостатній вміст кисню у повітрі – нестача кисню через яку людина задихається;
- руйнування будівель та споруд – пошкодженні елементи будівель можуть завдати шкоди людині, притиснути її, не даючи змоги рятуватися від полум’я або важко поранити падінням на людину;
- знижена видимість внаслідок задимлення – може завадити людині вибратися з небезпечного місця пожежі.

Задля того, щоб запобігти виникненню пожеж треба розібратися у тому через що вони виникають. Причинами виникнення пожеж у приміщенні, яке обладнане електронно-обчислювальною технікою можуть стати такі фактори:

- коротке замикання – може статися через несправність розеток, не правильну або пошкоджену ізоляцію дротів та через потрапляння води у електронну мережу. У такому випадку через підвищення тепловиділення, викликане різким зниженням опору мережі та підвищенням сили струму, трапляється займання;
- перенавантаження мережі – трапляється у випадку, коли велика кількість електроприладів підключена до однієї мережі одночасно. Така ситуація також може призводити до займання;
- не правильна експлуатація електронних приладів – сюди входить недбале використання дротів, розеток та власне техніки, її не правильне розміщення, нехтування системами охолодження.

При розгляданні питань пожежної безпеки слід зазначити, що вибір типу

вогнегасників та їх кількості насамперед залежить від наявності речовин та матеріалів певного класу пожежі, площі приміщення та його типу за вибухопожежною або пожежною небезпекою.

Наявність певних класів речовин у приміщеннях, що є об’єктами пожежної безпеки визначає клас будівлі. Тому у документі НАПБ Б.03.002-2007 приміщення та будинки класифіковано на такі категорії: А, Б, В, Г, Д.

Згідно до документу НАПБ Б.07.005-86 визначено категорію приміщення, в якій виконано роботу, як категорія В. Для такої категорії приміщення характерним є наявність таких речовин: горючі гази, легкозаймисті, горючі і важкогорючі рідини, а також речовини та матеріали, які здатні при взаємодії з водою, киснем повітря або один з одним вибухати і горіти або тільки горіти; горючий пил і волокна, тверді горючі та важкогорючі речовини і матеріали, за умови, що приміщення, в яких вони знаходяться (обертаються), не відносяться до категорій А, Б. Матеріали, що відносяться до вищеперерахованих типів є складовими таких елементів робочого місця програміста: офісні стільці, корпуси комп’ютерів, миші, клавіатури – у складу яких є пластмаса; дерев’яні меблі та власне комп’ютерна техніка, яка під навантаженням піддається нагріванню.

Така категорія приміщення потребує наявність таких засобів пожежної безпеки, як евакуаційні виходи, плани евакуації, системи сповіщення та первинні засоби пожежогасіння.

Виходячи з розміру приміщення, у якому знаходиться робоче місце, яке складає 98м^2 та з факту наявності у приміщенні персональних електронно-обчислювальних машин (ПЕОМ), можна зробити висновки про необхідність облаштування такого приміщення 13 вуглекислотними вогнегасниками кожен з котрих потрібен мати заряд вогнегасної речовини у розмірі 5кг відстань між котрими має бути не більшою за 15м. Що стосується моделі вогнегасника то вибір падає на модель ВВК-1,4. Застосування вуглекислотних вогнегасників ВВК-1,4 здійснюється у випадках спалаху речовин, горіння яких припиняється при відсутності кисню, та у тому числі для гасіння електроустановок під напругою до тисячі вольт.

Технічні параметри вогнегасника ВВК-1,4:

- скраплений діоксид вуглецю у якості вогнегасної речовини;
- внутрішній тиск 15мПа;
- розміри – 454х108мм;
- діапазон робочих температур від -20 °С до +50 °С.

У результаті виконання аналізу робочого місця програміста були виявлені шкідливі фактори такі як: зашумленість, недостатня освітленість робочої поверхні, вимушена робоча поза, підвищений рівень електромагнітного випромінювання та неоптимальні показники мікроклімату. До кожного фактору були надані рекомендації щодо оптимізації їх показників у вигляді організаційних та інженерних рішень.

Частиною аналізу безпеки на робочому місці стала пожежна безпека, якій було виділено особливу увагу. Були надані поради, дотримання яких є обов’язковим для зберігання пожежної безпеки на робочому місці.

ВИСНОВКИ

В результаті кваліфікаційної роботи виконано розробку та реалізацію модифікованого програмного забезпечення для приховування додаткової інформації у цифрових зображеннях, з можливістю подальшої передачі цього зображення та відновлення секретної інформації на стороні одержувача, для підвищення захисту інформації, що передається по відкритих каналах зв'язку.

Для досягнення поставленої мети розв'язані наступні задачі.

Описано приховування секретної інформації у цифрових зображеннях, розглянута мета стеганографії та її актуальність на теперішній час. Було надано визначення поняттю стеганографічної системи, докладно розібрано її характерні риси та основні структурні елементи з описанням кожного з них. Представлено аналіз існуючих стеганографічних методів, таких як: дискретне косинусне перетворення у глобальній адаптивній області, різниця модулів коефіцієнтів дискретного косинусного перетворення, вейвлет-перетворення та цілочисельне вейвлет-перетворення з використанням генетичного алгоритму. У результаті такого аналізу представлена характеристика цих методів, щодо їх сильних та слабких сторін. Було проведено огляд практичних реалізацій стеганоалгоритмів, що працюють з зображеннями у форматі з втратами. Розглянуті програмні продукти: Steg, JPHS, Steganography Online Codec.

Представлені теоретичні основи стеганографічного алгоритму, заснованого на сингулярному розкладі блоків матриці цифрового зображення.

Детально досліджено послідовність вбудовування та вилучення секретного повідомлення для початкового стеганографічного алгоритму. Після проведення аналізу ефективності, було виявлено причину низького показника відсотку правильного відновлення секретної інформації для даного алгоритму.

Задля вирішення проблеми було розроблено модифікацію, яка суттєво підвищує його ефективність. У розділі наведена покрокова робота алгоритму з модифікацією.

Для оцінки ефективності модифікованого стеганографічного алгоритму,

проведено обчислювальний експеримент з встановлення надійності сприйняття та відсотка правильно відновленого секретного повідомлення. Практично доведено, що модифікація алгоритму збільшує його ефективність.

Розглянута практична реалізація стеганографічного алгоритму, заснованого на сингулярному розкладі блоків матриці цифрового зображення. Виконано обґрунтування вибору середовища розробки програмного застосування. MATLAB містить необхідний функціонал для реалізації стеганоалгоритму, який розглядається. Також цей функціонал є зручним та зрозумілим для використання, що є безумовно великою перевагою MATLAB, як засобу реалізації програмних застосувань для стеганографічних досліджень, у порівнянні з його конкурентами. Представлена організація роботи програмного застосування. Описано функціонування модулів програмного застосування та їх взаємодія за допомогою діаграми послідовності. Приведена докладна інструкція з використання програмного застосування Описано функціонал та можливості застосування, яке реалізує початковий стеганографічний алгоритм та його версію із модифікацією, що підвищує ефективність. Програмний код реалізованого програмного застосування представлений у додатку А.

ПЕРЕЛІК ПОСИЛАНЬ

1. Грибунин В.Г., Оков И.Н., Туринцев И.В. Цифровая стеганография. М.: Солон-Пресс, 2002. 272с.
2. Аграновский А.В., Балакин А.В., Грибунин В.Г., Сапожников С.А. Стеганография, цифровые водяные знаки и стеганоанализ. М.: Вузовская книга, 2009. 220 с.
3. Конахович Г.Ф., Пузыренко А.Ю. Компьютерная стеганография. Теория и практика. К.: МК-Пресс, 2006. 288 с.
4. Корольов В.Ю., Поліновський В.В., Герасименко В.А., Горінштейн М.Л. Планування досліджень методів стеганографії та стеганоаналізу. *Вісник Хмельницького національного університету*. 2011. №2. С. 187-196.
5. Алиев А.Т., Балакин А.В. Оценка стойкости систем скрытой передачи информации. *Информационная безопасность.* Материалы VII Международной научно-практической конференции, 4-8 липня 2005. Таганрог, 2005. С. 199-204.
6. Кобозєва А.А., Хорошко В.А., Мачалін І.О. Аналіз захищеності інформаційних систем. К.: ДУІКТ, 2010. 316 с.
7. Rabie T., Kamel I. High-capacity steganography: a global-adaptive-region discrete cosine transform approach. *Multimedia Tools and Applications*. 2017. Vol. 76 (5). P 6473-6493.
8. Attaby A., Ahmed M., Alsammak A. Data hiding inside JPEG images with high resistance to steganalysis using a novel technique: DCT-M3. *Ain Shams Eng*. 2018. Vol.9(4). P. 1965-1974.
9. Divya V., Sasirekha N. High capacity steganography technique based on wavelet transform. *Green Engineering and Technologies.* Online International Conference. 2016. P. 1-5.
10. Miri A., Faez K. An image steganography method based on integer wavelet transform. *Multimedia Tools and Applications*. 2018. Vol. 77 (11). P 13133–13144.
11. Fabionet. *Steg*. URL: https://www.fabionet.org/stegnews
12. Acad. *JPHS*. URL: http://io.acad.athabascau.ca/~grizzlie/Comp607/

13. Pelock. *Steganography Online Codec*. URL: https://www.pelock.com/products/steganography-online-codec
14. Козіна М.О., Папковська О.Б. Стеганоалгоритм, що використовує сингулярне розкладання матриці контейнера. *Сучасний захист інформації*. 2018. №2(34). С. 47-52.
15. David H. *Introduction to matlab for engineering students.* URL: https://www.mccormick.northwestern.edu/documents/students/undergraduate/introduction-to-matlab.pdf
16. Mathworks. *MATLAB*. URL: https://www.mathworks.com/products/matlab.html
17. Exponenta. *Matlab Image Processing Toolbox*. URL: https://exponenta.ru/image-processing-toolbox
18. Sciencedirect. *Matlab Image Processing Toolbox Overview*. URL: https://www.sciencedirect.com / topics/computer-science/image-processing-toolbox
19. Exponenta. *Matlab Guide*. URL: https://docs.exponenta.ru/matlab/ref/guide.html
20. Касаяні А.В., Трифонова К.О. Програмні реалізації стеганографічних методів приховування даних у цифрових зображеннях. *Сборник научных трудов «Актуальные научные исследования в современном мире»*. 2022. №5(73), ч.2. С. 75–78.
21. Жидецький В.Ц. Охорона праці користувачів комп’ютерів. Львів, Афіша, 2000. 176 с.
22. ДСН 43.3.6.037-99. Санітарні норми виробничого шуму, ультразвуку та інфразвуку. [Чинний від 2016.06.01]. Київ, 1999. 34с.
23. ДБН В.2.5-28-2006. Природне і штучне освітлення. [Чинний від 2016.06.01]. Київ, 2006. 79с.
24. ДСанПіН 3.3.2.007-98. Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин. [Чинний від 1998.12.10]. Київ, 1998. 65с.

Додаток А

Лістинг програмного коду

```
function varargout = putMessageForm(varargin)

gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @putMessageForm_OpeningFcn, ...
                   'gui_OutputFcn',  @putMessageForm_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end

function putMessageForm_OpeningFcn(hObject, eventdata, handles,
varargin)
handles.output = hObject;
guidata(hObject, handles);

function varargout = putMessageForm_OutputFcn(hObject, eventdata,
handles)

varargout{1} = handles.output;

function upload_Callback(hObject, eventdata, handles)
[fname, pname] = uigetfile('*.bmp;*.jpg;*.jpeg;*.tif;*.tiff','Select
Image');
if fname~=0
    fullname = strcat(pname, fname);
    set(handles.file_name,'String', fullname);
    input_image = imread(fullname);
    %input_image = uigetfile('*.jpg');
    im_info = imfinfo(fullname);
    resolution = strcat(mat2str(im_info.Width), 'x');
    resolution = strcat(resolution, mat2str(im_info.Height));
    set(handles.im_resolution,'String', resolution);
    set(handles.im_size,'String', strcat(mat2str(im_info.FileSize),
' байт'));
    handles.image_name = fullname;
    guidata(hObject,handles);
    axes(handles.input_image);
    imshow(input_image);
end
```

```
function hide_Callback(hObject, eventdata, handles)
image_name = handles.image_name;
input_image = imread(image_name);
rgb = double(input_image);
b = rgb(:,:,3);

editbox_text=get(handles.secret_message,'string');
bin_txt=dec2bin(double(char(editbox_text)));

if(get(handles.check_alg1, 'Value') == 1.0)
    k = 1;
    for i=1:1:size(bin_txt,1)
        for j=1:1:size(bin_txt,2)
          binary_message(k) = bin_txt(i,j);
          k = k+1;
        end
    end
    handles.binary_message = binary_message;
    if size(binary_message) <=
fix(size(input_image,1)*size(input_image,2)/64)
        output_b = [];
        k = 1;
        coef = 1/4;
        power = 1;
        for i=1:8:size(b,1)
            if i+7 > size(b,1)
                break;
            end
            for j=1:8:size(b,2)
                if j+7 > size(b,2)
                    break;
                end
                block = b(i:i+7, j:j+7);
                [u,s,v] = svd(block);

                if k < size(binary_message, 2)

                   if binary_message(k) == '0'
                       coef = 1/4;
                   elseif binary_message(k) == '1'
                       coef = 3/4;
                   end

                   if s(1,1) >= 100
                       power = 2;
                   else
                       power =1;
                   end

                   s(1,1) = (fix((s(1,1)/10^power))*10^power) +
(coef * s(2,2));
                   k = k + 1;
```

```
                end
                output_block = u * s * v';
                output_b(i:i+7, j:j+7) = output_block;
            end
        end
        steg_message1 = rgb;
        if (size(output_b,1)) ~= size(steg_message1(:,:,3), 1) ||
(size(output_b,2)) ~= size(steg_message1(:,:,3), 2)
            output_b(size(output_b,1):size(steg_message1(:,:,3), 1),
size(output_b,2):size(steg_message1(:,:,3), 2)) =
steg_message1(size(output_b,1):size(steg_message1(:,:,3), 1),
size(output_b,2):size(steg_message1(:,:,3), 2));
        end
        steg_message1(:,:,3) = output_b;
        steg_message1 = uint8(steg_message1);
        axes(handles.output_image_alg1);
        imshow(steg_message1);
        handles.steg_message1 = steg_message1;
        guidata(hObject,handles);
    else
        msgbox('Занадто велике повідомлення для обраного
зображення');
    end
end
if(get(handles.check_alg2, 'Value') == 1.0)
    k = 1;
    for i=1:1:size(bin_txt,1)
        for j=1:1:size(bin_txt,2)
          binary_message(k) = bin_txt(i,j);
          k = k+1;
        end
    end
    if size(binary_message) <=
fix(size(input_image,1)*size(input_image,2)/64)
        output_b = [];
        k = 1;
        coef = 1/4;
        power = 1;
        for i=1:8:size(b,1)
            if i+7 > size(b,1)
                break;
            end
            for j=1:8:size(b,2)
                if j+7 > size(b,2)
                    break;
                end
                block = b(i:i+7, j:j+7);
                [u,s,v] = svd(block);

                if k < size(binary_message, 2)

                   if binary_message(k) == '0'
                       coef = 1/4;
```

```
                    elseif binary_message(k) == '1'
                        coef = 3/4;
                    end

                    if s(1,1) >= 100
                        power = 2;
                    else
                        power =1;
                    end

                    sigma_index = 2;
                    while sigma_index <= 8
                        if (0.75 * s(sigma_index,sigma_index) >=
10^power && sigma_index~=8)
                            sigma_index = sigma_index + 1;
                        else
                            s(1,1) = fix((s(1,1)/10^power))*10^power
+ coef * s(sigma_index,sigma_index);
                            break;
                        end
                    end
                    k = k + 1;
                end
                output_block = u * s * v';
                output_b(i:i+7, j:j+7) = output_block;
            end
        end
        steg_message2 = rgb;
        if (size(output_b,1)) ~= size(steg_message2(:,:,3), 1) ||
(size(output_b,2)) ~= size(steg_message2(:,:,3), 2)
            output_b(size(output_b,1):size(steg_message2(:,:,3), 1),
size(output_b,2):size(steg_message2(:,:,3), 2)) =
steg_message2(size(output_b,1):size(steg_message2(:,:,3), 1),
size(output_b,2):size(steg_message2(:,:,3), 2));
        end
        steg_message2(:,:,3) = output_b;
        steg_message2 = uint8(steg_message2);
        axes(handles.output_image_alg2);
        imshow(steg_message2);
        handles.steg_message2 = steg_message2;
        guidata(hObject,handles);
    else
        msgbox('Занадто велике повідомлення для обраного
зображення');
    end
end

% --- Executes on button press in download1.
function download1_Callback(hObject, eventdata, handles)
steg_message1 = handles.steg_message1;
%uisave(steg_message, 'steg_message.png');
%saveas(steg_message, 'steg_message.png');
imwrite(steg_message1, 'steg_message_alg1.png');
```

```
% --- Executes on button press in download2.
function download2_Callback(hObject, eventdata, handles)
steg_message2 = handles.steg_message2;
%uisave(steg_message, 'steg_message.png');
%saveas(steg_message, 'steg_message.png');
imwrite(steg_message2, 'steg_message_alg2.png');

% --- Executes on button press in psnr.
function psnr_Callback(hObject, eventdata, handles)
image_name = handles.image_name;
input_image = imread(image_name);

output_image1 = handles.steg_message1;
output_image2 = handles.steg_message2;

% [peaksnr1, snr1] = psnr(output_image1, input_image);
% [peaksnr2, snr2] = psnr(output_image2, input_image);


A=output_image1(:);
B=input_image(:);
[rows columns] = size(A);
squaredErrorImage = (double(A) - double(B)) .^ 2;
mse = sum(sum(squaredErrorImage)) / (rows * columns);
PSNR1 = 10 * log10( 256^2 / mse);

A=output_image2(:);
B=input_image(:);
[rows columns] = size(A);
squaredErrorImage = (double(A) - double(B)) .^ 2;
mse = sum(sum(squaredErrorImage)) / (rows * columns);
PSNR2 = 10 * log10( 256^2 / mse);

pick = [PSNR1 PSNR2];
labeles = categorical({'PSNR ALG1','PSNR ALG2'});

figure('Name','PSNR','NumberTitle','off');
bar(labeles, pick, 0.4);

% --- Executes on button press in recovery_percent.
function recovery_percent_Callback(hObject, eventdata, handles)
binary_message = handles.binary_message;
%--------------------------
stegomessage1 = handles.steg_message1;
rgb = double(stegomessage1);
b = rgb(:,:,3);
p = 1;
for i=1:8:size(b,1)
    if i+7 > size(b,1)
        break;
    end
    for j=1:8:size(b,2)
```

```
        if j+7 > size(b,2)
            break;
        end
    block = b(i:i+7, j:j+7);
    [u,s,v] = svd(block);

    if s(1,1) >= 100
           power = 2;
       else
           power = 1;
    end

    if s(1,1)-(fix(s(1,1)/10^power)*10^power) < 1/2*s(2,2)
        recovered_binary_message1(p) = '0';
    else
        recovered_binary_message1(p) = '1';
    end

    p = p + 1;
end
end
%---------------------------
stegomessage2 = handles.steg_message2;
rgb = double(stegomessage2);
b = rgb(:,:,3);
p = 1;
for i=1:8:size(b,1)
    if i+7 > size(b,1)
        break;
    end
    for j=1:8:size(b,2)
        if j+7 > size(b,2)
            break;
        end
    block = b(i:i+7, j:j+7);
    [u,s,v] = svd(block);

    if s(1,1) >= 100
           power = 2;
       else
           power = 1;
    end

    sigma_index = 2;
    while sigma_index <= 8
        if (0.75 * s(sigma_index,sigma_index) >= 10^power)
            sigma_index = sigma_index + 1;
        else
            if s(1,1)-(fix(s(1,1)/10^power)*10^power) <
1/2*s(sigma_index,sigma_index)
                recovered_binary_message2(p) = '0';
            else
                recovered_binary_message2(p) = '1';
```

```
            end
            break;
        end
    end
        p = p + 1;
    end
end
%--------------------------

success1 = 0;
success2 = 0;
for i=1:1:size(binary_message, 2)
    if(recovered_binary_message1(i) == binary_message(i))
        success1 = success1 + 1;
    end
     if(recovered_binary_message2(i) == binary_message(i))
        success2 = success2 + 1;
     end
end
persent1 = success1 / size(binary_message, 2);
persent2 = success2 / size(binary_message, 2);

persent = [persent1*100 persent2*100];
labeles = categorical({'ALG1','ALG2'});
figure('Name','Відсоток відновлення','NumberTitle','off');
bar(labeles, persent, 0.4);
```