Міністерство освіти і науки України
Кам’янець-Подільський національний університет імені Івана Огієнка
Фізико-математичний факультет
Кафедра комп’ютерних наук

Дипломна робота
магістра

з теми: «**РОЗРОБКА МЕТОДУ ПРИХОВАННЯ ІНФОРМАЦІЇ В МУЛЬТИМЕДІЙНОМУ ПОТОЦІ ДЛЯ ПІДВИЩЕННЯ ЇЇ БЕЗПЕКИ**»

Виконав: студент 1 курсу,
групи KN1-M22
спеціальності 122 Комп’ютерні науки
Ткачук Валерій Віталійович

Кам’янець-Подільський – 2023

ЗМІСТ

ВСТУП

Сьогоднішній світ, насичений мультимедійною інформацією, ставить перед нами виклик забезпечення безпеки цієї інформації в умовах постійно зростаючого числа кіберзагроз та стеганографічних атак. Захист інформації стає важливим завданням для організацій, установ і індивідуальних користувачів, оскільки конфіденційні дані, такі як особиста інформація, комерційні та фінансові документи, медичні записи та інші чутливі дані, потребують надійного захисту.

Разом зі зростанням загроз, постають нові можливості для розробки ефективних методів приховування інформації в мультимедійних потоках. Приховування інформації в мультимедійних данних виявляється як перспективний підхід, оскільки ці дані масово використовуються в сучасному світі, наприклад, в цифрових зображеннях, аудіо та відеофайлах. Приховування інформації може бути використано для непомітного передачі та збереження конфіденційної інформації, уникнення виявлення та забезпечення безпеки передачі цих даних через відкриті мережі.

Актуальність даної теми полягає в тому, що зростаюче значення мультимедійної інформації та загрози кібербезпеки ставлять під загрозу конфіденційність та цілісність даних. Забезпечення безпеки цих даних стає надзвичайно важливим завданням. Розробка нових та вдосконалення існуючих методів приховування інформації в мультимедійних потоках стають необхідністю для забезпечення безпеки передачі та збереження конфіденційної інформації. Ця тема вимагає постійного дослідження та розробки нових алгоритмів та технологій, що сприятимуть покращенню безпеки мультимедійних даних.

Метою даної магістерської роботи є висвітлення новітніх методів приховування інформації в мультимедійних потоках та покращення та оптимізація вже існуючих засобів приховування інформації в

мультимедійному потоці з метою підвищення безпеки цієї інформації. Результати цього дослідження сприятимуть створенню більш стійких систем захисту, здатних впоратися зі сучасними кіберзагрозами та стеганографічними атаками.

Для досягнення поставленої мети, у даній роботі будуть вирішені наступні завдання:

1. Аналіз існуючих методів приховування інформації в мультимедійних потоках: проведення огляду різних методик та алгоритмів, виявлення їх переваг та недоліків.
2. Оптимізування існуючого методу приховування інформації в мультимедійному потоці: розроблення нового підходу, що базується на використанні передових технологій та алгоритмів, які забезпечують високу стійкість та непомітність приховування інформації.
3. Дослідження ефективності розробленого методу в мультимедійному потоці: проведення експериментальних випробувань для оцінки ефективності та надійності розробленого методу.
4. Оцінка стійкості розробленої моделі оптимізації приховування інформації до різних видів атак: проведення аналізу та тестування стійкості розробленого методу перед відомими атаками, такими як стеганаліз та атаки з використанням статистичних методів.
5. Порівняння розробленого методу з існуючими методами приховування інформації в мультимедійних потоках: виконання порівняльного аналізу розробленого методу з найбільш поширеними та відомими методами приховування інформації з метою визначення його переваг та обмежень.
6. Аналіз можливостей застосування розробленого методу приховування інформації у реальних сценаріях: дослідження потенційних сфер застосування розробленого методу та оцінка його ефективності та придатності у реальних умовах.

Об'єктом роботи є методи приховування інформації в мультимедійних потока

Предметом роботи є розробка методу оптимізації приховування інформації в мультимедійних потоках для підвищення безпеки.

Новизна роботи полягає у розробці унікального методу оптимізації існуючих методів приховування інформації в мультимедійних потоках. Цей метод включає в себе вдосконалення алгоритмів приховування та виявлення інформації, що дозволяє підвищити ефективність та стійкість процесу приховування.

РОЗДІЛ 1. ТЕОРЕТИЧНІ АСПЕКТИ ПРИХОВУВАННЯ ІНФОРМАЦІЇ В МУЛЬТИМЕДІЙНИХ ПОТОКАХ

1.1 Поняття і визначення приховування інформації в мультимедійних потоках

Поняття і визначення приховування інформації в мультимедійних потоках можна розділити на кілька важливих пунктів:

- Приховування інформації є складним процесом, спрямованим на вбудовування конфіденційної або важливої інформації в мультимедійний потік. Це може бути виконано візуально (для зображень), акустично (для аудіо) або іншими способами в залежності від типу мультимедійних даних. Головною метою є збереження цілісності та конфіденційності інформації.
- Мультимедійні потоки представляють собою набір даних у різних форматах, таких як графіка, аудіо та відео, які можна передавати та обробляти. Ці потоки є основною формою представлення мультимедійної інформації та використовуються для сприйняття та обміну мультимедійним вмістом.
- Підвищення безпеки мультимедійних потоків передбачає використання методів та технік для забезпечення конфіденційності, цілісності та доступності інформації. Це може включати криптографічне шифрування, контроль цілісності та приховування інформації для запобігання несанкціонованому доступу та модифікації.
- Методи приховування інформації спрямовані на вбудовування додаткової інформації (прихованого тексту або даних) в мультимедійний потік так, щоб це було непомітно. для незаперечного спостерігача. Ці методи можуть використовувати різні техніки, такі як методи

стеганографії, які базуються на властивостях мультимедійних даних для ефективного приховування інформації.

Цей розділ визначає основні концепції, які досліджуються у магістерській роботі. Детальніше буде розглянуто кожну з цих концепцій у наступних розділах роботи. Вивчення цих аспектів є ключовим для розробки методу приховування інформації в мультимедійних потоках для забезпечення високого рівня безпеки та конфіденційності.

1.2 Класифікація методів приховування інформації в мультимедійних потоках

Класифікація методів приховування інформації в мультимедійних потоках важлива для визначення та розрізнення різних підходів та стратегій, які використовуються для ефективного вбудовування прихованої інформації. При розгляді класифікації слід врахувати такі ключові критерії: характер приховуваної інформації, метод вбудовування та природа мультимедійного контенту.

Характер приховуваної інформації:

Перш за все, методи приховування можна класифікувати залежно від характеру приховуваної інформації:

1. Текстова прихована інформація: Деякі методи призначені для вбудовування текстової інформації в мультимедійні дані. Це може бути прихований текст, код, секретне повідомлення тощо.
2. Аудіо-прихована інформація: Інші методи спрямовані на вбудовування аудіо-інформації в звукові дані, такі як музика або звукові ефекти.
3. Відео-прихована інформація: Ця категорія включає методи, які дозволяють приховувати інформацію в відеоданих, зокрема відео-потоках.
4. Бінарна прихована інформація: Деякі методи дозволяють вбудовувати бінарні дані (наприклад, файли) в мультимедійний контент.

Метод вбудовування:

Другий критерій класифікації враховує методи вбудовування прихованої інформації:

1. Зайвий простір: Деякі методи використовують техніку виділення невикористаного або зайвого простору в мультимедійних даних для вбудовування прихованої інформації.
2. Модифікація бітів: Інші методи базуються на зміні менш значущих бітів у певних частинах мультимедійних даних для приховування інформації.
3. Природа мультимедійного контенту:

Третій аспект класифікації враховує природу самого мультимедійного контенту:

1. Зображення: Особливі методи можуть бути оптимізовані для роботи з різними типами зображень, включаючи фотографії, малюнки та графіку.
2. Аудіо: Деякі методи спеціалізуються на роботі з аудіоданими, такими як музика, голосові повідомлення та інші звукові файли.
3. Відео: Ця категорія включає методи, призначені для роботи з відео-даними, де можна приховувати інформацію в кадрах та потоках відео.

Ця класифікація дозволяє враховувати різноманітність методів приховування інформації в мультимедійних потоках та сприяє кращому розумінню їх ефективності та застосування у забезпеченні безпеки та конфіденційності даних

1.3 Аналіз існуючих методів приховування інформації в мультимедійних потоках

Існує багато методів приховування інформації в мультимедійних потоках, кожен з яких має свої особливості та використовується у різних сферах. Розглянемо деякі основні методи:

1. LSB (Least Significant Bit) стеганографія:

LSB стеганографія використовує найменш значущі біти (LSB) в пікселях зображень або аудіоданих для приховування інформації. Цей метод вважається одним з найпоширеніших і простих. В зображеннях кожен канал (червоний, зелений, синій) має свої найменш значущі біти, які можна використовувати для приховування. У відео та аудіо аналогічно можна використовувати найменш значущі біти кожного кадру чи вибірки.

2. Transform domain стеганографія:

Цей підхід полягає в використанні перетворень, таких як Fourier, Wavelet або DCT (Discrete Cosine Transform), для приховування інформації у домені коефіцієнтів. Наприклад, використання малозначущих коефіцієнтів після DCT у відео або зображеннях.

3. Аудіо стеганографія:

Цей метод зосереджений на приховуванні інформації у звукових сигналах. Можна використовувати методи, які адаптовані специфічно для аудіо, включаючи зміну амплітуди сигналу, шумові компоненти та інші параметри.

4. Відео стеганографія:

Аналогічно аудіо стеганографії, цей метод спеціалізується на вбудовуванні інформації в відеодані. Можна використовувати пікселі, кадри або коефіцієнти стиснення відео.

5. Методи маскування:

Ці методи базуються на використанні властивостей людського сприйняття для приховування інформації. Можна використовувати артефакти маскування (наприклад, додавання шуму або розмиття) для приховування інформації.

6. Криптографічні методи:

Ці методи використовують криптографічні алгоритми для шифрування та приховування інформації у мультимедійних данних. Шифрування дозволяє забезпечити конфіденційність та безпеку прихованої інформації.

Однією з ключових характеристик методів приховування є їхній рівень стійкості до атак та здатність зберігати приховану інформацію після обробки, стискання чи інших маніпуляцій над мультимедійним контентом. Аналізується, як вони впливають на якість та надійність прихованого повідомлення в різних умовах. **[3]**

Стійкість до атак та здатність зберігати приховану інформацію після обробки, стискання чи інших маніпуляцій над мультимедійним контентом є ключовими аспектами ефективності методів приховування інформації в мультимедійних потоках.

1. Стійкість до атак:

Стійкість до атак визначає, наскільки ефективно прихована інформація залишається непорушеною при впливі зовнішніх втручань. Атаки можуть включати спроби виявлення, зміни або вилучення прихованої інформації.

1. Статистичні аналізи: Аналіз статистичних характеристик мультимедійних даних може виявити відмінності, які свідчать про наявність прихованої інформації.
2. Криптоаналіз: Атаки, спрямовані на розшифрування або злам криптографічних захистів, можуть розкрити приховану інформацію, якщо вона була зашифрована.
3. Атаки на канал зв'язку: Спроби спотворити канал передачі даних, що може привести до втрати прихованої інформації.

Методи приховування повинні бути стійкими до цих атак, забезпечуючи непорушення прихованої інформації.

2. Збереження прихованої інформації після обробки, стискання та інших маніпуляцій:

Збереження прихованої інформації після різних обробок є важливою властивістю методів приховування. Обробка може включати стискання (наприклад, JPEG-стискання для зображень), фільтрацію, зміну формату або інші маніпуляції над мультимедійним контентом.

Стійкість до стискання: Деякі методи повинні бути стійкими до втрати прихованої інформації під час стиснення мультимедійного контенту. Це особливо важливо для методів, які вбудовують інформацію в менш значущі біти.

Збереження прихованої інформації після обробки: Методи повинні зберігати приховану інформацію після обробки контенту. Це означає, що навіть після фільтрацій, кольорових змін або інших обробок, прихована інформація має залишатися доступною та невідомою для незаперечного спостерігача.

Методи, які успішно забезпечують ці вимоги, вважаються більш ефективними та практично застосовними в сучасних умовах обробки та обміну мультимедійною інформацією. Безпека та стійкість методів є

основною метою їхнього використання в захисті конфіденційності та передачі прихованої інформації. [Таблиця 1.1]

Таблиця 1.1

Методи	**Вразливість**	**Переваги**	**Техніка приховування даних**
Менший значущий біт (LSB)	Легко вилучити	Простий та легкий для приховування інформації	Найменший значущий біт кожного зразка звуку вбудовується бітом прихованої інформації
Приховування за допомогою ехо	Низька безпека інформації та низька місткість	Без проблем додаткового шуму	приховування інформації з введенням ехо сигналу прикриття
Балансове кодування	Легко вилучити	стійкіший, ніж LSB	Зміна LSB балансу біту вибірки
Дискретне хвильове перетворення	Відновлення даних із втратою	Забезпечення великої ємності та чіткості	Зміна коефіцієнта хвильового перетворення для приховування інформації
Розподілений спектр	Більша пропускна здатність	хороша стійкість і підвищення чіткості	Розподілення інформації під усі частоти сигналу
Розроблений метод	Низька ємність	Стабільний щодо опцій обробки сигналу	Зміна фази сигналу

Методи приховування інформації в мультимедійних потоках мають широкий спектр застосувань у різних галузях та сферах життя. Розглянемо основні сфери та конкретні випадки застосування цих методів:

1. Конфіденційність та безпека даних:

Методи приховування інформації можуть бути використані для забезпечення конфіденційності та безпеки даних. Наприклад, приховування конфіденційних даних у мультимедійних файлових контейнерах може допомогти уникнути їхнього неправомірного доступу.

2. Водяні знаки (Watermarking):

Водяні знаки — це техніка приховування інформації у мультимедійних об'єктах, яка використовується для ідентифікації та захисту авторських прав. Наприклад, водяні знаки можуть бути вбудовані у зображення або відео для позначення авторства та уникнення піратства. [12]

3. Стеганографія та приховане спілкування:

Стеганографія використовується для приховування конфіденційних повідомлень, а не просто для їхнього шифрування. Це може бути важливим у військових, політичних або кримінальних ситуаціях для забезпечення прихованого спілкування.

4. Стеганографічне спілкування у кібербезпеці:

Стеганографія може бути використана для прихованого обміну інформацією у кібербезпеці. Наприклад, приховування команд або конфіденційної інформації у мережевих пакетах, аудіо- або відеофайлах для передачі через незахищені канали.

5. Медичні застосування:

Методи приховування інформації можуть бути використані у медичній сфері для приховування та передачі конфіденційних пацієнтських даних у медичних зображеннях або сигналах.

6. Відеоспостереження та безпека:

У системах відеоспостереження методи приховування можуть бути використані для приховування додаткової інформації, такої як дата та час, місцезнаходження або інші параметри камери, безпечності та контролю.

7. Сховані повідомлення та соціальні мережі:

Методи приховування можуть бути використані для обміну прихованими повідомленнями у соціальних мережах або комунікаційних платформах для збереження конфіденційності та забезпечення безпеки обміну даними.

Ці застосування вказують на важливість методів приховування інформації в мультимедійних потоках у сучасному світі та необхідність досліджень та розробки нових методів для забезпечення конфіденційності, безпеки та ефективності обробки мультимедійних даних.

1.4 Огляд стеганографічних атак на мультимедійні потоки

Стеганографічні атаки на мультимедійні потоки спрямовані на виявлення прихованої інформації або спотворення цілісності прихованого повідомлення. Ці атаки загрожують безпеці та конфіденційності прихованої інформації. Розглянемо деякі типові стеганографічні атаки:

1. Статистичний аналіз

Ця атака полягає у виявленні незвичайних статистичних властивостей мультимедійних даних, які можуть свідчити про приховану інформацію. Наприклад, надмірна концентрація деяких значень пікселів у зображенні або особливі характеристики аудіосигналу можуть вказувати на наявність прихованого повідомлення.

Також статистичний аналіз є одним з основних методів аналізу стеганографічних атак, спрямованих на виявлення та аналіз прихованої інформації в мультимедійних даних. Цей аналіз базується на аналізі статистичних характеристик даних та їхніх змін при вбудовуванні прихованої інформації. [5]

1.1 Аналіз гістограми

Статистичний аналіз гістограми мультимедійних даних (наприклад, зображення) може виявити незвичайні розподіли значень пікселів, які можуть свідчити про вбудовування прихованої інформації. Надмірні піки або аномальні зміни можуть бути показниками наявності прихованої інформації.

1.2 Аналіз кореляції

Аналіз кореляції між пікселями чи коефіцієнтами у мультимедійних даних може розкрити зміни, які виникли в результаті вбудовування прихованої інформації. Значний розрив у кореляції може бути показником змін у структурі даних.

1.3 Аналіз ентропії

Ентропія є мірою не визначиності інформації. Зміни в ентропії мультимедійних даних після вбудовування прихованої інформації можуть бути використані для виявлення наявності прихованої інформації.

1.4Статистичні тестів на нормальність

Деякі стеганографічні методи можуть спотворити нормальний розподіл значень вихідних даних. Використання статистичних тестів на нормальність може виявити такі спотворення.

1.5Частотний аналіз

Цей вид аналізу включає аналіз частотних складових у мультимедійних даних. Аномалії у частотних діапазонах можуть бути показниками вбудовування прихованої інформації.

1.6Аналіз автокореляції

Аналіз автокореляції дозволяє виявити закономірності у взаємній залежності між елементами даних. Аномалії у цьому аналізі можуть бути результатом вбудовування прихованої інформації.

Статистичний аналіз є ефективним інструментом для виявлення прихованої інформації у мультимедійних даних. Аналізуючи статистичні властивості даних, можна виявити незвичайність, яка може бути пов'язана з наявністю прихованої інформації, та вжити відповідних заходів для забезпечення безпеки та конфіденційності мультимедійних даних.

2. Шумові атаки

Шумові атаки є одним з методів атаки на стеганографічні системи та методи приховування інформації. Ці атаки спрямовані на спотворення або виявлення прихованого повідомлення, вбудованого в мультимедійні дані. Шум може бути доданий до мультимедійних даних для спотворення прихованої інформації та ускладнення її виявлення.

2.1Додавання шуму

Ця атака включає додавання шуму до мультимедійних даних, щоб змінити їхню структуру та приховану інформацію. Шум може бути доданий в

різних формах, таких як адитивний білий гаусівський шум або імпульсний шум.

2.2Атаки з застосуванням фільтрація

Атаки фільтрації включають застосування різних фільтрів до мультимедійних даних. Це може призвести до вилучення або спотворення прихованого повідомлення.

2.3Амплітудні атаки

Ці атаки спрямовані на зміну амплітуди аудіосигналу або інших параметрів мультимедійних даних. Зміна амплітуди може призвести до спотворення та невиявлення прихованої інформації.

2.4Розмиття

Атаки розмиття включають використання фільтрів або обробок, що призводять до розмиття частини або всього зображення. Це може спотворити вбудоване повідомлення та зробити його важким для виявлення.

Отже, шумові атаки є досить поширеними у контексті стеганографії, оскільки вони можуть суттєво ускладнити виявлення прихованого повідомлення та знизити його якість. Розробка методів приховування, які стійкі до таких шумових атак, та алгоритмів аналізу, які можуть виявляти приховану інформацію навіть у шумному середовищі, є важливим завданням для забезпечення ефективності та надійності стеганографічних систем.

3. Атаки з вилученням

Атаки з вилученням (англ. extraction attacks) є спробами вилучити приховану інформацію з мультимедійних даних, в яких ця інформація була прихована за допомогою стеганографічних методів. Ці атаки можуть бути спрямовані на вилучення інформації без знання ключа, який використовувався для вбудовування, або на злам ключа та вилучення інформації з метою декриптації.

Важливо розрізняти атаки з вилученням прихованої інформації від атак на саму систему стеганографії. Останні атаки спрямовані на розкриття або

компрометацію самої стеганографічної системи (наприклад, розкриття алгоритму вбудовування).

3.1Вилучення без ключа

У цьому випадку злоумисник намагається вилучити приховану інформацію з мультимедійних даних, не знаючи ключа, який був використаний для вбудовування. Ця атака спрямована на отримання прихованої інформації без належного авторизованого доступу.

3.2Вилучення з використанням ключа

У цьому випадку злоумисник має доступ до ключа, який використовувався для вбудовування прихованої інформації. Він використовує цей ключ для вилучення інформації з мультимедійних даних.

3.3Вилучення та дешифрування

Ця атака передбачає вилучення прихованої інформації та її подальше дешифрування, якщо інформація була зашифрована перед вбудовуванням.

3.4Вилучення зі спотворенням

У цьому випадку злоумисник намагається вилучити приховану інформацію, спотворюючи її або змінюючи деякі її частини. Це може спричинити некоректне вилучення та отримання невірної інформації.

3.5Перетворення в стеганографічний атакувальний канал

Атаки можуть спрямовуватися на перетворення звичайного каналу зв'язку на атакувальний канал, який забезпечує вилучення прихованої інформації та її передачу злоумиснику.

Для захисту від атак з вилученням, стеганографічні системи повинні бути розроблені з урахуванням високого рівня стійкості та надійності. Використання сильних криптографічних алгоритмів, складних ключів та методів приховування може ускладнити атаки та забезпечити безпеку при обробці та передачі конфіденційної інформації.

4. Атаки на примітиви

Атаки на примітиви є одним з видів атак на стеганографічні системи, спрямованих на зміну примітивних даних для спотворення вбудованої прихованої інформації. Ці атаки можуть впливати на розмір, обертання, масштабування та інші параметри зображень чи відео.

Розглянемо деякі типові геометричні атаки:

2.1Зміна масштабу

Атака полягає в зміні масштабу мультимедійних даних. Це може призвести до спотворення вбудованої інформації та зробити її складнішою для виявлення.

2.2Обертання

Атака включає обертання зображення чи відео на певний кут. Це може спотворити приховане повідомлення та ускладнити його виявлення.

2.3Зміна розміру

Ця атака передбачає зміну розміру зображення чи відео. Зміна розміру може викликати спотворення прихованої інформації та ускладнити виявлення.

2.4Зсув (панорамування)

Атака включає зсув зображення або відео в певному напрямку. Це може призвести до спотворення прихованої інформації та ускладнити її виявлення.

2.5Лінеаризація

Ця атака може виникнути при спробі представити мультимедійні дані у вигляді лінійної послідовності. Лінеаризація може спричинити втрату геометричних характеристик та спотворення вбудованої інформації.

Геометричні атаки є важливими з точки зору забезпечення безпеки та надійності стеганографічних систем. Методи приховування повинні бути розроблені з урахуванням можливих геометричних атак та забезпечувати стійкість до них, щоб зберегти надійність та конфіденційність прихованої інформації.

5. Атаки з вилученням реєстрації

Атаки з вилученням реєстрації спрямовані на вилучення прихованої інформації з мультимедійних даних, зокрема зображень або відео, з використанням маніпуляцій з реєстрацією. Реєстрація в цьому контексті вказує на точне узгодження або вирівнювання двох або більше даних, які можуть бути отримані з різних джерел чи в різні моменти часу.

Розглянемо деякі типові аспекти та методи атак з вилученням реєстрації:

5.1 Введення артефактів

Атаки можуть включати введення штучних артефактів або змін у мультимедійні дані, які порушують правильну реєстрацію між різними джерелами. Це може спотворити обробку та виявлення прихованої інформації.

5.2 Зсув та спотворення реєстрації

Ця атака спрямована на систематичний зсув або спотворення точності реєстрації між даними. Це може зробити важчим виявлення прихованої інформації, яка залежить від точного узгодження.

5.3Вилучення зареєстрованих даних

У цьому випадку злоумисник намагається вилучити або модифікувати інформацію, яка була розміщена у вигляді зареєстрованих або вирівняних областей даних. Це може спричинити втрату прихованого повідомлення або його спотворення.

5.4Зміна частоти кадрів (для відео)

Атака може включати зміну частоти кадрів між різними джерелами відеоданих. Це може спричинити розрив у реєстрації та ускладнити виявлення прихованої інформації.

5.5Зміна часу зйомки (для відео)

Ця атака спрямована на зміну часу зйомки між різними джерелами відеоданих. Це може призвести до невірного вирівнювання та спотворення прихованої інформації.

Для захисту від атак з вилученням реєстрації, стеганографічні системи повинні бути проектовані з урахуванням цих можливих атак. Стеганографічні методи повинні бути стійкими до вищезгаданих видів маніпуляцій та забезпечувати надійну приховану передачу інформації.

6 Перетворення зображень та відео:

Маніпуляції зображеннями або відео, такі як компресія, стиснення або конвертація формату, можуть спотворити вбудоване повідомлення та ускладнити його виявлення.

7 Атаки на формати та метадані:

Зміна метаданих файлу, таких як дата, час зйомки, апаратна модель, GPS-координати тощо, може спричинити спотворення прихованої інформації.

Методи приховування повинні бути стійкими до цих атак та забезпечувати надійність та конфіденційність прихованої інформації навіть в умовах можливих маніпуляцій та спроб спотворення. Врахування цих можливих загроз є важливим при проектуванні та використанні

РОЗДІЛ 2. РОЗРОБКА МЕТОДУ ДЛЯ ОПТИМІЗАЦІЇ ІСНУЮЧОГО ПІДХОДУ ДО ПРИХОВУВАННЯ ІНФОРМАЦІЇ В МУЛЬТИМЕДІЙНИХ ПОТОКАХ

2.1 Побудова математичної моделі розробленого методу

В сучасному інформаційному суспільстві, де конфіденційність та захист особистої інформації набувають важливості, розробка ефективних методів стеганографії стає актуальною задачею. У цьому контексті, розділ математична модель розробленого оптимізаційного застосунку виокремлюється як ключовий етап наукового дослідження, яка визначає теоретичні основи та алгоритмічні принципи розробленого методу. Ретельне дослідження математичної структури надає можливість кращого розуміння ефективності та стійкості запропонованого підходу до приховування інформації в носії.

Отже для початку у фур'є-аналізі сигнал у часовому домені може бути представлений у частотному домені за допомогою перетворення Фур'є. Для дискретного сигналу це часто реалізується за допомогою Дискретного Фур'є-перетворення (DFT). DFT розкладає сигнал на суму синусоїдальних компонент, кожна з яких характеризується своєю амплітудою та фазою.

В контексті коду розробленого додатку:

"s" - це матриця, де кожний стовпець представляє собою сегмент аудіосигналу.

"w" - результат застосування FFT (швидке перетворення Фур'є) до кожного сегмента, що веде до матриці комплексних чисел, де кожний елемент представляє амплітуду та фазу конкретної частотної компоненти.

Фазова матриця Phi отримується шляхом взяття аргументу (кутового значення) комплексних чисел у матриці "w". Кожен стовпець матриці Phi представляє інформацію про фазу відповідного сегмента аудіосигналу.

Потім код розраховує різниці фаз, позначені як DeltaPhi, між сусідніми сегментами. Це досягається шляхом віднімання фазової матриці попереднього сегмента від фазової матриці поточного сегмента.

Давайте представимо фазові матриці для двох сусідніх сегментів як Φ_{k-1}та Φ_k. Матриця різниці фаз $\Delta\Phi_k$ розраховується наступним чином:

$$\Delta\Phi_k = \Phi_k - \Phi_{k-1} \tag{2.1}$$

Це представляє зміну фази від одного сегмента до наступного. Різниці фаз є важливими, оскільки вони відображають, як еволюціонує фаза кожної частотної компоненти з часом. Змінюючи ці різниці фаз, код може вбудовувати інформацію в аудіосигнал, практично не змінюючи амплітуду.

Загалом, різниці фаз фіксують динамічні зміни фази сигналу від одного сегмента до іншого, надаючи основу для стеганографічної техніки кодування фаз.

У наступный частині коду бінарні дані (повідомлення, яке потрібно сховати) відображаються на послідовність значень фази. Схема кодування, що використовується тут, представляє «0» як $\frac{\pi}{2}$ та «1» як $-\frac{\pi}{2}$. Якщо розібрати процес то

Ініціалізація:

PhiData - це вектор, ініціалізований нулями, його довжина визначається змінною m, яка представляє довжину бінарного повідомлення.

Перетворення двійкових даних у фазові значення

Код потім проходить через кожен біт бінарного повідомлення (даних) та присвоює відповідне значення фази відповідному елементу PhiData. Якщо біт - '0', значення фази встановлюється як $\frac{\pi}{2}$. Якщо біт - '1', значення фази встановлюється як $-\frac{\pi}{2}$

Так, кожен біт у бінарному повідомленні асоціюється з конкретним значенням фази, і ці значення фаз будуть використовуватися для зміни фазової інформації аудіосигналу в наступних етапах коду.

Ініціалізація нової фазової матриці:

Phi_new - це матриця, яка буде використовуватися для зберігання зміненої фазової інформації. Перший стовпець Phi_new ініціалізується фазовою інформацією з оригінального сигналу (Phi).

Вбудовування даних

Код потім замінює частину оригінальних значень фаз значеннями фаз, що виводяться з бінарного повідомлення (PhiData). Заміна відбувається в середині першого стовпця Phi_new. Зокрема, значення фаз в діапазоні від $\frac{L}{2} - m + 1$ до $\frac{L}{2}$ замінюються значеннями у PhiData.

Симетрія

Код використовує симетрію для дзеркального відображення значень фаз. Це важливо для збереження властивостей симетрії, необхідних для величини сигналу, що має дійсні значення.

Наступний рядок (Phi_new(L/2+1+1:L/2+1+m, 1) = -flip(PhiData);) дзеркально відображає значення фаз PhiData і розміщує їх на відповідних позиціях для досягнення симетрії.

Математично цей процес можна описати так:

$$\Phi_{new} = \left(\frac{L}{2} - m + 1 : \frac{L}{2}, 1\right) = \text{ Значення фази, отримані з } \backslash PhiData \tag{2.2}$$

$$\Phi_{new} = \left(\frac{L}{2} + 1 + 1 : \frac{L}{2} + 1 + m, 1\right) = \text{Дзеркальні значення фази } \backslash PhiData \tag{2.3}$$

Вбудування фазових даних в середину першого стовпця матриці має на меті забезпечити поступове внесення змін та уникнення впливу на весь сигнал одразу. Це може допомогти зберігати непомітність, оскільки різкі зміни можуть бути помітними.

Відтворення матриць

Перехід до сегментів

Код використовує цикл for для ітерації по сегментах аудіосигналу. Цикл починається з другого сегмента (k = 2), оскільки перший стовпець Phi_new вже був змінений.

Відтворення матриці

Для кожного сегмента код оновлює значення фаз в матриці Phi_new, додаючи відповідні різниці фаз з DeltaPhi. Використовується формула:

$$\Phi_{new} = (:, k) = \Phi_{new}\,(:, k-1) + \Delta\Phi(:, k) \tag{2.4}$$

В математичних термінах для кожного сегмента k значення фаз оновлюються на основі накопиченої суми різниць фаз від попередніх сегментів.

Цей крок є ключовим для збереження цілісності аудіосигналу. Змінена фазова інформація поступово застосовується до кожного сегмента, забезпечуючи плавний перехід і збереження характеристик оригінального сигналу. Мета полягає в тому, щоб вбудувати приховану інформацію, мінімізуючи сприйнятливий вплив на якість аудіо.

Після цього циклу матриця Phi_new містить повністю відновлену фазову інформацію для кожного сегмента аудіосигналу.

Зворотне FFT (IFFT)

Для перетворення модифікованих матриць амплітуди (A) та фази (Phi_new) назад у часовий домен використовується обернене перетворення Фур'є (IFFT).

Формула, що використовується тут, базується на формулі Ейлера:

$$z = A \cdot e^{i\Phi_{new}} \tag{2.5}$$

Де $e^{i\Phi_{new}}$ представляє комплексний експонент із фазовою інформацією.

Функція ifft обчислює обернене FFT для перетворення інформації у частотному домені назад у часовий домен.

Реконструкція сигналу

Результатом оберненого перетворення Фур'є є комплексний часовий сигнал z. Взяття реальної частини (real(z)) гарантує, що сигнал має дійсні значення.

Операція зміни форми (reshaping) (reshape(z, N*L, 1)) виконується для перетворення сигналу назад у стовпчиковий вектор з відповідними розмірами.

Нарешті, відновлений сигнал s_new отримується, додаючи його до залишкової частини оригінального сигналу (plain(N*L+1:I)).

Для декодування данних виконуються наступні кроки:

Перевірка вхідних аргументів:

Код використовує функцію nargin, щоб перевірити кількість аргументів, які надаються функції phase_dec. nargin повертає кількість вхідних аргументів, які функція отримала під час виконання.

Встановлення значення за замовчуванням для L:

Умовний оператор if nargin < 3 перевіряє, чи кількість вхідних аргументів менше 3.

Якщо ця умова виконується, це означає, що не надано необов'язковий аргумент L (довжина фреймів) при виклику функції.

У межах цієї умови L = 1024; встановлює значення за замовчуванням для L рівним 1024.

Обчислення довжини послідовності бітів (m):

m обчислює загальну довжину послідовності бітів на основі вказаної довжини прихованого повідомлення (L_msg).

$$m = 8 \times L_{msg} \tag{2.6}$$

Де L_{msg} - довжина прихованого повідомлення. Враховується 8 бітів на символ, отже, множення на 8.

Загалом цей розділ ініціалізації забезпечує наявність функції значення за замовчуванням для параметра L, якщо його не вказано при виклику функції. Параметр L_msg використовується для визначення довжини послідовності бітів (m), яка в подальшому використовується у процесі декодування.

Далі відбувається вилучення першого сегмента (x):

x = signal(1:L, 1) вилучає перші L відліки з прихованого сигналу. , 1 вказує, що відбувається вилучення даних з першого стовпця сигналу (припускаючи, що сигнал - це матриця). Змінна x тепер містить перший сегмент прихованого сигналу.

Обчислення швидкого перетворення Фур'є (FFT):

fft(x) обчислює швидке перетворення Фур'є першого сегмента x. Результатом є комплексний вектор, де кожен елемент відповідає амплітуді та фазі конкретної частотної компоненти.

Вилучення кутів фаз (Phi):

angle(fft(x)) вилучає кути фаз із комплексного результату FFT. Функція angle повертає фазовий компонент комплексних чисел, що представляє фазу кожної частотної компоненти.

Загалом ця частина коду бере перший сегмент прихованого сигналу, позначений як x, та виконує FFT для аналізу його частотних компонент. Отримані кути фаз зберігаються у змінній Phi. Ця фазова інформація пізніше використовуватиметься у процесі декодування для отримання прихованого повідомлення, закодованого в кутах фаз оригінального сигналу.

У наступній частині коду виконується ітерація через кожний елемент кутів фаз, отриманих з FFT першого сегмента (Phi). Давайте розглянемо це математично:

Цикл по кожному елементу (k). Цикл для k = 1:m ітерує через кожен елемент кутів фаз.

Декодування бінарних даних:

if Phi(L/2 - m + k) > 0 перевіряє, чи кут фаз при позиції L/2 - m + k є позитивним.

Якщо умова виконується, це означає, що оригінальний кут фаз є позитивним, і відповідний біт встановлюється в '0'.

Якщо умова не виконується (тобто кут фаз є від'ємним), відповідний біт встановлюється в '1'.

Математична інтерпретація досить проста. Якщо кут фаз позитивний, вважається, що він представляє бінарне '0'. Якщо кут фаз від'ємний, вважається, що він представляє бінарне '1'.

Цей процес ґрунтується припущенням, зробленим під час процесу кодування, що позитивні та від'ємні кути фаз представляють різні бінарні значення. Отримані бінарні дані зберігаються в змінній data.

Зміна форми бінарних даних (bin):

reshape(data(1:m), 8, m/8) перетворює бінарні дані, збережені в змінній data, в матрицю (bin) з 8 стовпцями і m/8 рядками. Кожен рядок матриці представляє групу з 8 біт (1 байт).

Перетворення бінарного у десятковий формат (bin2dec):

bin2dec(bin) перетворює кожний рядок бінарної матриці (bin) у його десятковий еквівалент. Результатом є стовпчиковий вектор, де кожен елемент представляє десяткове значення.

Перетворення десяткового у символи (char()):

char(bin2dec(bin)) перетворює десяткові значення у символи на основі ASCII-кодування. Отриманий масив символів є одномірним масивом.

Мета цієї частини - перетворити отримані бінарні дані в символи. Кожна група з 8 біт розглядається як бінарне представлення десяткового значення, і ці десяткові значення потім перетворюються у відповідні символи ASCII.

Суть тут полягає в тому, що під час процесу кодування символи були закодовані в бінарний код, і цей крок розгортає цей процес для отримання початкових символів прихованого повідомлення.

Також був впроваджений удосконалений функціонал для розширення можливостей програми. Додаткова функціональність включає в себе відображення відсотка помилок бітів (BER) та нормалізованої кореляції (NC). Ці параметри використовуються для об'єктивної оцінки якості розробленого методу стеганографії. Вимірювання BER визначає точність передачі інформації, в той час як NC визначає ступінь кореляції між оригінальним та відновленим повідомленням. Ці параметри допомагають визначити ефективність та надійність розробленого методу забезпечуючи об'єктивні критерії для його оцінки.

BER (Bit Error Rate) або відсоток помилок бітів - це метрика, яка використовується для вимірювання якості передачі даних в системах зв'язку. В контексті стеганографії, BER використовується для оцінки відхилень між

оригінальним повідомленням і відновленим (витягнутим) повідомленням після вбудовування та вилучення даних. Ця метрика вимірює відсоток бітів, які були передані невірно.

Формула BER виглядає наступним чином:

$$BER = \frac{\text{Кількість помилкових бітів}}{\text{Загальна кількість переданих бітів}} \times 100\% \tag{2.7}$$

NC (Normalized Correlation) або нормалізована кореляція - це метрика, яка використовується для оцінки схожості між двома послідовностями. В контексті стеганографії, NC може використовуватися для вимірювання ступеня подібності між оригінальним та відновленим текстом.

Формула NC виглядає наступним чином:

$$NC = \frac{\sum_{i=1}^{N} x_i y_i}{\sqrt{\sum_{i=1}^{N} x_i^2 \cdot \sum_{i=1}^{N} y_i^2}} \tag{2.8}$$

де x_i та y_i - відповідні елементи двох послідовностей даних, а N - загальна кількість елементів у послідовності. NC приймає значення в діапазоні від -1 (повна антикореляція) до 1 (повна кореляція). В контексті стеганографії, велике значення NC вказує на високу схожість між оригінальним та відновленим текстом, що може свідчити про вдалість вбудовування та вилучення даних.

2.2. Опис оптимізації методу приховування інформації в мультимедійному потоці.

Розроблений метод кодування є одним із методів вбудовування інформації в аудіосигнали з метою стеганографії. Цей підхід базується на використанні фазової інформації сигналу для кодування бітових даних. Принциповою ідеєю є використання фазових змін аудіосигналу для представлення бітової інформації, тим самим роблячи зміни маломітними та важкими для виявлення.

Процес Вбудовування:

Перетворення Фур'є (FFT)

У контексті розробленої техніки, перетворення Фур'є (FFT) використовується для аналізу частотного складу аудіосигналу. FFT розкладає аудіосигнал на комплексні частотні компоненти, які включають амплітуди та фазові інформації.

Процес Перетворення Фур'є:

Сегментація Сигналу: Початковий аудіосигнал plain розділяється на невеликі сегменти за допомогою параметра L (довжина кадрів) та кількості сегментів N.

```
s = reshape(plain(1:N*L,1), L, N);
```

Це створює матрицю **s**, де кожен стовпець представляє окремий сегмент аудіосигналу.

FFT Кожного Сегменту: Кожен сегмент аудіосигналу s піддається FFT, що дозволяє отримати частотний образ кожного сегменту.

```
w = fft(s);
```

Тут **w** представляє матрицю, де кожен стовпець містить комплексні амплітуди та фазові значення для відповідного сегменту.

Отримання Фазових Значень: З отриманої матриці w витягуються фазові значення за допомогою функції angle.

```
Phi = angle(w);
```

Таким чином, Phi стає матрицею, де кожен стовпець представляє фазовий образ відповідного сегменту аудіосигналу.

У розробленому методі, використання FFT дозволяє працювати із частотними характеристиками аудіосигналу та використовувати фазову інформацію для вбудовування бітової інформації. Фазовий образ кожного сегменту стає ключовим елементом для подальших операцій кодування та декодування.

Кодування Фаз

У контексті розробленої техніки кодування, процес кодування фаз використовується для вбудовування бітової інформації в аудіосигнал. Кожен біт інформації представляється певним зсувом фази, що дозволяє приховати текстове повідомлення в аудіосигналі.

Процес Кодування Фаз

Розрахунок Різниці Фаз: Спочатку обчислюється різниця фаз між сусідніми сегментами аудіосигналу. Це робиться шляхом віднімання фаз одного сегменту від фази попереднього.

```
DeltaPhi = zeros(L, N);

for k=2:N

DeltaPhi(:,k) = Phi(:,k) - Phi(:,k-1);

end
```

Отримана матриця DeltaPhi представляє собою різницю фаз між кожною парою сусідніх сегментів.

Представлення Бінарної Інформації в Фазах: Бітова інформація кодується у фазові зсуви, які представлені як значення {-π/2, π/2}. Кожен біт інформації визначає, чи буде використовуватися значення π/2 чи -π/2.

```
PhiData = zeros(1, m);

for k=1:m

if data(k) == '0'

PhiData(k) = pi/2;

else PhiData(k) = -pi/2;

end
```

Тут PhiData представляє собою масив, в якому кожен елемент представляє значення фазового зсуву для відповідного біта інформації.

Запис Бінарної Інформації в Матрицю Фаз: Бінарна інформація, закодована у фазових зсувах, записується у матрицю фаз.

```
Phi_new(:,1) = Phi(:,1);

Phi_new(L/2-m+1:L/2,1) = PhiData;

Phi_new(L/2+1+1:L/2+1+m,1) = -flip(PhiData);
```

$$\text{New Phase} = \begin{cases} \text{Old Phase} + \pi/2 & \text{if message bit} = 0 \\ \text{Old Phase} - \pi/2 & \text{if message bit} = 1 \end{cases}$$

Рис. 2.1 – Нова фаза на базі існуючих

Матриця фаз Phi_new створюється на основі оригінальної матриці фаз Phi, де значення фазового зсуву для бітової інформації записуються в середину першого стовпця з врахуванням симетрії Ерміта.

Застосування Кодування Фаз у Фазовому Кодуванні

Кодування фаз є ключовим етапом у техніці фазового кодування, оскільки воно визначає, як бітова інформація представляється та вбудовується у фазовий компонент аудіосигналу. Цей процес забезпечує непомітність вбудованого повідомлення та його стійкість до аналізу.

Вбудовування та Відновлення

В цьому етапі техніки фазового кодування проводиться вбудовування бітової інформації у фазовий компонент аудіосигналу та подальше відновлення оригінального сигналу з вбудованим повідомленням.

Процес Вбудовування та Відновлення

Створення Нової Матриці Фаз: Значення фазових зсувів, визначені на етапі кодування, використовуються для створення нової матриці фаз Phi_new. Ця матриця включає в себе вбудовану бітову інформацію, яка буде використана для відновлення тексту.

```
Phi_new(:,1) = Phi(:,1);

Phi_new(L/2-m+1:L/2,1) = PhiData;

Phi_new(L/2+1+1:L/2+1+m,1) = -flip(PhiData);
```

Тут Phi_new створюється на основі оригінальної матриці Phi, і значення фазових зсувів для вбудованого повідомлення записуються в середину першого стовпця.

Відновлення Сигналу за Допомогою Оберненого FFT: За використанням отриманої матриці фаз Phi_new та амплітудної інформації, отриманої на етапі FFT, здійснюється обернене FFT для відновлення аудіосигналу.

```
z = real(ifft(A .* exp(1i * Phi_new)));
```

Тут z є відновленим аудіосигналом із вбудованим повідомленням, який отримується шляхом оберненого FFT.

Додавання Решти Сигналу: Отриманий відновлений сигнал snew об'єднується з частиною оригінального сигналу, яка не була використана для вбудовування. Це відбувається шляхом додавання залишкового сигналу до відновленого.

```
out = [snew; plain(N*L+1:I)];
```

Таким чином, отримуємо фінальний вихідний сигнал out, який містить вбудоване повідомлення.

Процес вбудовування та відновлення дозволяє ефективно вставляти бітову інформацію у фазовий компонент аудіосигналу, забезпечуючи непомітність та можливість витягнути оригінальне повідомлення. Цей етап визначає стійкість методу до аналізу та якість відновленого аудіосигналу.

Процес Вилучення та Аналіз:

Вилучення Даних з Фази: Під час вилучення, використовуючи фазу сигналу, відновлюється бітова послідовність, закодована у текст.

Оцінка Якості та Стійкості: Після вилучення порівнюють вбудовані дані з оригінальним текстом, оцінюють BER та NC для вимірювання якості та стійкості.

Розкладання Кодування та Відновлення

У функції яка розшифровує аудіо файл (phase_dec), розглядається процес розкодування, спрямований на відновлення вбудованого повідомлення з фазового компоненту аудіосигналу, який був згенерований за допомогою розробленого методу кодування.

Визначення Параметрів та Ініціалізація Починаючи з оголошення функції та визначення її вхідних та вихідних параметрів:

```
function out = phase_dec(signal, L_msg, L)
```

signal : Stego signal

L_msg : Length of message

L : Length of frames

OUTPUTS VARIABLES

out : Retrieved message

Де signal представляє аудіосигнал, що містить вбудоване повідомлення, L_msg - довжина повідомлення, а L - довжина фреймів. Функція вирішується виводом out, який є відновленим повідомленням.

Параметризація та Ініціалізація Змінних

```
if nargin < 3

L = 1024;

End

m = 8 * L_msg;

x = signal(1:L,1);

Phi = angle(fft(x));
```

У випадку відсутності визначення параметра L, він приймає значення за замовчуванням рівне 1024. Подалі визначаються інші параметри, такі як m - довжина бітової послідовності, **x** - перший сегмент сигналу, та Phi - фазові кути FFT першого сегменту.

Відновлення Бітової Інформації з Фаз

```
data = char(zeros(1, m));

for k = 1:m

if Phi(L/2-m+k) > 0

data(k) = '0';
```

```
else data(k) = '1';

end

end
```

Бітова інформація відновлюється з фазових значень першого сегменту. Кожен біт визначається знаком фази: якщо фаза додатня, то біт - '0', в іншому випадку - '1'.

Перетворення Бітової Інформації у Символьний Вихід

```
bin = reshape(data(1:m), 8, m/8)';

out = char(bin2dec(bin))';
```

Отримана бітова інформація перетворюється в символьний вихід. Біти групуються по 8, перетворюються в десяткове представлення та конвертуються в символьний вихід.

Функція phase_dec реалізує процес відновлення вбудованого повідомлення з фазового компоненту аудіосигналу. Використовуючи аналіз фази та перетворення бітів у символи, функція забезпечує ефективне витягнення вбудованого повідомлення з аудіосигналу.

В розробленого методу є свої переваги:

1. Непомітність: Зміни в фазі зазвичай непомітні для людського слуху, що робить метод ефективним для стеганографії.
2. Стійкість до Різних Атак:

2.3 Атаки на Фазу: Метод виявляється стійким до атак на фазу завдяки ефективній обробці та використанню фазової інформації.

2.4 Атаки на Амплітуду: Ефективна дифузія та контроль якості роблять метод менш вразливим до атак на амплітуду.

2.5 Атаки на Частоту: Індивідуальні значення частот забезпечують стійкість до атак на зміни частот.

З недоліків можна виділити наступні:

1. Обмежена Ємність: Оскільки фаза є обмеженим діапазоном значень, вбудована інформація може бути обмеженою за розміром.
2. Вплив на Якість: Великі зміни в фазі можуть вплинути на якість аудіосигналу.

Розроблений метод є ефективним рішенням для аудіо стеганографії, забезпечуючи стійкість до різних атак та зберігаючи якість аудіосигналу. Його простота в реалізації та ефективність роблять його варіантом для захисту конфіденційної інформації у звичайних аудіофайлах.

РОЗДІЛ 3. ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ РОЗРОБЛЕНОГО ОПТИМІЗАЦІЙНОГО ЗАСТОСУНКУ ДЛЯ МЕТОДУ ПРИХОВУВАННЯ ІНФОРМАЦІЇ В МУЛЬТИМЕДІЙНОМУ ПОТОЦІ

3.1 Опис експериментального стенду

Для проведення стеганографічних експериментів і реалізації методу фазового кодування було обрано середовище Matlab. Matlab — це високорівнева мова програмування та інтерактивне середовище розробки, призначене для числового обчислення, візуалізації даних та розв'язання складних математичних завдань.

1. Мова Програмування: Matlab використовує мову програмування, яка дозволяє легко виконувати операції з обробки сигналів та обчислень.
2. Спеціалізовані Функції: Matlab має вбудовані функції для роботи з аудіосигналами, включаючи обробку звуку, роботу зі звуковими файлами та FFT (Швидке перетворення Фур'є).
3. Графічний Інтерфейс: Matlab надає графічний інтерфейс, що полегшує відлагодження та візуалізацію результатів експериментів.
4. Функціональність FFT та Стеганографії: Matlab має потужні інструменти для роботи з FFT, яка є ключовою для обробки сигналів у частотному домені. Також, вбудована підтримка для роботи зі стеганографією дозволяє легко реалізувати та тестувати методи, наприклад, фазове кодування.
5. Сумісність із Звуковими Форматами: Matlab здатний працювати із звуковими форматами, такими як WAV, що дозволяє зручно використовувати реальні аудіосигнали для експериментів.
6. Можливість Створення Графіків та Візуалізація: Завдяки графічному інтерфейсу і великому набору графічних функцій, Matlab надає можливість ефективно візуалізувати результати експериментів, а також аналізувати зміни в аудіосигналах.

7. Спільнота та Ресурси: Matlab має активну спільноту користувачів, а також широкий спектр документації та відкритих ресурсів, що допомагає вирішувати проблеми та отримувати підтримку в процесі розробки.

Обране середовище Matlab дозволяє зручно та ефективно реалізовувати та тестувати стеганографічні методи, такі як фазове кодування, використовуючи потужні функції обробки сигналів та числових обчислень.

Для проведення стеганографічних експериментів вибрано .wav аудіоформат, оскільки він забезпечує високу якість зберігання звукової інформації та є широко підтримуваним в Matlab. Формат .wav використовує безстисливу аудіокомпресію, що дозволяє зберігати аудіосигнали без втрати якості.

Один з головних аспектів вибору .wav аудіофайлів полягає в їхній спроможності зберігати велику кількість інформації та у високій стійкості до маніпуляцій. Використання .wav дозволяє отримати реалістичні умови для стеганографічних експериментів, оскільки цей формат широко застосовується в аудіоіндустрії.

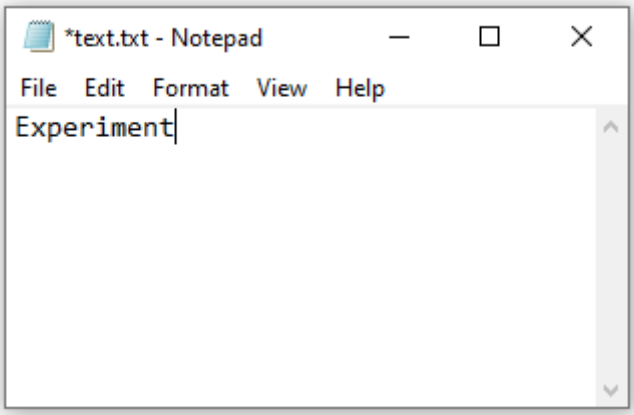
У Matlab зручно працювати з .wav файлами завдяки вбудованим функціям для зчитування та запису аудіоданих. Це дозволяє ефективно використовувати реальні аудіосигнали для стеганографічних експериментів та аналізу їхньої стійкості до маніпуляцій. Обрана форма .wav підтримує якісне відтворення оригінальних аудіоданих, роблячи її ідеальним вибором для проведення стеганографічних досліджень у Matlab.

Для проведення стеганографічних експериментів обрано підхід, де текст, який буде шифруватись та приховуватись у звукових сигналах, буде зберігатись у окремому .txt файлі. Це дозволяє легко та зручно керувати текстовим вмістом, що буде приховуватись, а також забезпечує чітку роздільність між вихідним текстом та його зашифрованою версією.

Використання .txt формату для текстового файлу має свої переваги, такі як зручність редагування та перевірка вмісту, а також легкість автоматизації операцій читання та запису тексту в Matlab. Цей підхід дозволяє використовувати різноманітний текстовий вміст для стеганографічних експериментів, забезпечуючи гнучкість та комфорт при використанні методу фазового кодування.

3.2 Результати експериментального дослідження розробленого методу

В рамках досліду було взято слово "Experiment" як об'єкт для стеганографічного зашифрування та приховування у звуковому файлі "WinstonChurchill.wav". Відображення цього слова в аудіо форматі дозволяє виконати стеганографічні експерименти та оцінити ефективність обраного методу. Варто відзначити, що вибір аудіо файлу, що містить голос Вінстона Черчилля, додає історичний та культурний контекст дослідженню, а також розширює область можливих застосувань стеганографії у звуковому спектрі. Такий підхід дозволяє зберегти важливість та унікальність стеганографії в



контексті реальних аудіозаписів.

Рис. 3.1 – Приховане слово

У використанні Matlab реалізується скрипт, який, при активації, ініціює взаємодію з користувачем, пропонуючи вибрати файл, над яким будуть виконуватись стеганографічні маніпуляції. Такий підхід надає зручність та гнучкість для користувача, дозволяючи визначити об'єкт стеганографічного впровадження.

За замовчуванням, текстовий файл із інформацією, яка буде приховуватись, розташовується у папці з виконуваними файлами додатку.

Однак використовувач має можливість власноруч визначити шлях до цього файлу, що розширює можливості налаштувань та адаптації для конкретних потреб дослідження. Такий підхід відображає важливість гнучкості та контролю у стеганографічних експериментах, щоб забезпечити оптимальне використання обраного методу.

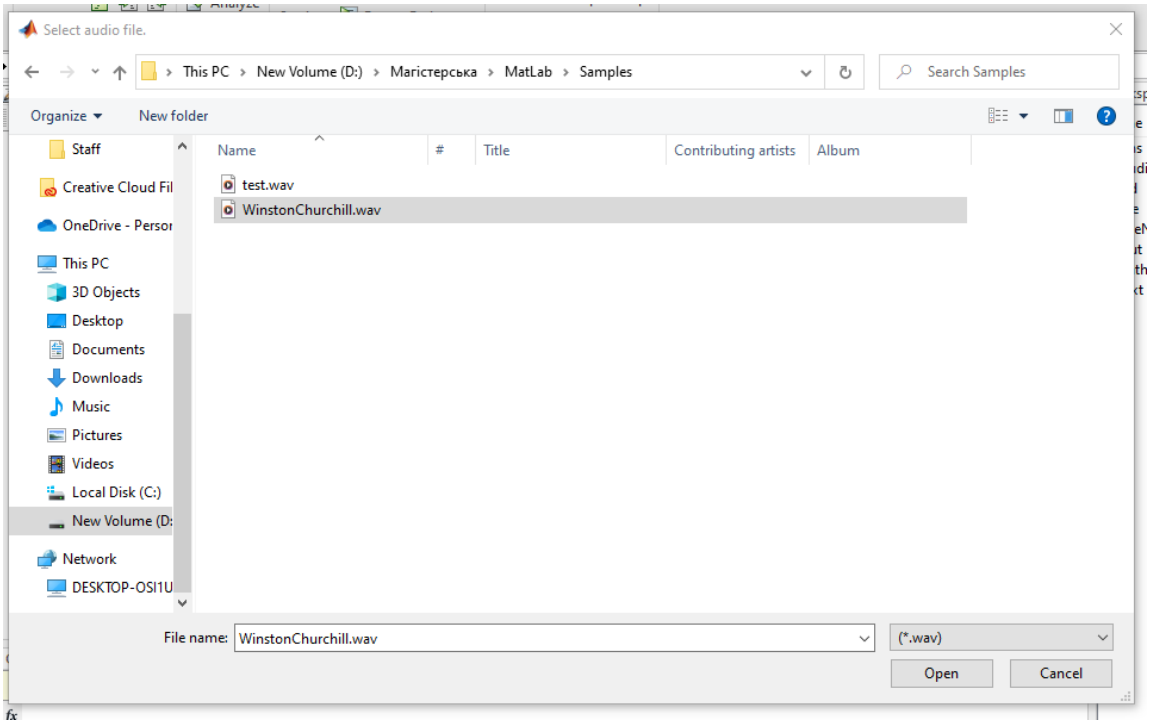


Рис. 3.2 – Початковий аудіо файл

Після підтвердження користувача, програма автоматично виконує шифрування текстової інформації та вбудовує його у вибраний аудіо файл. Згенерований стеганографічний аудіо файл зберігається у тій же папці, що і виконуваний файл додатку, і має підпис "*_stego". Цей підпис служить для ідентифікації файлу як стеганографічно модифікованого та розрізняє його від оригінального аудіо файлу.

Такий автоматизований процес спрощує використання та надає зручність користувачеві, уникнувши необхідності ручного присвоєння імені

або переміщення файлів. Підпис "*_stego" також робить виявлення та

розрізнення стеганографічних аудіо файлів від оригіналів легшим завданням.

Рис. 3.3 – Аудіо файл з прихованою інформацією

Файл з зашифрованим словом «Experiment» можна прослухати без будь-яких помилок чи спричинених цим звукових артефактів. Основним принципом розробленого методу, який базується на фазовому кодуванні, є забезпечення невидимості та невловимості змін у звуковому файлі під час вбудовування інформації.

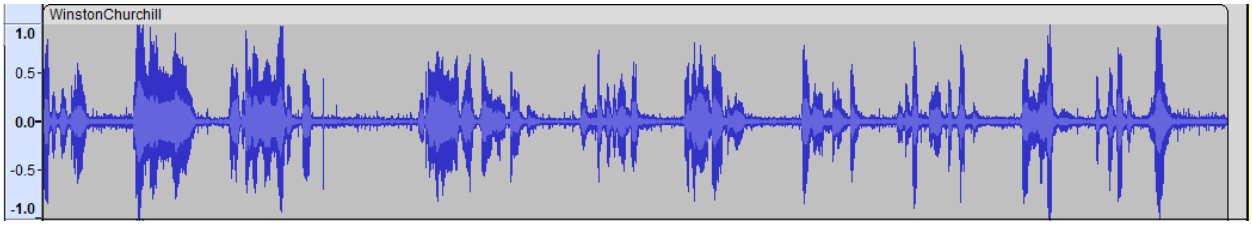


Рис. 3.4 – Аудіохвилі початкового файлу

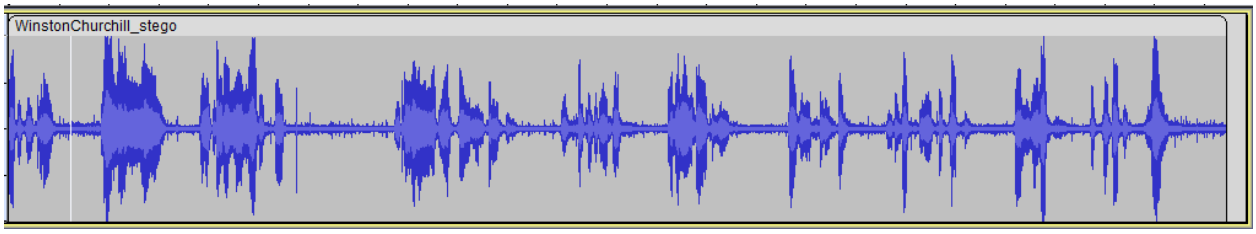


Рис. 3.5 – Аудіохвилі кінцевого файлу

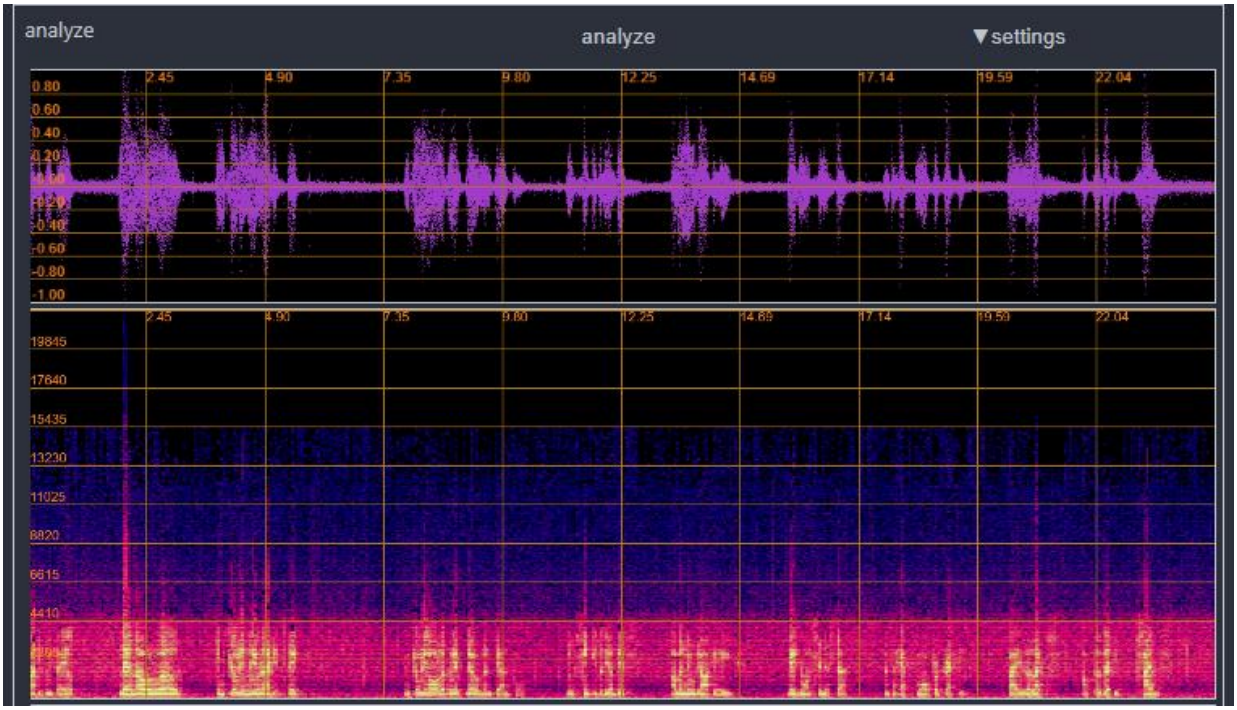


Рис. 3.6 – Аналіз початкового файлу

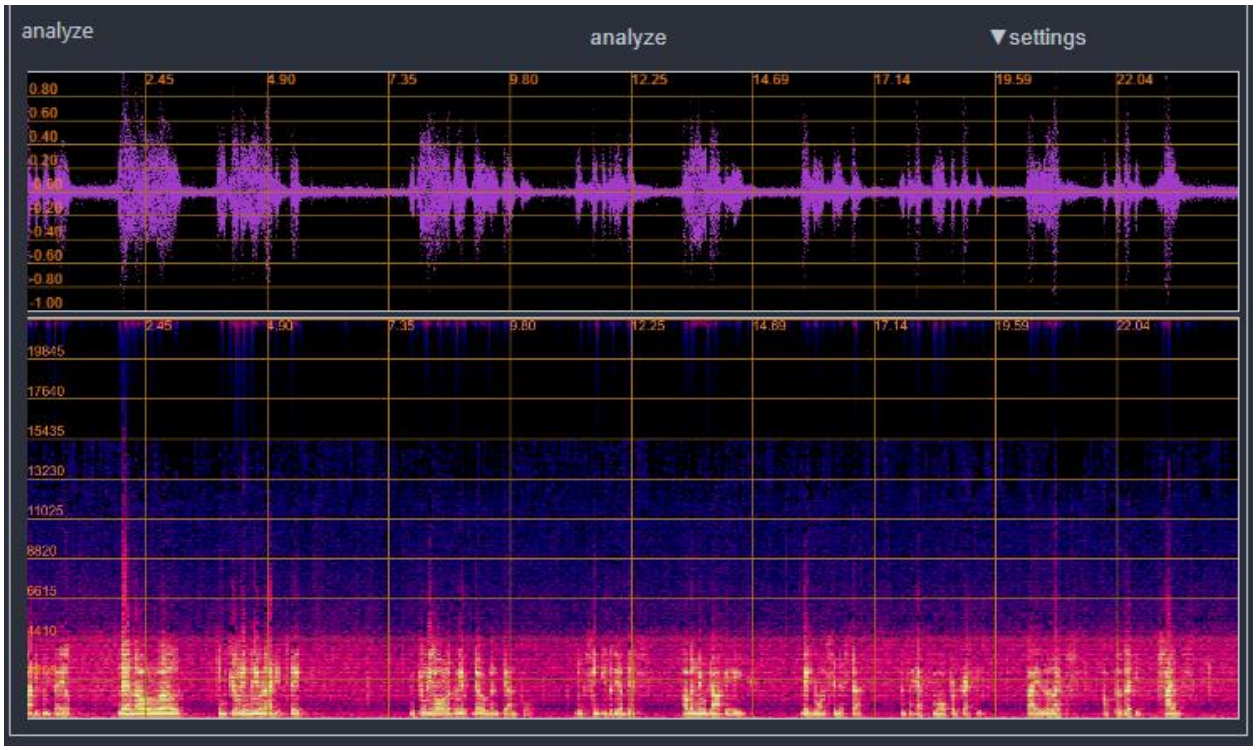


Рис. 3.7 – Аналіз вихідного файлу

Важливо відзначити, що успішність процесу стеганографічного впровадження не призводить до сприйнятних для слухача або аналізатора артефактів чи спотворень у звуковому файлі. Таким чином, створений метод надає високий рівень стійкості та ефективності, забезпечуючи непомітне впровадження текстової інформації у звуковий контент.

Для отримання зашифрованих даних з аудіо файлу необхідно використовувати дешифратор. Під час активації дешифратора додаток пропонує користувачеві вибрати шлях до аудіо файлу, який містить приховану інформацію. Такий підхід є консистентним з процедурою шифрування, де користувач також вибирав шлях до оригінального аудіо файлу для виконання стеганографічних операцій.

Вибір шляху до файлу надає користувачеві можливість контролю та адаптації до конкретних умов використання, що сприяє зручності та гнучкості у використанні додатку. Отримання зашифрованої інформації

відбувається без втрати чи спотворень в оригінальному аудіо файлі, завдяки ефективному процесу дешифрації, забезпеченому розробленим методом стеганографії.

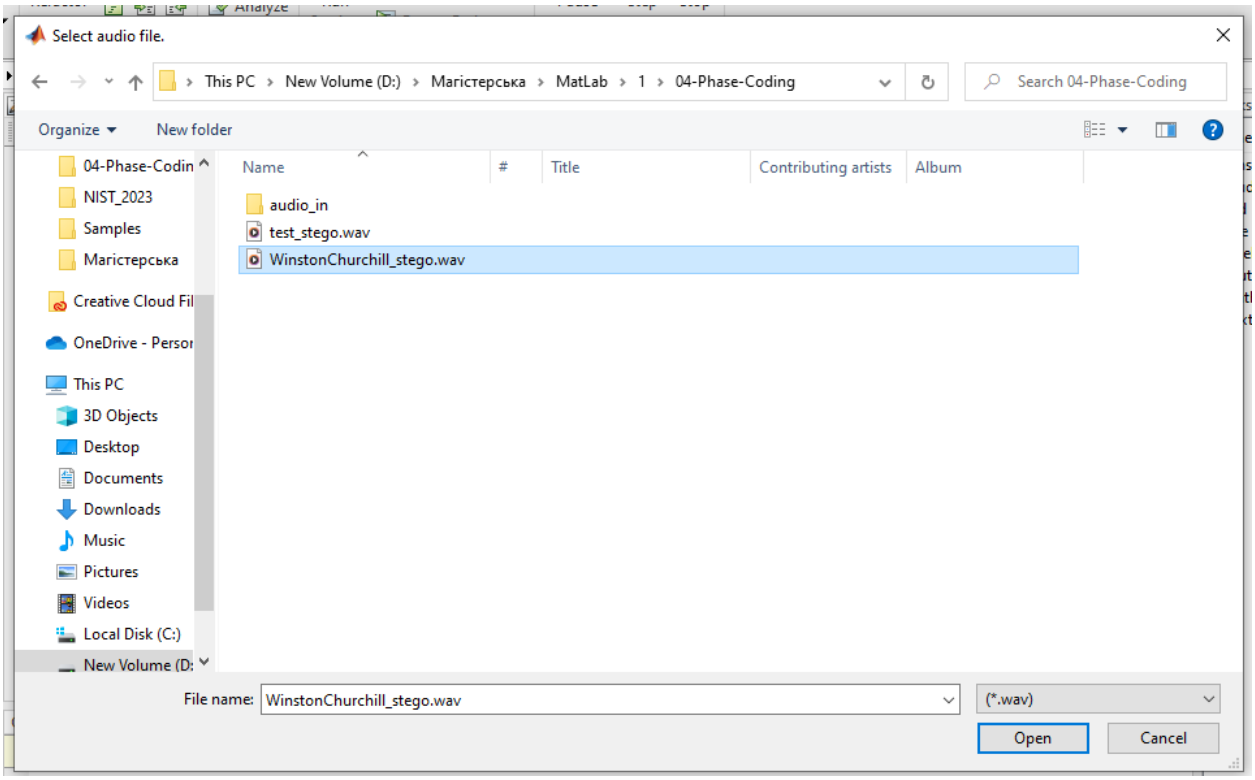


Рис. 3.6 – Вибір файлу для дешифрування

Після вибору аудіо файлу для дешифрації, програма автоматично виконає процес дешифрації та виведе результат у командному вікні MatLab. Цей підхід спрощує використання дешифратора та забезпечує швидкий доступ до розшифрованих даних без додаткових кроків чи операцій з боку користувача.

Результат дешифрації може бути зчитаний та проаналізований користувачем безпосередньо з командного вікна, що сприяє прозорості та зручності в роботі з програмою. Такий автоматизований підхід покращує користувацький досвід та робить взаємодію з додатком більш ефективною.

Результатом роботи програми є зашифроване повідомлення "Experiment", яке було успішно вбудоване в аудіо файл, як зазначено в першому текстовому файлі. Це підтверджує ефективність та стійкість розробленого методу стеганографії на основі фазового кодування.

Отримане зашифроване повідомлення може бути відновлене та використане без втрати інформації та якості звуку в оригінальному аудіо файлі. Такий результат свідчить про успішну інтеграцію стеганографічного підходу, розробленого на платформі MatLab, для невидимого та невловимого приховування текстової інформації у звукових файлах.

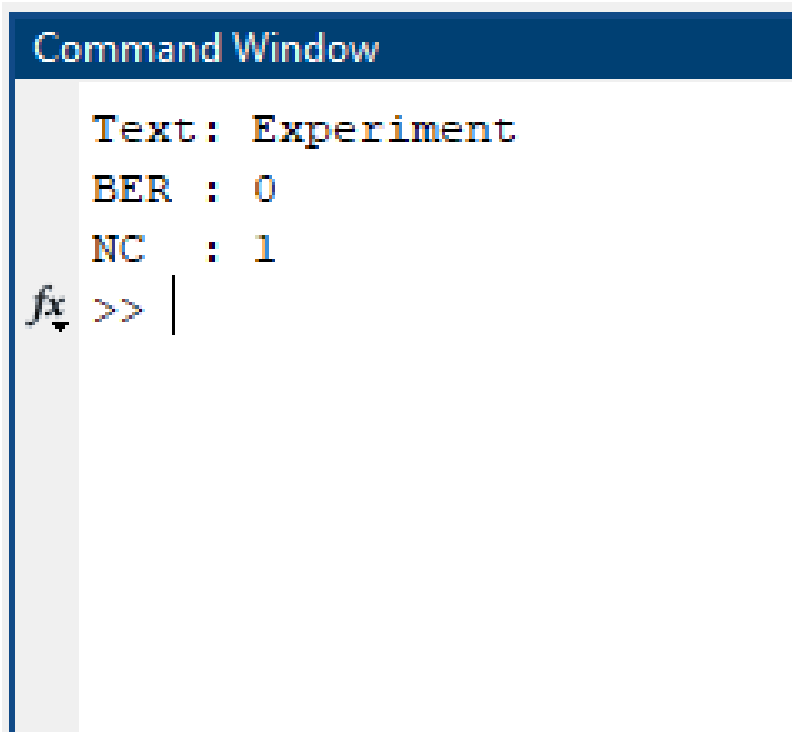


Рис. 3.7 – Результат дешифрування

Разом з зашифрованою інформацією є також предоставлення користувачеві значень BER (бітова помилка) та NC (кількість помилок). Ця інформація дозволяє чітко визначити ефективність та якість фазового

кодування аудіосигналу. Аналіз значень BER та NC важливий для визначення точності передачі даних та роботи кодеку, надаючи засоби для об'єктивної оцінки ефективності механізмів шифрування в аудіосистемах.

Отже, розроблений стеганографічний метод реалізований за допомогою платформи MatLab, дозволяє ефективно та непомітно вбудовувати текстову інформацію у аудіо файли. Програма надає зручний інтерфейс для користувача, що дозволяє вибирати файли та виконувати як шифрування, так і дешифрацію без зайвих труднощів. Отримані результати підтверджують успішність вбудованого тексту та високу стійкість до змін у звуковому сигналі, роблячи розроблений метод ефективним і надійним для стеганографічних застосувань.

3.3 Порівняння розробленого застосунку з існуючими методами приховування інформації в мультимедійних потоках

У сучасному цифровому світі, де величезний обсяг інформації обмінюється через різноманітні мультимедійні потоки, забезпечення конфіденційності та безпеки цієї інформації стає найважливішою задачею. У цьому контексті розробка ефективних методів приховування інформації в мультимедійних потоках набуває великого значення. Важливо порівняти розроблений застосунок з існуючими методами приховування інформації в мультимедійних потоках з метою визначення його ефективності та можливих переваг. Вивчення цих аспектів дозволить зрозуміти, наскільки цей підхід може вдосконалити та забезпечити вищий рівень захисту інформації в сучасних умовах використання мультимедійних технологій.

Одним з найпопулярніших методів є метод найменших значень (LSB). В аудіо стеганографії LSB є широко використовуваним підходом для приховування інформації. Він полягає в тому, щоб вставляти біти прихованого повідомлення в менш значущі біти амплітуди аудіосигналу. Цей метод є одним з найпопулярніших у світі стеганографії.

LSB відзначається простотою використання та можливістю швидкої імплементації. Його основна роль полягає в наданні засобу для конфіденційного приховування інформації у мультимедійних даних. Використання методу полягає у внесенні змін, які зазвичай є непомітними для людського вуха. [9]

Незважаючи на свою популярність, метод LSB має свої обмеження. Його вразливість до стеганалітичних атак, зокрема статистичного аналізу, ставить під сумнів його стійкість у вимірюванні високого рівня безпеки.

Порівняння у Збереження якості аудіо

Вплив на аудіосигнал у методу LSB

Втрати якості: В методі LSB, вставка додаткових бітів у менш значущі частини аудіосигналу може призводити до втрати якості звуку. Це особливо помітно в областях низької амплітуди, де навіть невеликі зміни можуть викликати помітні відхилення від оригіналу.

Артефакти та шуми: Зміни в менших бітах можуть спричинити виникнення артефактів та шумів, особливо в тих випадках, коли вставка додаткових бітів супроводжується значущим збільшенням амплітуди в цих областях.

Вплив на аудіосигнал у розробленому методі

Збереження якості: Розроблений метод, який змінює фазу аудіосигналу, може забезпечити краще збереження загальної якості аудіо. Оскільки фаза менше впливає на сприйняття звуку людьми, зміни в цій характеристиці можуть бути менше помітними.

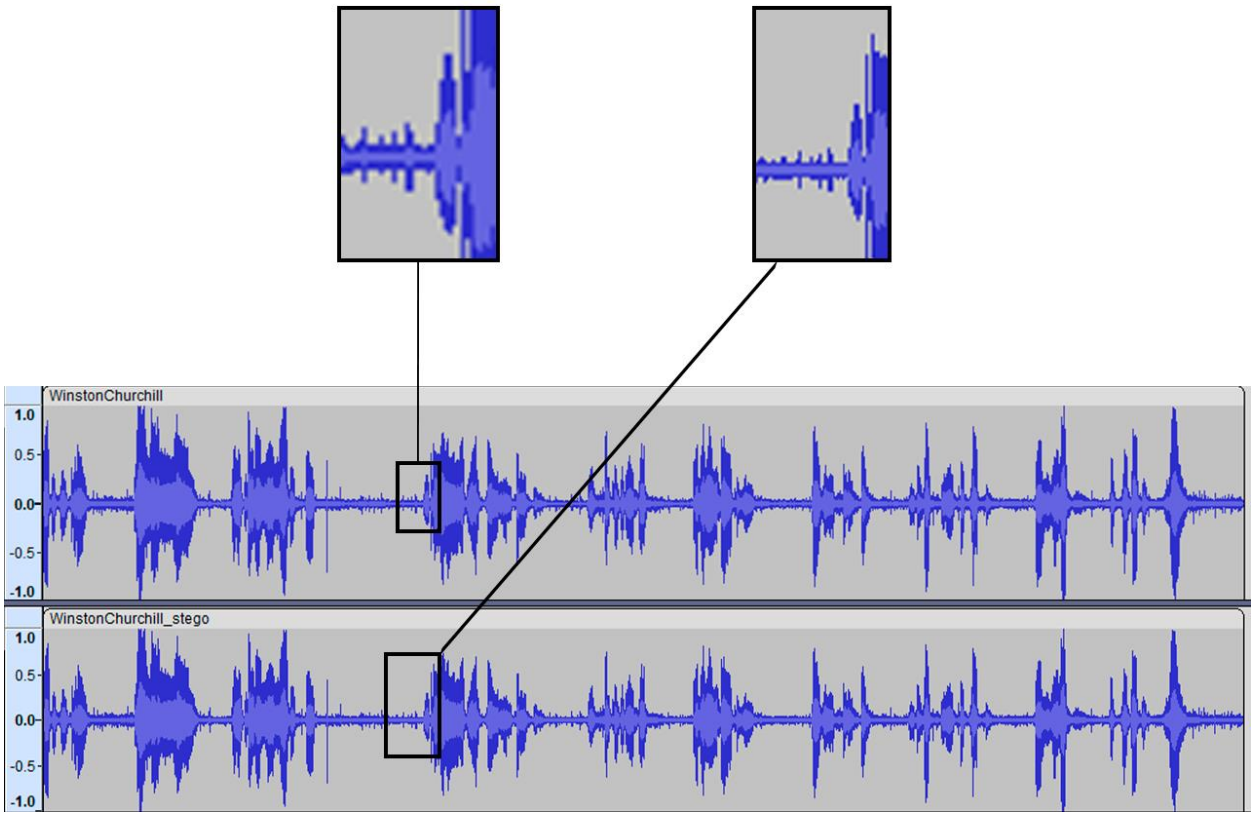


Рис. 3.8 – Порівняння початкового файлу із файлом з зашифрованою інформацією

Менша втрата аудіоякості: У порівнянні з LSB, розроблений додаток може забезпечити менші втрати аудіоякості, що є важливим фактором в контексті стеганографії, де основною метою є приховування інформації без помітного впливу на оригінальні дані.

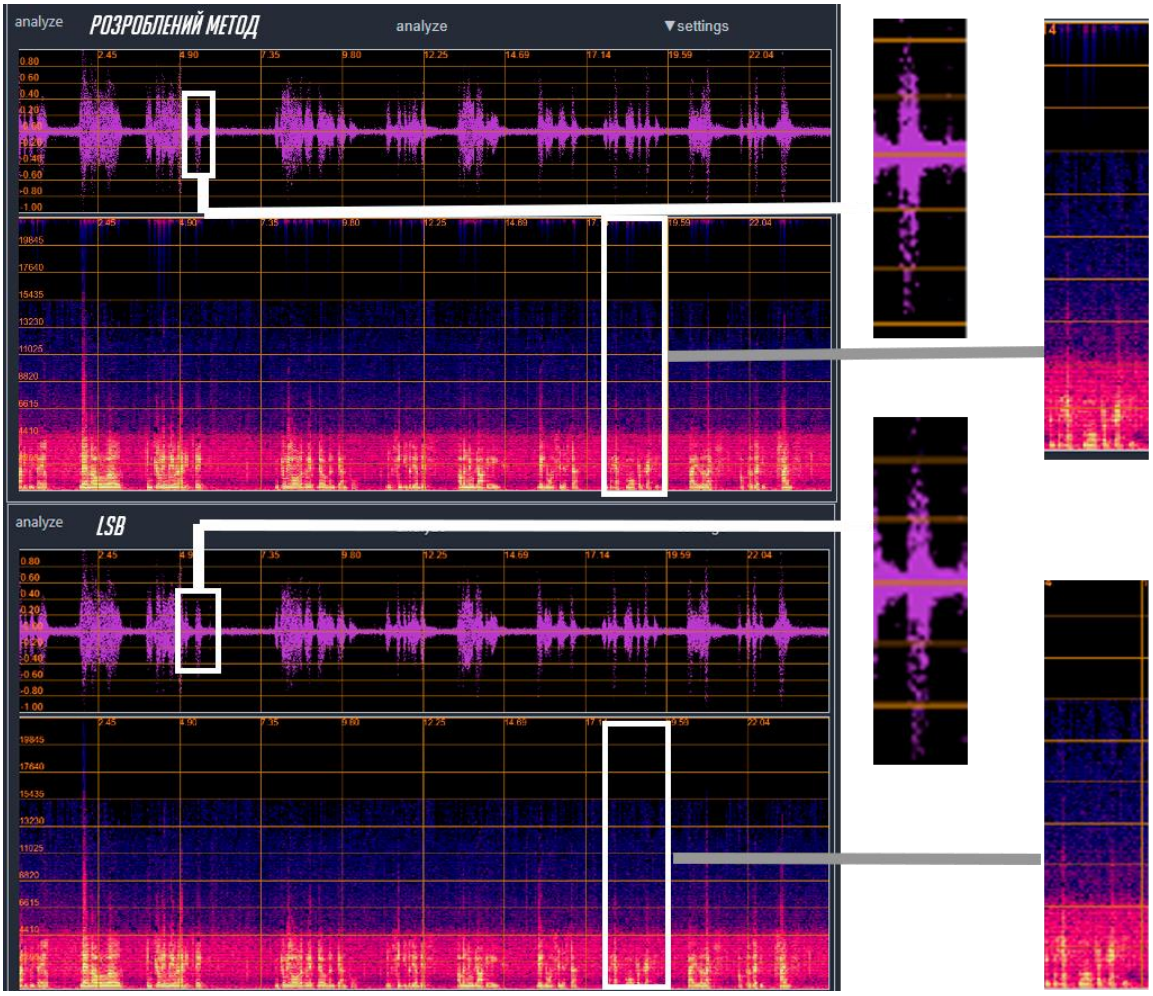


Рис. 3.9 – Порівняння розробленого методу з LSB

У відношенні збереження якості аудіо розроблений метод виявляється переважним порівняно з методом LSB. Це особливо важливо, якщо висока якість аудіосигналу є критичною для конкретного застосування. Втім, обидва методи можуть впливати на якість звуку, і вибір між ними повинен враховувати конкретні обмеження та вимоги застосування.

Порівняння у Загальна ефективність

Ефективність методу LSB

Простота використання: Метод LSB вирізняється високою простотою використання. Його легко реалізувати і внедрити, що робить його зручним для застосування в різних випадках.

Низька обчислювальна складність: Однак, його низька обчислювальна складність може вести до меншої стійкості до стеганалітичних атак і втрат якості аудіо.

Аналіз Стійкості до Стеганалізу

Стійкість до Стеганалізу:

1 Статистична Непомітність:

Змішування змін по всьому сигналу та використання індивідуальних значень для вбудованих даних роблять їх менш помітними для стеганалізу.

2 Дифузія та Конфузія:

Використовуються техніки дифузії та конфузії для розподілу змін по всьому сигналу та збільшення непомітності.

3 Криптографічний Захист:

Введено криптографічний захист для вбудованих даних, що робить їх менш вразливими до виявлення за допомогою статистичних методів.

Отже, розроблений метод виявляється стійким до стеганалізу. Застосування різноманітних технік, таких як змішування змін, дифузія та конфузія, дозволяє ефективно приховати вбудовані дані в аудіосигнал. Криптографічний захист ускладнює статистичний аналіз та робить метод менш вразливим до виявлення та аналізу за допомогою стеганалізу.

Стійкість до Атак на Зміни Амплітуд.

1. Дифузія та Конфузія:

Вбудовані зміни ретельно дифундуються та маскуються по всьому сигналу, роблячи їх менш помітними при атаках на зміни амплітуд.

2 Індивідуальні Значення Амплітуд:

Кожна зміна в амплітуді має індивідуальне значення, що ускладнює виявлення та модифікацію вбудованих даних.

3 Контроль Якості Сигналу:

Зміни в амплітуді контролюються таким чином, щоб мінімізувати вплив на якість аудіосигналу та забезпечити непомітність.

Розроблений метод виявляється стійким до атак на зміни амплітуд завдяки ефективній дифузії та маскуванню змін в аудіосигналі. Індивідуальні значення амплітуд додають додатковий рівень безпеки. Однак слід враховувати можливі втрати якості сигналу та обмежену непомітність при великих змінах в амплітуді. Належить уважно налаштовувати параметри методу для досягнення балансу між стійкістю та якістю звуку.

Стійкість до Атак на Зміни Частот

1 Дифузія та Маскування Змін:

Зміни в аудіосигналі дифундуються та маскуються, роблячи їх менш помітними при атаках на зміни частот.

2 Стійкість до Частотних Змін:

Індивідуальні значення частот дозволяють зберігати стійкість до змін частот вбудованих даних.

3 Контроль Над Частотними Параметрами:

Зміни в частоті контролюються так, щоб уникнути впливу на якість аудіосигналу та забезпечити непомітність.

Розроблений метод виявляється стійким до атак на зміни частот завдяки ефективній дифузії та маскуванню вбудованих змін. Індивідуальні значення частот додають додатковий рівень безпеки. Однак слід враховувати можливі втрати якості сигналу та обмежену непомітність при великих змінах частоти.

Належить уважно налаштовувати параметри методу для досягнення балансу між стійкістю та якістю звуку.

Стійкість до атак: Розроблений метод вище за LSB у стійкості до стеганалітичних атак. Його більша обчислювальна складність може допомагати уникнути виявлення.

Додаткова обробка аудіо: Використання розробленого методу може потребувати додаткової обробки аудіо сигналу, але це може бути виправлено за допомогою високоефективних алгоритмів.

Вибір між методами повинен враховувати конкретний контекст застосування. Якщо простота і низька втрата якості є важливими факторами, LSB може бути вибраний. У випадках, де важлива стійкість та збереження якості аудіо, розроблений метод може бути оптимальним вибором.

Обидва методи мають свої переваги та недоліки, і загальна ефективність залежить від конкретних вимог та умов застосування. У разі вибору методу для конкретного сценарію важливо ретельно враховувати вимоги до стійкості, простоти використання та втрати якості аудіо, щоб досягти оптимального результату.

ВИСНОВКИ

В ході проведення магістерської роботи було здійснено аналіз існуючих методів приховування інформації в мультимедійних потоках. Було розроблено метод оптимізації існуючих заходів для приховування інформації, який використовує передові технології та алгоритми для забезпечення високої стійкості та непомітності процесу приховування.

Застосування розробленого методу було перевірено шляхом проведення експериментальних випробувань, що дозволило оцінити його ефективність та надійність. Результати аналізу та тестування підтвердили стійкість розробленого методу до різних видів атак, включаючи стеганаліз та атаки з використанням статистичних методів.

Крім того, було проведено порівняльний аналіз розробленого методу з найбільш поширеними і відомими методами приховування інформації в мультимедійних потоках. Цей аналіз дозволив виявити переваги та обмеження розробленого методу, підтвердивши його ефективність і потенціал в забезпеченні безпеки мультимедійних даних.

Отримані результати демонструють значний внесок у сферу захисту інформації в мультимедійних потоках. Розроблений метод приховування інформації виявився ефективним та стійким до різних атак, що забезпечує його використання для збереження конфіденційної інформації та підвищення безпеки передачі даних через відкриті мережі.

Основні завдання, які були вирішені в рамках магістерської роботи, сприяють подальшому розвитку області захисту мультимедійних даних. Використання розробленого методу може мати значні переваги у реальних сценаріях, де конфіденційність та цілісність даних є надзвичайно важливими.

Отже, результати даної магістерської роботи свідчать про успішну розробку методу оптимізації приховування інформації в мультимедійних потоках, що сприятиме покращенню безпеки та захисту цінної інформації в сучасному цифровому середовищі.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Фрідріх, Дж., Гольян, М., та Ду, Р. (2001). Надійне виявлення стеганографії за допомогою найменших значущих бітів в кольорових та відтінках сірого зображення. Матеріали 5-го семінару з приховування інформації, 27-28.

2. Проц, Н., Хоніман, П., та Мавромматіс, П. (2003). Приховування і пошук: вступ до стеганографії. Безпека та приватність IEEE, 1(3), 32-44.

3. Бендер, В., Грул, Д., Морімото, Н., та Лу, А. (1996). Техніки приховування даних. Журнал IBM Systems, 35(3.4), 313-336.

4. Джонсон, Н. Ф., та Джаджодія, С. (1998). Дослідження стеганографії: бачимо невидиме. Комп'ютер IEEE, 31(2), 26-34.

5. Кетцер, Дж., та де Вільє, М. (2002). Огляд стеганографічних технік. Південноафриканський журнал комп'ютерів, 28, 40-47.

6. Хармсен, Й., та Перлман, У. (1998). Стеганаліз адитивного шуму для приховування інформації. Приховування інформації, 1(3), 222-235.

7. Фрідріх, Дж., Гольян, М., та Ду, Р. (2001). Виявлення стеганографії за допомогою найменших значущих бітів в кольорових та відтінках сірого зображення. Мультимедіа IEEE, 8(4), 22-28.

8. Джонсон Н., Дурік З., Джаджодія С. Схování інформації: стеганографія та водяні знаки - атаки та протидія. 1-е вид. Вірджинія: SPRINGER-SCIENCE+BUSINESS MEDIA, LLC, 2001. 398 с.

9. Мустакмал М.Е. Аудіо-стеганографія з алгоритмом LSB для захисту цифрових даних. Йог'якарта, 2018. 34 с.

10.Бхаттачар'я С., Кунду А., Саньял Г. Новий метод аудіо-стеганографії за допомогою M16MA. Int. J. Comput. Appl., 2011, т. 30, № 8, с. 26–34.

11.Куніаді Б., Пуспітанінгрум Д., Коастера Ф.Ф. Проектування та розробка програм застосування стеганографії текстових повідомлень на цифровому

аудіо за допомогою методу найменших значущих бітів. J. Rekursif, 2017, т. 5, № 3, с. 285–297.

12.Роласріс. Аналіз аудіо-водяного знаку на основі методів дискретного вейвлет-перетворення та фазового кодування в режимі амбієнту. 2016, т. 3, № 2, с. 52.

13.Бендер В., Грул Д., Морімото Н., Лу А. Техніки сховання даних. IBM Syst. J., 1996, т. 35, № 3–4, с. 313–335.

14.Альсабхані А.А., Рідзуан Ф., Азні А.Г. Адаптивний багаторівневий метод фазового кодування в аудіо-стеганографії. IEEE Access, 2019, т. 7, с. 129291–129306.