УДК 004.85 DOI https://doi.org/10.32782/1995-0519.2023.3.22

СТЕГАНОГРАФІЧНІ ПІДХОДИ ДО ОБРОБЛЕННЯ АУДІОСИГНАЛІВ

Євгеній Світловський
аспірант кафедри акустичних та мультимедійних електронних систем
Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», просп. Берестейський, 37, Київ, Україна, 03056, zsvetlovskiy336@gmail.com
ORCID: 0000-0002-8071-2221

Кирило Трапезон
кандидат технічних наук, доцент, доцент кафедри акустичних та мультимедійних електронних систем
Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», просп. Берестейський, 37, Київ, Україна, 03056, kirill.trapezon@gmail.com
ORCID: 0000-0001-5873-9519

Стеганографія в інформаційних системах використовується як один із дієвих технічних засобів для приховування інформації в мультимедійному контенті, а саме на етапах оброблення зображень, аудіо- та відеофайлів, а також під час передавання цього контенту з установленими елементами захисту від несанкціонованого доступу. Дослідження розкриває програмні особливості оброблення звукових сигналів на основі стеганографічного методу LSB. Алгоритм стеганографії може бути реалізований на етапі оцифрування записаних аналогових звукових сигналів, а також під час відновлення структури пошкоджених сигналів. Як приклад наведено основні етапи використання методу LSB для приховування короткого текстового словосполучення. Проаналізовано особливості у структурі аудіосигналу після застосування стеганографічного методу на основі отриманої спектрограми та амплітудно-частотної характеристики. Знайдено, що після застосування стеганографічного методу додане текстове повідомлення майже не вплинуло на частотні та спектральні характеристики аудіосигналу. Також під час прослуховування не було помічено спотворень.

Даний алгоритм має широкий спектр застосування і може бути використаний також і під час розроблення охоронних систем та систем моніторингу навколишнього оточення, визначення штучно змонтованих записів, реставрації та відновлення архівних аудіозаписів, що мають культурну цінність для певного проміжку розвитку суспільства. Знайдені обмеження слід ураховувати під час застосування стеганографічних методів до різнопланового аудіовізуального контенту.

Для аналізу звукового файлу до та після обробки, а також для розміщення необхідної інформації в оригінальний файл було застосовано інструменти мови програмування Python та можливості її бібліотек, таких як wave, librosa. Проаналізовано результати приховування текстового фрагменту в аудіосигналі та визначено основні кількісні відмінності. Метод стеганографії LSB є ефективним для передачі інформації з подальшим декодуванням на пристроях з обмеженою пропускною здатністю мережі або технічними обмеженнями.

Ключові слова: стеганографія, звук, сигнал, метод, алгоритм, частотна характеристика.

Актуальність роботи. Усе частіше стеганографія використовується як один із технічних засобів для приховування інформації під час роботи з мультимедійним контентом, особливо під час оброблення зображень, аудіо- та відеофайлів, а також передавання цього контенту з установленими елементами захисту від несанкціонованого доступу. Проте з розвитком технологій методи стеганографії стають усе більш уразливими до кібератак, до яких можна віднести перебір паролів, внесення змін у зображення або аудіофайли. Окрім того, потрібно враховувати і популярні сьогодні підходи машинного навчання для розпізнавання схем приховання інформації [1].

Зважаючи на сучасні досягнення, методи стеганографії виявили значний прогрес у приховуванні інформації з метою забезпечення конфіденційності. Однак вони все ще мають певні недоліки, які обмежують їх використання.

По-перше, це обмежена вмісткість, оскільки методи стеганографії здатні приховати лише невеликий обсяг інформації в контейнері. Прагнення збільшити обсяг прихованої інформації може призвести до помітних змін у контейнері, а це у цілому знижує ефективність стеганографії.

По-друге, ці методи можуть бути уразливими до різних атак, зокрема перехоплення даних може розкрити приховану інформацію, особливо якщо вона зберігається в незашифрованому вигляді.

Окрім того, недостатня стійкість являє собою ще одну проблему. Деякі методи стеганографії можуть бути легко зламані чи виявлені спеціалізованими алгоритмами або програмним забезпеченням, що може поставити під загрозу конфіденційність інформації [2]. Також важливою проблемою виступає вплив на якість контейнера, тобто оригінальний файл, у який було закодовано дані. Такі спотворення привертають увагу зловмисників, і це також створює загрозу злому та погіршення сприйняття основної інформації, такої як візуальні або акустичні характеристики.

Окрім того, деякі методи стеганографії вимагають певного типу контейнера або формату даних, що також може обмежити можливості приховування інформації в різних умовах. Також можна зазначити, що деякі методи застосовують значний ресурс обчислювального обладнання та обмежують застосування через тривалий час кодування, що може вплинути на продуктивність або призвести до затримок під час використання або внесення змін у контейнер [3].

Усі ці недоліки показують, що стеганографія хоча і є зручним та корисним інструментом для захисту конфіденційних даних, проте потребує обережного вибору методів та врахування контексту використання з метою забезпечення максимальної ефективності і безпеки.

Як приклад можна розглянути поширений метод стеганографії PixInWav. Як і в інших методах стеганографії, PixInWav піддається атакам на виявлення, коли криптоаналітики намагаються знайти приховані дані в носії. Внесення додаткових даних у зображення чи аудіофайл може призвести до зниження якості цього носія. За значних обсягів додаткової інформації може відбутися помітне спотворення або втрата деталей в оригінальному носії. Деякі методи стеганографії, включаючи PixInWav, можуть визначатись обмеженою пропускною здатністю для внесення додаткових даних, і це може ускладнювати передачу значних обсягів інформації [4].

Внесення додаткової інформації у зображення або аудіофайл може призводити до збільшення розміру останніх.

Отже, сьогодні можна відокремити основні чинники, які можуть бути притаманні різним методам стеганографії: низька стійкість до злому, швидкість кодування, спотворення оригінального файлу, обмеження в обсязі інформації [5]. Для вирішення або зменшення впливу зазначених чинників спочатку визначимо та розглянемо оптимальний спосіб кодування інформації, далі проведемо аналіз спотворення файлу, стійкість його до злому і, врешті-решт, запропонуємо один з алгоритмів для досягнення максимально швидкого кодування задля приховування потрібної інформації [6].

Метою статті є аналіз стеганографічного алгоритму додавання закодованої інформації з ознаками приховання до визначеного аудіофайлу на основі використання методу маски нижніх бітів. Для досягнення мети були сформульовані такі **завдання**:

– визначити основні етапи застосування методу LSB для процедури приховування текстового повідомлення у структурі звукового сигналу;

– знайти особливості, які виникають у структурі сигналу після застосування стеганографічного методу.

Матеріал та результати досліджень. Процес додавання текстового фрагменту в аудіофайл за допомогою одного з методів стеганографії LSB (Least Significant Bit, найменш значущий біт) полягає у механізмі додавання створених бітів із текстового фрагмента в менш значущі біти ауді-оданих. Тобто основна ідея методу LSB полягає у заміні найменш значущих бітів пікселів або семплів аудіо/відеоінформацією, яку ми хочемо приховати, без значних змін у візуальному або акустичному сприйнятті файлу [7].

Маска нижніх бітів – це значення, що використовується для вибору конкретних бітів у числових кодах. У контексті методу LSB-стеганографії маска нижніх бітів визначає, які саме біти будуть змінюватися для вбудовування прихованої інформації.

Для пояснення маски нижніх бітів припустимо, що ми маємо 8-бітний код для представлення значення пікселя зображення, семпла аудіо або іншого числового значення. Кожен біт у цьому коді може мати значення 0 або 1.

Маска нижніх бітів уявляється у вигляді бітової послідовності, де кожен біт визначає, які саме біти будуть впливати на вбудовування інформації. Зазвичай маска нижніх бітів має значення, у якому всі біти, крім найменш значущого, встановлені в 1, а найменш значущий біт – у 0.

Наприклад, якщо ми маємо маску нижніх бітів «11111110» для 8-бітного коду, це означає, що всі біти, крім найменш значущого, будуть змінюватися, тоді як найменш значущий біт залишиться без змін [8].

Застосовуючи маску, вбудовувані біти прихованої інформації можна замінити на відповідні біти менш значущих пікселів або семплів аудіо [9].

Основні кроки для кодування тексту в аудіо за допомогою методу LSB такі:

1. Перетворення тексту. Початковий текст, який потрібно приховати, спочатку перетворюється на бітовий формат. Кожен символ тексту може бути представлений у вигляді бітової послідовності, використовуючи, наприклад, кодування ASCII або Unicode.

2. Вибір аудіофайлу. Вибирається цільовий аудіофайл, у який буде вбудовуватися текстовий фрагмент. Це може бути будь-який формат аудіо, наприклад WAV, MP3 або FLAC.

3. Підготовка аудіофайлу. Аудіофайл може бути поділений на малі фрагменти або звуки, такі як семпли. Це залежить від точності, з якою ми хочемо вбудувати текстовий фрагмент.

4. Кодування. Найменш значущі біти аудіоданих (наприклад, амплітуди семплів) замінюються бітами тексту. Кожен біт тексту замінює лише один найменш значущий біт аудіоданих, щоб забезпечити мінімальні зміни у звучанні аудіофайлу. Наприклад, найменш значущий біт може бути замінений на біт тексту.

5. Збереження аудіофайлу. Після додавання текстового фрагмента біти аудіофайлу змінюються, і отриманий модифікований аудіофайл зберігається.

6. Вилучення тексту. Щоб прочитати прихований текст, необхідно провести процедуру зворотного вбудовування. Найменш значущі біти аудіоданих «витягуються» і перетворюються на текстовий формат, щоб отримати прихований текст.

Важливо враховувати, що вбудовування текстового фрагмента в аудіо може призвести до деякої втрати якості аудіофайлу або до зміни його звучання, тому необхідно збалансувати рівень прихованості та якості під час використання методу LSB для кодування тексту в аудіофайл.

Метод стеганографії звукового фрагменту за допомогою спеціальної бібліотеки wave від додатку Python полягає у вбудовуванні прихованої інформації у звуковий файл. Для цього використовується якраз метод LSB. Далі розглянемо алгоритм використання цього методу для приховування текстової інформації в аудіосигналі.

Спочатку імпортуємо аудіофайл без стиснення та візуалізуємо його спектральну характеристику до процедури обробки, тобто до застосування стеганографічного методу LSB (рис. 1).

Додатково наведемо графік залежності амплітуди від частоти (рис. 2).

Передусім необхідно завантажити бібліотеку wave та зчитати звуковий файл і далі за допомогою методу LSB закодувати повідомлення Secret Message в аудіофрагмент. Програмний код цієї процедури наведено на рис. 3.

Тобто повідомлення конвертується у бінарний формат, і далі відбувається ітерація по елементах аудіофайлу, де для кожного елемента в масці нижніх бітів установлюється відповідне бінарне значення повідомлення. Після цього зберігаємо закодований файл для подальшого аналізу.

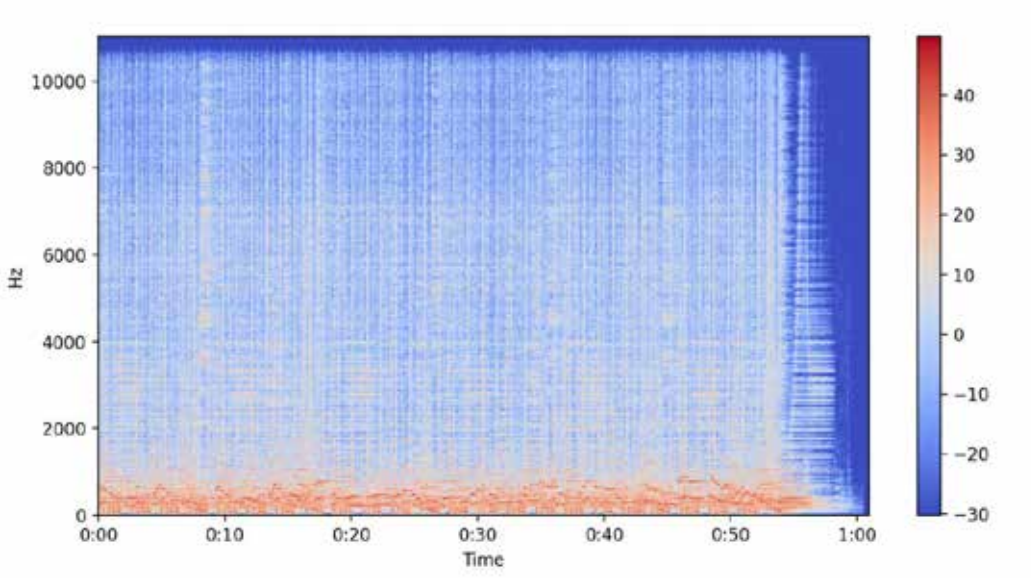

Рис. 1. Спектрограма сигналу до обробки

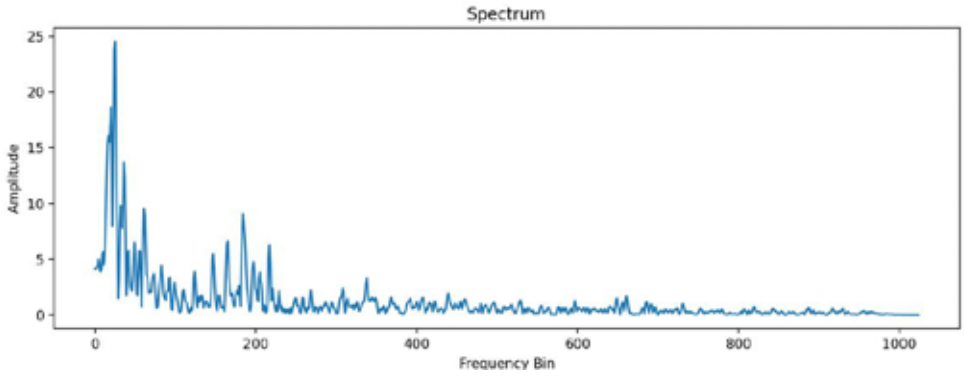

Рис. 2. Спектральна характеристика сигналу до обробки

```
import wave
# read wave audio file
song = wave.open("song.wav", mode='rb')
# Read frames and convert to byte array
frame_bytes = bytearray(list(song.readframes(song.getnframes())))

# The "secret" text message
string='Secret Message'
# Append dummy data to fill out rest of the bytes. Receiver shall detect and remove these characters.
string = string + int((len(frame_bytes)-(len(string)*8*8))/8) *'#'
# Convert text to bit array
bits = list(map(int, ''.join([bin(ord(i)).lstrip('0b').rjust(8,'0') for i in string])))

# Replace LSB of each byte of the audio data by one bit from the text bit array
for i, bit in enumerate(bits):
    frame_bytes[i] = (frame_bytes[i] & 254) | bit
# Get the modified bytes
frame_modified = bytes(frame_bytes)

# Write bytes to a new wave audio file
with wave.open('song_embedded.wav', 'wb') as fd:
    fd.setparams(song.getparams())
    fd.writeframes(frame_modified)
song.close()
```

Рис. 3. Кодування текстового повідомлення

Після застосування стеганографічного методу LSB проаналізуємо закодоване аудіо за допомогою частотної характеристики та спектрограми (рис. 4, 5).

Аналізуючи отримані результати після застосування стеганографічного методу, можна відзначити, що закодоване повідомлення майже не вплинуло на частотні та спектральні характеристики аудіосигналу. Також під час прослуховування не було помічено спотворень.

Наступним кроком у рамках дослідження розшифруємо повідомлення зі створеного аудіофрагменту. Для цього завантажимо закодований аудіофайл та виконаємо ітерацію по елементах аудіофайлу в межах заданого масиву, де для кожного елемента отримується бінарне значення за допомогою оператора.

Кожні вісім бінарних значень складаються у символ, який додається до повідомлення. Якщо досягнуто кінця повідомлення (символа '\0'), ітерація припиняється і повідомлення виводиться на екран (рис. 6).

У результаті в консолі отримуємо декодований текст: Successfully decoded: Secret Message. У рамках дослідження проаналізуємо основні параметри, що змінилися після обробки сигналу (табл. 1).

У табл. 1 наведено такі скорочення. RMS-амплітуда – означає нормування за середньоквадратичним значенням рівня звуку у файлі. Повна протилежність піковій нормалізації. За цього способу величина звуку вимірюється в децибелах [10]. Коефіцієнт гучності – відносне значення

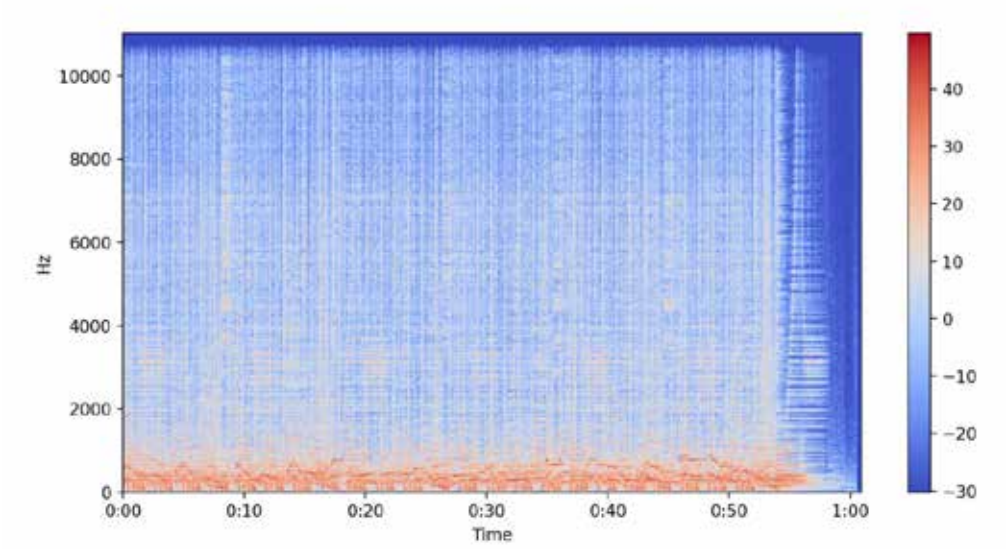

Рис. 4. Спектрограма сигналу після обробки

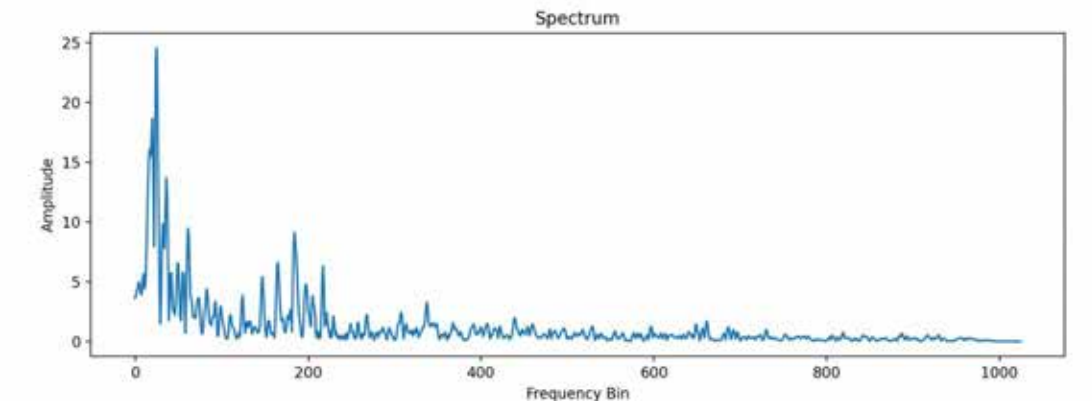

Рис. 5. Спектральна характеристика сигналу після обробки

```
73  import wave
74  song = wave.open("song_embedded.wav", mode='rb')
75  # Convert audio to byte array
76  frame_bytes = bytearray(list(song.readframes(song.getnframes())))
77
78  # Extract the LSB of each byte
79  extracted = [frame_bytes[i] & 1 for i in range(len(frame_bytes))]
80  # Convert byte array back to string
81  string = "".join(chr(int("".join(map(str,extracted[i:i+8])),2)) for i in range(0,len(extracted),8))
82  # Cut off at the filler characters
83  decoded = string.split("###")[0]
84
85  # Print the extracted text
86  print("Sucessfully decoded: "+decoded)
87  song.close()
```

Рис. 6. Програмний код

амплітуди, щоб пік абсолютної амплітуди становив Параметр RMS lev dB – значення RMS в дБ. Pk count – кількість разів, коли сигнал досяг мінімального або максимального рівня гучності.

Таблиця 1

Аналіз основних характеристик оригінального та закодованого сигналів

Параметр	**Оригінальний запис**	**З закодованим фрагментом**
Максимальна амплітуда	0.988500	0.995275
Мінімальна амплітуда	-0.986547	-0.994359
Середнє значення амплітуди	0.000977	0.000458
RMS-амплітуда	0.335910	0.335956
Коефіцієнт гучності	1.012	1.005
RMS lev dB	-9.48	-9.47
Pk count	2.5	2

Порівнявши результати аналізу звукового фрагменту до та після кодування прихованого повідомлення, можна зробити такі висновки:

– закодований фрагмент має вищий показник середньоквадратичного значення, що свідчить про додаткову інформацію в аудіо;

– файл із закодованим повідомленням має менше значення мінімальної амплітуди та більше значення максимальної, що свідчить про зміни в бінарному коді, а отже, впливає на відтворення звуку. Хоча варто відзначити, що різниця незначна і її неможливо помітити без аналізу файлу;

– коефіцієнт гучності виявився більшим в оригінальному записі, це свідчить про втрату корисної інформації за рахунок заміщення менш значущих бітів. Так само на це вказують показники RMS та кількість досягнутих пікових показників: через заміну звукової інформації закодованим повідомленням маємо меншу кількість звукової інформації.

Висновки. Знайдено, що в результаті додавання текстового повідомлення спектральна характеристика та спектрограма аудіосигналу практично не зазнали змін після кодування. Окрім того, під час прослуховування аудіо не було помічено жодних спотворень. Це можна пояснити тим, що обсяг закодованого текстового фрагмента значно менший порівняно з вихідним аудіосигналом. Найбільш вірогідно, що якщо б кодувалося відео або зображення, то різниця між оригінальним звуковим файлом та закодованим була б помітнішою.

Однак під час детального аналізу показників амплітуди та гучності були помічені незначні відмінності, такі як (табл. 1): вищий показник нелінійних спотворень, спричинений закодованою інформацією, різниця максимальної та мінімальної амплітуди на графіках амплітудно-частотної характеристики становила 0.006775 дБ для верхньої точки та 0.007812 дБ для нижньої. Водночас отримали і зменшення коефіцієнту гучності на 0.007, і менші показники RMS та кількість досягнутих піків, і це свідчить про втрату корисної інформації звукового фрагменту.

Метод стеганографії LSB є ефективним для передачі інформації з подальшим декодуванням на пристроях з обмеженою пропускною здатністю мережі або технічними обмеженнями. Із переваг можна відзначити варіативність та можливість розширення й удосконалення за допомогою застосування додаткових бібліотек як для аналізу, так і для кодування повідомлення з мінімальною втратою якості.

Проте варто врахувати, що процес кодування та використання відповідної бібліотеки є доступним для широкого кола користувачів, що може створювати потенційну загрозу щодо їх уразливості до потенційного злому. Тому хоча цей метод є ефективним для передачі додаткової інформації в обмежених умовах, проте він не підходить для передачі вкрай конфіденційної чи секретної інформації, оскільки існує ризик його несанкціонованого доступу.

ЛІТЕРАТУРА

1. Pevný, T., Filler, T., & Bas, P. Using high-dimensional image models to perform highly undetectable steganography. *Proceedings of the 11th ACM workshop on multimedia and security*, 15–26 (2010). URL: https://doi.org/10.1007/978-3-642-16435-4_13.

2. Provos, N., & Honeyman, P. Hide and seek: an introduction to steganography. *IEEE Security & Privacy,* 3, 1, 32–44 (2003). URL: https://doi.org/10.1109/msecp.2003.1203220.

3. ASLANTAŞ, F., HANİLÇİ, C. Comparative Analysis Of Audio Steganography Methods. *Journal of Innovative Science and Engineering (JISE),* 1–6 (2022). https://doi.org/10.38088/jise.932549.

4. Goljan, M., Fridrich, J., Cogranne, R. Rich model for Steganalysis of color images. *IEEE International Workshop on Information Forensics and Security (WIFS),* 1–6 (2014). URL: https://doi.org/10.1109/wifs.2014.7084325.

5. Pevný, T., Filler, T., & Bas, P. Using high-dimensional image models to perform highly undetectable steganography. *Proceedings of the 11th ACM workshop on multimedia and security*, 15–26 (2010). URL: https://doi.org/10.1007/978-3-642-16435-4_13.

6. Іванова О.М., Дрозд О.В., Защолкін К.В., Кузнєцов М.О. Підхід до нееквівалентного стеганографічного вбудовування додаткових даних у програмний код блоків lut FPGA. *Вісник Кременчуцького національного університету імені Михайла Остроградського.* 2021. Вип. 6/2021(131). С. 60–61. URL: https://doi.org/10.30929/1995-0519.2021.6.60-65.

7. Cachin, C. An information-theoretic model for steganography. Information Theory. *In Information and Computation*, 1, 192, 41–56. (2004). URL: https://doi.org/10.1016/j.ic.2004.02.003.

8. Yang, G., Zhang, H. Using Higher Order DCT Difference to Effective Improve Markov Process Based JPEG Steganalysis Detection Rate. *In Asia-Pacific Conference on Information Processing.* (2009). URL: https://doi.org/10.1109/apcip.2009.148.

9. Fridrich, J., Goljan, M., & Du, R. Reliable detection of LSB steganography in color and grayscale images. *in Proceedings of the 2001 workshop on Multimedia and security new challenges.* 145–152 (2001). URL: https://doi.org/10.1145/1232454.1232466.

10. Giuliani C., Gerosa D., Brugnara M.F. Improved automatic speech recognition through speaker normalization. *In Computer Speech & Language*, 1,20, 107–123 (2006). URL: https://doi.org/10.1016/j.csl.2005.05.002.

STEGANOGRAPHIC APPROACHES TO AUDIO SIGNAL PROCESSING

Yevhenii Svitlovskyi
Postgraduate Student at the Department of Acoustic and Multimedia Electronic Systems
National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”, 37 Beresteyskyi ave., Kyiv, Ukraine, 03056, zsvetlovskiy336@gmail.com
ORCID: 0000-0002-8071-2221

Kyrylo Trapezon
Candidate of Technical Sciences, Associate Professor, Associate Professor at the Department of Acoustic and Multimedia Electronic Systems
National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”, 37 Beresteyskyi ave., Kyiv, Ukraine, 03056, kirill.trapezon@gmail.com
ORCID: 0000-0001-5873-9519

Purpose. The aim of the article is to analyze a steganographic algorithm for adding encoded information with hiding features to a given audio file based on the use of the lower-bit mask method. **Methodology.** The methodology for adding hidden information to an audio signal is determined by software tools and algorithms of the Python object-oriented programming language libraries. Obtaining the signal spectrum is realized programmatically based on the fast Fourier transform. **Findings.** There is a method of encoding a text fragment into an audio file using the "wave" Python library. The original and decoded signal was analyzed, as well as program codes of nodes of this circuit. Frequency response, spectrogram and main parameters of the audio signal before and after coding were studied. Disadvantages, advantages, availability and possible areas of application of the presented method are indicated. **Originality.** The paper presents a simplified new algorithm for implementing the steganographic method, which allows adding textual information of various contents to an audio signal using only the tools of a programming language. A simple method for decoding audio data based on the approach of iterating over individual elements within a given array is defined. **Practical value.** The steganography embedding approach proposed in this paper allows efficient use of equipment resources and ensures high quality reproduction of both the encoded and the original fragment. It can be used in a large amount of audio reproduction systems with the need to transfer additional information through it. **Conclusions**. It was found that as a result of adding a text message, the spectral response and spectrogram of the audio signal remained virtually unchanged after encoding. In addition, no distortion was noticed when listening to the audio. The LSB steganography method is effective for transmitting information with subsequent decoding on devices with limited network bandwidth or technical limitations. Among the advantages are the variability and the possibility of expansion and improvement through the use of additional libraries, both for analyzing and encoding messages with minimal loss of quality.

Key words: steganography, sound, signal, method, algorithm, frequency characteristic.

REFERENCES

1. Pevný, T., Filler, T., & Bas, P. Using high-dimensional image models to perform highly undetectable steganography. *Proceedings of the 11th ACM workshop on multimedia and security*, 15–26 (2010). URL: https://doi.org/10.1007/978-3-642-16435-4_13

2. Provos, N., & Honeyman, P. Hide and seek: an introduction to steganography. *IEEE Security & Privacy,* 3, 1, 32–44 (2003). URL: https://doi.org/10.1109/msecp.2003.1203220

3. ASLANTAŞ, F., HANİLÇİ, C. Comparative Analysis Of Audio Steganography Methods. *Journal of Innovative*

Science and Engineering (JISE), 1–6 (2022). URL: https://doi.org/10.38088/jise.932549

4. Goljan, M., Fridrich, J., Cogranne, R. Rich model for Steganalysis of color images. *IEEE International Workshop on Information Forensics and Security (WIFS),* 1–6 (2014). URL: https://doi.org/10.1109/wifs.2014.7084325

5. Pevný, T., Filler, T., & Bas, P. Using high-dimensional image models to perform highly undetectable steganography. *Proceedings of the 11th ACM workshop on multimedia and security*, 15–26 (2010). URL: https://doi.org/10.1007/978-3-642-16435-4_13

6. O.M. Ivanova, O.V. Drozd, K.V. Zashcholkin, M.O. Kuznietsov. Pidkhid do neekvivalentnoho stehanohrafichnoho vbudovuvannia dodatkovykh danykh v prohramnyi kod blokiv lut FPGA. *Visnyk Kremenchutskoho natsionalnoho universytetu imeni Mykhaila Ostrohradskoh*, Vyp. 6/2021 (131), 60–61 (2021). [In Ukranian] URL: https://doi.org/10.30929/1995-0519.2021.6.60-65

7. Cachin, C. An information-theoretic model for steganography. Information Theory. *In Information and Computation*, 1,192, 41–56. (2004). URL: https://doi.org/10.1016/j.ic.2004.02.003

8. Yang, G., Zhang, H. Using Higher Order DCT Difference to Effective Improve Markov Process Based JPEG Steganalysis Detection Rate. *In Asia-Pacific Conference on Information Processing.* (2009). URL: https://doi.org/10.1109/apcip.2009.148

9. Fridrich, J., Goljan, M., & Du, R. Reliable detection of LSB steganography in color and grayscale images. *in Proceedings of the 2001 workshop on Multimedia and security new challenges.* 145–152 (2001). URL: https://doi.org/10.1145/1232454.1232466

10. C Giuliani, D., Gerosa, M., Brugnara, F. Improved automatic speech recognition through speaker normalization. *In Computer Speech & Language*, 1,20, 107–123 (2006). URL: https://doi.org/10.1016/j.csl.2005.05.002

Стаття надійшла 15.06.2023