



Я, як студент ХНУРЕ, розумію та підтримую політику закладу з академічної доброчесності. Я не надавав та не одержував недозволену допомогу під час підготовки кваліфікаційної роботи. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.

08.01.2025



Моргунов М.М.

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

Факультет \_\_\_\_\_ АКТ  
Кафедра \_\_\_\_\_ КІТАР  
Рівень вищої освіти \_\_\_\_\_ другий (магістерський)  
Спеціальність \_\_\_\_\_ 174 Автоматизація та комп'ютерно-інтегровані технології  
та робототехніка  
Тип програми \_\_\_\_\_ Освітньо-професійна  
Освітня програма \_\_\_\_\_ Комп'ютерно-інтегровані технологічні процеси і  
виробництва  
(шифр і назва)

ЗАТВЕРДЖУЮ:

Зав.кафедри \_\_\_\_\_  
(підпис)

«\_\_» \_\_\_\_\_ 2025р.

**ЗАВДАННЯ  
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

здобувачеві \_\_\_\_\_ Моргунову Максиму Максимовичу  
(шифр і назва)

1. Тема роботи: \_\_\_\_\_ Розроблення програмного модуля для передачі інформації  
про стан мобільного робота в зображенні

Затверджена наказом університету від \_\_\_\_\_ 22.11.2024 №1231Ст

2. Термін подання студентом роботи до екзаменаційної комісії \_\_\_\_\_ 12.01.2025р.

3. Вихідні дані до роботи: 3.1 Метод LSB; 3.2 Вдосконалення методу  
шифрування даних; 3.3 Передача технічної інформації в зображенні

4. Перелік питань, що потрібно опрацювати в роботі: 4.1 Вступ; 4.2 Аналіз  
методів передачі технічної інформації про стан мобільного робота;

4.3 Аналіз сучасних методів кодування та декодування технічної інформації;

4.4 Дослідження метода LSB з точки зору кодування інформацій про стан  
мобільного робота; 4.5 Математичний опис метода кодування та

розкодування технічної інформації; 4.6 Приклад кодування та розкодування  
технічної інформації; 4.7 Розробка загального алгоритму роботи програми;

4.8 Розробка функцій кодування та розкодування технічної інформації;

4.9 Експериментальні дослідження та аналіз отриманих результатів;

4.10 Висновки

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій Графічний демонстраційний матеріал в форматі PowerPoint(\*.ppt) формату А4 – 10-15 сторінок.

6. Консультанти розділів роботи

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

**КАЛЕНДАРНИЙ ПЛАН**

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Аналіз методів передачі технічної інформації про стан мобільного робота	11.09.2024-22.09.2024	виконано
2	Аналіз сучасних методів кодування технічної інформації	23.09.2024-01.10.2024	виконано
3	Аналіз сучасних методів декодування технічної інформації	02.10.2024-09.10.2024	виконано
4	Дослідження метода LSB з точки зору кодування інформацій про стан мобільного робота	10.10.2024-11.10.2024	виконано
5	Математичний опис метода кодування технічної інформації	25.10.2024-07.11.2024	виконано
6	Математичний опис метода розкодування технічної інформації	08.11.2024-13.11.2024	виконано
7	Приклад кодування та розкодування технічної інформації	14.11.2024-21.11.2024	виконано
8	Розробка загального алгоритму роботи програми	22.11.2024-03.12.2024	виконано
9	Розробка функцій кодування та розкодування технічної інформації	04.12.2024-16.12.2024	виконано
10	Експериментальні дослідження та аналіз отриманих результатів	17.12.2024-25.12.2024	виконано

Дата видачі завдання 2 вересня 2024р.

Здобувач \_\_\_\_\_



Моргунов М.М.

(прізвище, ініціали)

Керівник роботи \_\_\_\_\_

(підпис)

Євсєєв В.В.

(прізвище, ініціали)

## РЕФЕРАТ

Пояснювальна записка: 92 с., 4 табл., 15 рис., 4 дод., 22 джерела.

МОБІЛЬНИЙ РОБОТ, ПЕРЕДАЧА ДАНИХ, ДЕКОДУВАННЯ,  
ШИФРУВАННЯ, ЗОБРАЖЕННЯ, АЛГОРИТМИ, ТЕХНІЧНА  
ІНФОРМАЦІЯ.

Мета дослідження – підвищення ефективності передачі технічної інформації про стан мобільного робота, за рахунок вдосконалення методу шифрування даних в зображенні.

Об'єкт дослідження – процес передачі технічної інформації.

Предмет дослідження – методи, алгоритми та програмне забезпечення передачі інформації про стан мобільного робота.

В кваліфікаційній роботі проведено аналіз методів передачі технічної інформації про стан мобільного робота. Проаналізовано сучасні методи кодування та декодування технічної інформації. Досліджено метод LSB з точки зору кодування інформацій про стан мобільного робота. Математично описано методи кодування та розкодування технічної інформації. Наведено приклад кодування та розкодування технічної інформації. Вибрано середовище розробки, розроблено загальний алгоритм роботи програми та функції кодування та розкодування технічної інформації. Проведено експериментальні дослідження та аналіз отриманих результатів.

Отримані результати кваліфікаційної роботи можна віднести до Цілі сталого розвитку 9 «Промисловість, інновації та інфраструктура», а саме 9.4.

## ABSTRACT

Explanatory note: 92 pages, 4 tables, 15 figures, 4 app, 22 sources.

MOBILE ROBOT, DATA TRANSMISSION, DECODING, ENCRYPTION, IMAGES, ALGORITHMS, TECHNICAL INFORMATION.

The purpose of the study is to increase the efficiency of transmitting technical information about the state of a mobile robot by improving the method of data encryption in images.

The object of the study is the process of transmitting technical information.

The subject of the study is methods, algorithms and software for transmitting information about the state of a mobile robot.

The qualification work analyzes the methods of transmitting technical information about the state of a mobile robot. Modern methods of encoding and decoding technical information are analyzed. The LSB method is studied from the point of view of encoding information about the state of a mobile robot. Methods of encoding and decoding technical information are mathematically described. An example of encoding and decoding technical information is given. A development environment is selected, a general algorithm for the program and functions of encoding and decoding technical information are developed. Experimental studies and analysis of the results are carried out.

The results of the qualification work can be attributed to Sustainable Development Goal 9 "Industry, Innovation and Infrastructure", namely 9.4.

## ЗМІСТ

Перелік умовних скорочень .....	9
Вступ.....	10
1 Аналіз переваг та недоліків сучасних методів передачі технічної інформації про стан мобільного робота .....	12
1.1 Аналіз методів передачі технічної інформації про стан мобільного робота .....	12
1.2 Аналіз сучасних методів кодування технічної інформації.....	15
1.3 Аналіз сучасних методів декодування технічної інформації.....	17
1.4 Постановка задач дослідження .....	20
2 Розробка метода кодування технічної інформації про стан мобільного робота на базі метода LSB.....	21
2.1 Дослідження метода LSB з точки зору кодування інформацій про стан мобільного робота.....	21
2.2 Математичний опис метода кодування технічної інформації.....	25
2.3 Математичний опис метода розкодування технічної інформації.....	30
2.4 Приклад кодування та розкодування технічної інформації .....	33
2.5 Висновки до 2 розділу .....	41
3 Розробка програми кодування та розкодування технічної інформації на базі потокового відео.....	43
3.1 Вибір середовища розробки.....	43
3.2 Розробка загального алгоритму роботи програми.....	44
3.3 Розробка функцій кодування та розкодування технічної інформації ...	52
3.4 Висновки до 3 розділу .....	57
4 Експериментальні дослідження та аналіз отриманих результатів.....	58
4.1 Постановка мети та задач експерименту .....	58
4.2 Проведення експерименту та аналіз отриманих результатів .....	61

4.3 Охорона праці.....	69
4.4 Висновки до 4 розділу .....	70
Висновки .....	72
Перелік джерел посилання .....	73
Додаток А Програма кодування технічної інформації про стан мобільного робота в відео потоці .....	77
Додаток Б Програма декодування технічної інформації про стан мобільного робота .....	80
Додаток В Апробація результатів наукових досліджень.....	82
Додаток Г Демонстраційний матеріал .....	91

## ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

CRC – циклічні коди;

DCT – дискретне косинусне перетворення;

DWT – хвилькові перетворення;

IDE – інтегроване середовище розробки;

LDPC – low-density parity-check codes;

LSB – least significant bit.

## ВСТУП

З розвитком технологій шифрування та декодування, новітні алгоритми стають більш адаптивними, забезпечуючи компроміс між швидкістю, якістю та обсягом обчислень. Зокрема, використання методів, що дозволяють вбудовувати інформацію у зображення, створює додаткові можливості для підвищення стійкості даних до перехоплення або втрат.

Актуальність даного дослідження обумовлена зростаючою роллю мобільних роботів у сучасних технологічних системах, таких як автоматизоване виробництво, логістика, розвідка та обслуговування. Для ефективного функціонування цих систем необхідна надійна передача інформації про технічний стан роботів, що включає ключові параметри їхнього руху, працездатності сенсорів та енергоспоживання. Вдосконалення методів передачі та декодування даних дозволяє зменшити затримки, підвищити точність відновлення інформації та забезпечити захищеність комунікацій навіть у складних умовах роботи.

Мета дослідження – підвищення ефективності передачі технічної інформації про стан мобільного робота, за рахунок вдосконалення методу шифрування даних в зображенні.

Об'єкт дослідження – процес передачі технічної інформації.

Предмет дослідження – методи, алгоритми та програмне забезпечення передачі інформації про стан мобільного робота.

Для досягнення поставленої мети необхідно вирішити такі завдання:

- провести аналіз методів передачі технічної інформації про стан мобільного робота
- провести аналіз сучасних методів кодування технічної інформації
- провести аналіз сучасних методів декодування технічної інформації

- провести дослідження метода LSB з точки зору кодування інформації про стан мобільного робота
- провести математичний опис метода кодування технічної інформації
- провести математичний опис метода розкодування технічної інформації
- навести приклад кодування та розкодування технічної інформації
- провести вибір середовища розробки
- розробити загальний алгоритм роботи програми
- розробити функції кодування та розкодування технічної інформації
- провести експериментальні дослідження та аналіз отриманих результатів.

Кваліфікаційна робота виконана згідно ДСТУ 3008:2015 [1], керуючись навчальним посібником з дипломного проектування [2] та методичними вказівками [3]. Результати кваліфікаційної роботи отримали апробацію на наукових конференціях [4].

# 1 АНАЛІЗ ПЕРЕВАГ ТА НЕДОЛІКІВ СУЧАСНИХ МЕТОДІВ ПЕРЕДАЧІ ТЕХНІЧНОЇ ІНФОРМАЦІЇ ПРО СТАН МОБІЛЬНОГО РОБОТА

## 1.1 Аналіз методів передачі технічної інформації про стан мобільного робота

Передача технічної інформації про стан мобільного робота є критично важливою складовою для забезпечення надійного управління, моніторингу та діагностики його роботи. У сучасних системах використовуються різноманітні методи передачі даних, кожен із яких має свої особливості, переваги та недоліки. Одним із найбільш поширених методів є передача даних через бездротові мережі Wi-Fi, цей метод забезпечує високошвидкісний обмін інформацією на значних відстанях і дозволяє легко інтегрувати мобільного робота в локальну мережу або інтернет. Перевагами Wi-Fi є велика пропускна здатність, доступність і простота реалізації. Однак, основними недоліками є залежність від інфраструктури мережі та вразливість до перешкод і кібератак [5].

Ще один популярний метод – Bluetooth-з'єднання, яке широко використовується для передачі даних на короткі відстані. Перевагами Bluetooth є низьке енергоспоживання, висока швидкість з'єднання та стабільність у середовищі з низьким рівнем перешкод. Проте, основними обмеженнями є невеликий радіус дії (до 100 метрів) і менша швидкість передачі в порівнянні з Wi-Fi. Мобільні мережі (3G, 4G, 5G) дозволяють передавати інформацію про стан робота в реальному часі на будь-які відстані, що робить їх гарними для застосувань, де потрібна постійна комунікація. Наприклад, 5G забезпечує мінімальні затримки та високу швидкість передачі даних, недоліками ж таких мереж є залежність від

покриття мобільного оператора, висока вартість послуг і значне енергоспоживання.

Такий метод, як радіозв'язок на основі спеціалізованих передавачів і приймачів часто використовується у військових і промислових системах. Цей метод є надійним і може функціонувати в умовах, де інші технології не працюють через перешкоди. Водночас він потребує складного налаштування апаратного забезпечення і не забезпечує такої швидкості передачі, як Wi-Fi чи мобільні мережі. Метод передачі даних через ультразвукові сигнали є менш поширеним, але має унікальні переваги, зокрема стійкість до електромагнітних завад та може бути використаний у специфічних середовищах, таких як підводні роботи. Недоліками даного методу є невелика дальність і низька швидкість передачі даних. Інфрачервоний зв'язок застосовується для передачі інформації на короткі відстані, не потребує ліцензування частотного спектра і є енергоефективним, проте, цей метод вимагає прямої видимості між передавачем і приймачем, що обмежує його використання [6].

У сучасних мобільних роботах активно впроваджуються технології Zigbee та Z-Wave, які є оптимальними для систем із низьким енергоспоживанням. Вони дозволяють створювати мережі із багатьма вузлами для передачі даних між пристроями.

Zigbee – це бездротовий протокол, розроблений для низькоенергетичних мереж передачі даних. Його основне призначення – створення систем інтернету речей, а також взаємодія пристроїв із низьким енергоспоживанням. Zigbee працює у частотних діапазонах 2,4 ГГц, 868 МГц (Європа) і 915 МГц (США), забезпечуючи універсальність використання. Zigbee підтримує різні топології мереж – зіркову, деревоподібну і сітчасту (mesh). Остання дозволяє пристроям передавати дані через інші вузли, збільшуючи покриття мережі та її стійкість до збоїв. Технологія розроблена для пристроїв із тривалим терміном автономної роботи (до кількох років від

одного елемента живлення). Швидкість передачі даних варіюється від 20 до 250 кбіт/с, що підходить для передачі невеликих обсягів інформації, таких як сенсорні дані або команди управління. Zigbee забезпечує передачу на відстань до 100 метрів у відкритому просторі, а в сітчастих мережах ця відстань може бути збільшена за рахунок ретрансляції. До недоліків цієї технології можна віднести невисоку пропускну здатність, що обмежує використання у задачах, які вимагають швидкої передачі великих обсягів даних та залежність від чіткого налаштування для забезпечення стабільності роботи мережі [7].

Z-Wave – це ще один бездротовий протокол для побудови IoT-систем, який схожий за функціоналом із Zigbee, але має свої відмінності. Z-Wave працює в ліцензованих частотних діапазонах, що знижує ризик перешкод від інших пристроїв, які використовують популярний діапазон 2,4 ГГц. Як і Zigbee, Z-Wave підтримує сітчасту топологію, що дозволяє будувати розширені та стійкі до збоїв мережі. Z-Wave використовує єдину чітко визначену екосистему пристроїв із сертифікатами сумісності, що забезпечує простоту налаштування і гарантовану взаємодію між пристроями. Швидкість передачі даних Z-Wave становить до 100 кбіт/с, що є достатнім для задач моніторингу та управління. Z-Wave забезпечує стабільну передачу даних на відстань до 100 метрів у відкритому просторі, а через mesh-мережу ця відстань може збільшуватися. Недоліками даної технології є вища вартість обладнання порівняно з Zigbee, обмежена кількість вузлів у мережі, а саме до 232 пристроїв та менша популярність у порівнянні з Zigbee, що обмежує кількість доступних пристроїв [8].

Обидві технології мають свої унікальні переваги та недоліки, які визначають їх застосування в конкретних системах. Zigbee краще підходить для великих мереж із багатьма вузлами, тоді як Z-Wave оптимальний для систем із високими вимогами до стабільності та сумісності пристроїв. Для мобільних роботів, залежно від задачі, вибір між цими технологіями може

залежати від вимог до енергоспоживання, дальності зв'язку та пропускну здатності.

Ще одним із новітніх методів є використання лазерних технологій для передачі інформації. Вони забезпечують високу точність і захищеність даних, однак недоліками є висока вартість обладнання і складність налаштування. Окрім фізичних методів передачі, важливу роль відіграють протоколи обміну даними, такі як MQTT, ROS і DDS. MQTT, наприклад, є простим і енергоефективним протоколом для передачі невеликих обсягів даних, але має обмеження в забезпеченні безпеки. Порівнюючи ці методи, важливо враховувати специфічні умови застосування мобільного робота, так як, наприклад, у системах зі строгою вимогою до енергоспоживання обирають технології, як-от Zigbee, а для високошвидкісної передачі великих обсягів даних – Wi-Fi чи 5G [9].

## 1.2 Аналіз сучасних методів кодування технічної інформації

Кодування технічної інформації є ключовим етапом у передачі даних про стан мобільного робота, оскільки забезпечує їх захист, компактність і ефективність передачі. Розглянемо сучасні методи кодування, які використовуються для збереження та обробки інформації в системах мобільної робототехніки.

Одним із поширених методів є бінарне кодування, яке передбачає представлення даних у вигляді послідовностей бітів (0 та 1). Цей підхід широко використовується через його простоту та сумісність із цифровими системами. Бінарне кодування має високу швидкість обробки, але через відсутність стиснення може вимагати значних ресурсів для зберігання великих обсягів інформації.

Такий метод, як стискання з втратами, наприклад, через алгоритми JPEG чи MP3, може бути застосоване для зменшення обсягу технічної

інформації, яка передається через зображення або аудіосигнали, у випадку мобільних роботів, які передають відео чи фото, дозволяє суттєво знизити вимоги до каналу зв'язку. Проте, використання таких методів може призводити до втрати важливих деталей, що є критичним у випадках, коли інформація повинна бути точною. Методи стискання без втрат, такі як алгоритми Huffman чи LZW, дозволяють кодувати дані так, щоб уникнути втрати інформації. Вони використовуються для зберігання технічних параметрів та діагностичних даних мобільних роботів. Перевагою є збереження точності даних, проте швидкість роботи алгоритмів може бути нижчою, ніж у методів зі стисненням з втратами [10].

У сучасних мобільних роботах також використовується кодування Хеммінга, яке дозволяє не лише передавати дані, але й виявляти та виправляти помилки, що виникають під час передачі. Це особливо важливо в умовах з високим рівнем перешкод. Недоліком методу є збільшення обсягу переданих даних через додаткові біти, які додаються для перевірки.

Циклічні коди (CRC) є ще одним ефективним методом для виявлення помилок у переданих даних. CRC широко використовується в комунікаційних протоколах, таких як Zigbee чи Bluetooth, що забезпечує надійність передачі даних у реальному часі. Однак, цей метод не виправляє помилки, а лише виявляє їх, тому потрібні додаткові механізми для корекції. Одним із новітніх підходів є кодування на основі штучних нейронних мереж, яке дозволяє створювати адаптивні системи для оптимального стиснення даних. Нейромережі можуть аналізувати структуру технічної інформації та забезпечувати найкраще співвідношення між якістю даних і їх обсягом. Недоліком такого підходу є висока обчислювальна складність і потреба в навчанні моделі [11].

Для передачі технічної інформації в зображенні часто застосовуються стеганографічні методи, які дозволяють вбудовувати дані у вигляд невидимих елементів. Наприклад, використання алгоритмів LSB (Least

Significant Bit) дає змогу передати інформацію, не змінюючи візуальних характеристик зображення. Основним недоліком є вразливість до модифікації даних, наприклад, при зміні формату зображення.

Метод квантового кодування є перспективною технологією, яка забезпечує високий рівень безпеки передачі інформації завдяки використанню квантових властивостей частинок. Для мобільних роботів ця технологія поки що перебуває на стадії досліджень, проте в майбутньому може забезпечити повну захищеність даних. Ще одним цікавим методом є застосування кодів LDPC (Low-Density Parity-Check Codes), які забезпечують високу ефективність передачі даних навіть у шумних каналах. LDPC використовуються в сучасних стандартах зв'язку, таких як 5G, і можуть знайти застосування у мобільних роботах із підвищеними вимогами до надійності [12].

### 1.3 Аналіз сучасних методів декодування технічної інформації

Декодування технічної інформації є важливим етапом у передачі даних, оскільки від цього залежить правильність відтворення отриманих повідомлень. Сучасні методи декодування включають алгоритми корекції помилок, стискання даних та адаптивного відновлення. Далі розглянемо найпоширеніші методи, їхні характеристики, переваги та обмеження.

Одним із базових методів декодування є бітове декодування, яке передбачає безпосереднє відновлення інформації на рівні бітів. Воно використовується для простих цифрових сигналів та протоколів, де дані передаються без додаткового стискання. Цей метод простий у реалізації та має високу швидкість, але не здатний виявляти чи виправляти помилки, що виникають у процесі передачі. Для забезпечення стійкості до помилок застосовуються алгоритми декодування за методом Хеммінга, які дозволяють виправляти одну або кілька помилок у переданих даних. Це робить метод

ефективним для передачі технічної інформації в умовах шумних каналів. Основним недоліком є підвищене навантаження на канали зв'язку через додавання контрольних бітів [13].

Ще одним популярним підходом є декодування циклічних кодів, що використовується для виявлення помилок. Цей метод застосовується у багатьох сучасних протоколах передачі даних, таких як Ethernet, Bluetooth і Zigbee. CRC дозволяє швидко виявляти помилки, проте виправлення таких помилок потребує додаткових засобів. А у випадках, коли дані передаються у стиснутому вигляді, застосовуються алгоритми декодування LZW, цей метод дозволяє ефективно відновлювати інформацію, стиснуту без втрат. Його перевагами є висока точність і універсальність, проте він може бути обчислювально затратним при роботі з великими обсягами даних.

Для стиснення з втратами використовуються специфічні алгоритми, такі як декодування JPEG або MPEG, які часто застосовуються для зображень чи відео. У мобільних роботах ці алгоритми можуть використовуватися для передачі візуальної інформації. Основним недоліком є втрата деталей при відновленні даних, що може бути критичним для технічних систем. Також, уваги заслуговує декодування на основі нейронних мереж, яке стає все популярнішим завдяки адаптивності та здатності обробляти великі обсяги інформації. Цей метод дозволяє не лише декодувати дані, а й прогнозувати можливі помилки, проте головним обмеженням є необхідність значних обчислювальних ресурсів і часу для навчання моделі. Далі наведемо порівняльну таблицю 1.1, основних методів декодування технічної інформації, представивши їх переваги та недоліки [14].

Таблиця 1.1 – Основні методи декодування технічної інформації

Метод декодування	Переваги	Недоліки
1	2	3
Бітове декодування	Висока швидкість, простота реалізації	Відсутність виявлення та виправлення помилок

Продовження таблиці 1.1

1	2	3
Декодування Хеммінга	Можливість виправлення помилок	Додаткове навантаження через контрольні біти
Циклічні коди	Швидке виявлення помилок	Не дозволяє виправляти помилки, потребує додаткових механізмів
Алгоритм LZW	Висока точність відновлення, стиснення без втрат	Обчислювальна складність
JPEG/MPEG	Ефективне стиснення для візуальних даних	Втрата якості при декодуванні
Нейронні мережі	Адаптивність, здатність прогнозувати помилки	Висока складність реалізації, необхідність навчання

Загальний аналіз сучасних методів декодування технічної інформації демонструє, що вибір конкретного підходу залежить від вимог до системи передачі даних. Просте бітове декодування забезпечує високу швидкість обробки, але не гарантує виявлення та виправлення помилок. Методи з корекцією, такі як декодування Хеммінга або CRC, підвищують надійність передачі в умовах шумних каналів, проте накладають додаткові вимоги на ресурсну базу. Стискання даних, реалізоване через алгоритми LZW чи JPEG, дозволяє зменшити обсяг інформації, але може призводити до втрати деталей, що критично у високоточних системах.

Інноваційні підходи, такі як нейронні мережі або квантове декодування, відкривають нові можливості для ефективною передачі даних, забезпечуючи адаптивність, високу точність та безпеку. Однак вони потребують значних обчислювальних ресурсів та спеціалізованого обладнання, що обмежує їх використання в умовах мобільної робототехніки [15].

#### 1.4 Постановка задач дослідження

В ході проведеного аналізу у першому розділі кваліфікаційної роботи, було виявлено, що тема дослідження є актуальною. Метою дослідження є підвищення ефективності передачі технічної інформації про стан мобільного робота, за рахунок вдосконалення методу шифрування даних в зображенні. Об'єктом дослідження є процес передачі технічної інформації. Предметом дослідження є методи, алгоритми та програмне забезпечення передачі інформації про стан мобільного робота. Для досягнення поставленої мети потрібно вирішити наступні завдання:

- провести дослідження метода LSB з точки зору кодування інформацій про стан мобільного робота
- провести математичний опис метода кодування технічної інформації
- провести математичний опис метода розкодування технічної інформації
- навести приклад кодування та розкодування технічної інформації
- провести вибір середовища розробки
- розробити загальний алгоритм роботи програми
- розробити функції кодування та розкодування технічної інформації
- провести експериментальні дослідження та аналіз отриманих результатів.

## 2 РОЗРОБКА МЕТОДА КОДУВАННЯ ТЕХНІЧНОЇ ІНФОРМАЦІЇ ПРО СТАН МОБІЛЬНОГО РОБОТА НА БАЗІ МЕТОДА LSB

### 2.1 Дослідження метода LSB з точки зору кодування інформації про стан мобільного робота

Порівняно з іншими методами кодування, такими як трансформаційні методи, що використовують дискретне косинусне перетворення (DCT) або хвилькові перетворення (DWT), LSB-кодування не змінює частотні характеристики зображення і не викликає артефактів компресії, що може негативно вплинути на якість відео. Порівняння методів LSB, DCT, DWT для кодування технічної інформації про стан мобільного робота в потоковому відео представлено в таблиці 2.1.

Таблиця 2.1 – Порівняння методів кодування LSB, DCT, DWT з точки зору використання для передачі інформації про стан мобільного робота в потоковому відео

Характеристика	Методи		
	LSB	DCT	DWT
1	2	3	4
Принцип роботи	Заміна найменш значущих бітів пікселів	Зміна коефіцієнтів DCT перетворення	Зміна коефіцієнтів DWT перетворення
Простота реалізації	Висока	Середня	Середня
Обчислювальна складність	Низька	Висока	Висока

Продовження таблиці 2.1

1	2	3	4
Помітність змін	Мінімальна	Середня	Середня
Гнучкість	Висока	Середня	Середня
Безпека	Низька	Висока (при додатковому шифруванні)	Висока (при додатковому шифруванні)
Непомітність	Висока	Середня	Середня
Використання ресурсів	Низьке	Високе	Високе
Приклади використання	Прості системи реального часу	Складні системи зі статичними зображеннями	Складні системи зі статичними зображеннями

Розглянемо кожен метод, представлений у таблиці 2.1 детальніше, метод DCT (Discrete Cosine Transform) є потужним інструментом для кодування технічної інформації про стан мобільного робота в потоковому відео. Він базується на перетворенні блоків зображення з часової області в частотну, що дозволяє ефективно приховувати інформацію у високочастотних компонентах зображення, які менш помітні для людського ока. Для кодування технічної інформації, такої як координати, швидкість або стан батареї робота, ці дані спочатку перетворюються в бінарний формат. Потім ці бінарні дані вбудовуються в DCT-коефіцієнти зображення.

Цей метод забезпечує високу ступінь стійкості до компресії, оскільки DCT використовується у багатьох алгоритмах стиснення зображень, таких як JPEG. Важливою перевагою DCT є те, що він дозволяє приховати значну кількість даних без значного спотворення зображення. Це робить його придатним для додавання додаткової технічної інформації без помітного

впливу на якість відео. Базова форма кодування методом DCT представлена у виразі (2.1):

$$DCT(u, v) = \sum I(x, y) \cos\left(\frac{(2x+1)u\pi}{2N}\right) \cos\left(\frac{(2y+1)v\pi}{2N}\right), \quad (2.1)$$

де  $DCT(u, v)$  – коефіцієнт DCT для частот  $(u, v)$ ;

$I(x, y)$  – інтенсивність пікселя  $(x, y)$  в оригінальному зображенні;

$N$  – розмір блоку перетворення (наприклад, 8 для 8x8 блоків);

$\Sigma$  – сума по всіх пікселях блоку.

Однак, метод DCT є обчислювально більш складним, порівняно з простішими методами, такими як LSB, і потребує більшого обсягу ресурсів для кодування та декодування інформації. В загальному, DCT є ефективним методом для приховання технічної інформації в відео, забезпечуючи баланс між якістю зображення та кількістю прихованих даних.

Наступний метод DWT (Discrete Wavelet Transform) є ефективним підходом для кодування технічної інформації про стан мобільного робота в потоковому відео. Він працює шляхом розкладання зображення на різні частотні діапазони, що дозволяє ефективно приховувати інформацію у малопомітних областях зображення. Завдяки використанню хвилькових перетворень, DWT може зберігати високий рівень деталей і забезпечувати високу стійкість до різних видів компресії.

Для кодування технічної інформації, такої як координати робота, швидкість або стан батареї, ці дані конвертуються в бінарний формат і вбудовуються в коефіцієнти DWT. Цей підхід дозволяє приховати велику кількість даних без помітного впливу на якість зображення. Однією з ключових переваг DWT є його здатність ефективно обробляти зображення з високою роздільною здатністю, що робить його ідеальним для використання в сучасних відеопотоках. Базова форма кодування методом DWT представлена у виразі (2.2):

$$DWT(a, b) = \frac{1}{\sqrt{a \int I(t) \varphi\left(\frac{t-b}{a}\right) dt}}, \quad (2.2)$$

де  $DWT(a, b)$  – коефіцієнт DWT для масштабів ( $a$ ) і зсувів ( $b$ );

$I(t)$  – сигнал (або зображення) у часовій області;

$\varphi\left(\frac{t-b}{a}\right)$  – базисна хвилькова функція (wavelet).

Однак, метод DWT є обчислювально інтенсивним і потребує більше ресурсів для реалізації, що може бути викликом для систем з обмеженими ресурсами. Незважаючи на це, його здатність зберігати високу якість зображення та забезпечувати приховання даних робить DWT привабливим вибором для кодування технічної інформації в потоковому відео.

Метод найменш значущих бітів (LSB) є одним із найефективніших і найбільш популярних методів стеганографії для кодування інформації у зображеннях, особливо в контексті передачі технічної інформації про стан мобільного робота в потоковому відео. Однією з основних переваг методу LSB є його простота та швидкість виконання. Процеси кодування та розкодування інформації не потребують великих обчислювальних ресурсів, що робить цей метод особливо привабливим для систем реального часу, таких як мобільні роботи, які мають обмежені ресурси для обробки даних. Базова форма кодування методом LSB представлена у виразі (2.3):

$$T_b[l] = (I_{i,j,k}) \& 1, \quad (2.3)$$

де  $T_b[l]$  – біт, який кодується у найменш значущий біт;

$I_{i,j,k}$  – змінений колірний канал пікселя ( $i, j$ ) у зображенні каналу  $k$ ;

1-бітова маска для вилучення найменш значущого біту.

LSB кодування відрізняється низькою обчислювальною складністю, що робить його привабливим для реального часу додатків, таких як моніторинг мобільних роботів. Крім того, цей метод не потребує складних алгоритмів і може бути легко реалізований на апаратному рівні, що є важливим для систем з обмеженими ресурсами.

Недоліком LSB є його чутливість до атак і змін, таких як стиснення або фільтрація, які можуть легко знищити приховану інформацію. Незважаючи на це, LSB є популярним методом для передачі технічної інформації в потоковому відео завдяки своїй простоті й ефективності, особливо в умовах, де точність передачі даних не є критичною [16].

## 2.2 Математичний опис метода кодування технічної інформації

Нехай, в рамках даного дослідження введемо наступні позначення:

$I$  – матриця зображення (кадру відео) розміром:

$$I = H \times W \times 3, \quad (2.4)$$

де  $H$  – висота зображення;

$W$  – ширина зображення;

$3$  – кількість каналів (R, G, B), кількість каналів може змінюватися в залежності від потрібного формату відео потоку;

$T$  – технічна інформація про стан мобільного робота, яка включає наступні дані:

- поточні координати робота ( $X, Y, Z$ );
- напрямок руху (кутові положення);
- швидкість руху;
- стани сенсорів (температура, вологість, тощо);
- інші параметри (наприклад, рівень заряду батареї);

$T_b$  – технічна інформація в бінарному вигляді;

$I'$  – матриця зображення з закодованою технічною інформацією.

На першому кроці розробляемого методу проведемо конвертацію технічної інформації у бінарний вигляд. Представимо, що  $T$  складається з  $n$  символів, кожний символ можна представити як 8-бітний бінарний код. Виходячи з цього, технічна інформація в бінарному вигляді ( $T_b$ ) можна представити в рамках даних досліджень наступним чином, як представлено у виразі (2.5).

$$T_b = \bigcup_{i=1}^n bin(T_i), \quad (2.5)$$

де  $bin(T_i)$  – бінарне представлення  $i$ -го символу з технічної інформації про стан мобільного робота ( $T$ ).

Виходячи з (2.4) та (2.5), наступним кроком можна приступити до кодування бінарної інформації ( $T_b$ ), використовуючи найменш значущі біти пікселів зображення  $I$ . Найменш значущі біти (LSB) пікселів в методі LSB (Least Significant Bit) кодування – це біти, які мають найменшу вагу в числовому представленні значення кольору пікселя. У контексті кольорових зображень, кожен піксель зазвичай представлений трьома значеннями (для червоного, зеленого та синього каналів, або RGB). Кожне з цих значень коливається від 0 до 255 і записується у вигляді 8-бітного двійкового числа. Найменш значущі біти – це останні біти в цьому двійковому числі, які мають найменший вплив на загальне значення кольору. Наприклад, для значення 255, яке у двійковій формі виглядає як 11111111, найменш значущим бітом є останній '1'. У методі LSB стеганографії ці біти використовуються для прихованого запису інформації. Зміна найменш значущого біта призводить до дуже незначної зміни кольору, яка непомітна для людського ока. Це дозволяє приховувати текстові дані або інші бінарні дані в зображеннях без

видимих змін візуальної якості зображення. Завдяки цьому методу можливо ефективно приховувати та передавати інформацію через звичайні зображення.

Нехай  $I_{i,j,k}$  – значення каналу  $k$ , в рамках даних досліджень це  $(R, G, B)$ , пікселя на позиції  $(i, j)$ . Припустимо, що в розробляємому методі змінюємо цей піксель за наступною формулою:

$$\hat{I}_{i,j,k} = (I_{i,j,k} \& \sim 1) | T_b[l], \quad (2.6)$$

де  $\hat{I}_{i,j,k}$  – значення каналу  $k$  з закодованою технічною інформацією;

$I_{i,j,k}$  – значення каналу  $k$   $(R, G, B)$ , пікселя на позиції  $(i, j)$ ;

$\& \sim 1$  – це операція AND з інвертованим 1 (тобто, змінюємо найменш значущий біт до 0);

$T_b[l]$  – біт з інформації  $T_b$ , з індексом біта  $l$ .

Виходячи з формули (2.6), кодування інформації у зображенні можна представити шляхом змінювання пікселів ( $\hat{I}_{i,j,k}$ ) за наступною формулою:

$$\hat{I}_{i,j,k} = (I_{i,j,k} \& 254) | b, \quad (2.7)$$

де  $\hat{I}_{i,j,k}$  – нове значення кольору з прихованою інформацією;

$I_{i,j,k}$  – початкове значення кольору;

$\& 254$  – це операція AND з  $11111110_2$  (всі біти 1, крім LSB);

$|$  – побітова операція OR;

$b$  – біт з інформацією  $T_b$ , який потрібно закодувати.

Приведемо приклад кодування інформації. Як було зазначено вище, першим кроком, необхідно перетворити інформацію у бінарний вигляд. Нехай  $T$  – це рядок з технічною інформацією наступного виду: "Відстань до

перешкоди: 15 см.". Проведемо конвертацію кожного символу у 8-бітний код ( $T_b$ ) за наступним виразом:

$$T_b = \text{bin}(T_1) \cup \text{bin}(T_2) \cup \dots \cup \text{bin}(T_n), \quad (2.8)$$

Тобто «В»  $\rightarrow$  10001000, «і»  $\rightarrow$  10001001, «д»  $\rightarrow$  10001010, «с»  $\rightarrow$  10010000 і т.д. Відповідно до (2.8) отримаємо результуючий бінарний рядок:

$$T_b = 100010001000100110001010 \dots 00101110. \quad (2.9)$$

Проведемо кодування інформації у пікселі зображення, на базі формули (2.4). Нехай початковий піксель має значення  $(I_{i,j,0}, I_{i,j,1}, I_{i,j,2}) = (101, 202, 153)$ , тоді бінарне представлення початкових значень буде мати наступний вид:

$$101 \rightarrow 01100101;$$

$$202 \rightarrow 11001010;$$

$$153 \rightarrow 10011001.$$

Відповідно до (2.9) отримаємо результуючий бінарний рядок, який буде мати наступний вигляд:

$$T_b = 1000100010001001 \dots \quad (2.10)$$

Проведемо кодування першого біта  $T_b = 1$  у першому каналі  $I_{i,j,0}$ :

$$\begin{aligned} \hat{I}_{i,j,0} &= (101 \& 254) | 1 = (01100101 \& 11111110) | 1 = \\ &= 01100100 | 00000001 = 01100101 = 101. \end{aligned} \quad (2.11)$$

Проведемо кодування другого біта  $T_b = 0$  у другому каналі  $I_{i,j,1}$ :

$$\begin{aligned} \hat{I}_{i,j,1} &= (202 \& 254) | 0 = (11001010 \& 11111110) | 0 = \\ &= 11001010 | 00000000 = 11001010 = 202. \end{aligned} \quad (2.12)$$

Проведемо кодування третього біта  $T_b = 0$  у третьому каналі  $I_{i,j,2}$ :

$$\begin{aligned} \hat{I}_{i,j,2} &= (153 \& 254) | 0 = (10011001 \& 11111110) | 0 = \\ &= 10011000 | 00000000 = 10011000 = 152. \end{aligned} \quad (2.13)$$

Як можна бачити з виразів (2.11)-(2.13) нове значення пікселя з прихованою інформацією буде мати наступний вигляд:

$$(\hat{I}_{i,j,0}, \hat{I}_{i,j,1}, \hat{I}_{i,j,2}) = (101, 202, 152). \quad (2.14)$$

Запропонований метод кодування інформації у пікселі зображення за допомогою найменш значущих бітів (LSB) має кілька важливих переваг для передачі технічної інформації про складові мобільного робота. По-перше, цей метод дозволяє приховувати інформацію без помітних змін у якості зображення, що важливо для забезпечення непомітності даних. Це означає, що передача даних може відбуватися приховано, не викликаючи підозри у сторонніх спостерігачів. По-друге, метод LSB є досить простим у реалізації та не потребує складних обчислювальних ресурсів, що дозволяє легко інтегрувати його в реальні системи мобільних роботів. Це особливо корисно для роботів з обмеженими обчислювальними можливостями. Крім того, LSB метод забезпечує високу швидкість кодування та декодування інформації, що є критичним для реального часу передачі даних. Наприклад, можна швидко оновлювати інформацію про поточну позицію робота, рівень заряду батареї, швидкість руху та відстань до перешкод. Цей метод також є універсальним, оскільки може використовуватися з різними форматами зображень і типами даних. Ще однією перевагою є те, що метод LSB не вимагає значного

збільшення обсягу даних для зберігання додаткової інформації, оскільки використовуються вже наявні пікселі зображення. Це дозволяє ефективно використовувати ресурси пам'яті та зберігати великі обсяги технічних даних. Нарешті, метод LSB може бути легко адаптований для різних застосувань у мобільній робототехніці, таких як навігація, діагностика систем та моніторинг стану робота. Завдяки своїй простоті, ефективності та універсальності, метод LSB є потужним інструментом для прихованої передачі технічної інформації в мобільних роботах [17].

### 2.3 Математичний опис метода розкодування технічної інформації

Розкодування інформації, закодованої методом найменш значущих бітів (LSB), полягає в читанні прихованих бітів з пікселів зображення та відновленні початкової текстової інформації. Тобто, на першому кроці, необхідно провести знаходження та витягування бінарних даних із зображення. Це можливо описати наступним чином, хай для кожного пікселя  $(i, j)$  каналу  $k$ :

$$T_b[l] = \hat{I}_{i,j,k} \& 1, \quad (2.15)$$

де  $T_b[l]$  – це  $l$ -й біт бінарного рядка  $T_b$ , який містить закодовану інформацію. Розкодування інформації починається зі зчитування найменш значущих бітів зображення та додавання їх до цього рядка;

$\hat{I}_{i,j,k}$  – значення кольору  $(R, G, B)$  в пікселі з координатами  $(i, j)$  та каналом  $k$  у матриці зображення  $\hat{I}$ , яке містить приховану інформацію. Матриця зображення має розміри відповідно до 2.4;

$i$  – індекс рядка в матриці зображення  $\hat{I}$ , який визначає вертикальну позицію пікселя в зображенні. Відповідно що  $i$  варіюється від 0 до  $H - 1$ ;

$j$  – індекс стовпця в матриці зображення  $\hat{I}$ , який визначає горизонтальну позицію пікселя в зображенні. Відповідно, що  $j$  варіюється від 0 до  $W - 1$ ;

$k$  – індекс кольорового каналу (0 для червоного, 1 для зеленого, 2 для синього) у пікселі з координатами  $(i, j)$ , де  $k$  варіюється від 0 до 2;

$\&1$  – побітова операція AND з числом 1, яка виділяє найменш значущий біт (LSB) значення кольору. Ця операція встановлює всі біти, крім найменш значущого, у 0, залишаючи лише LSB. Якщо  $\hat{I}_{i,j,k}$  – значення кольору у пікселі, то  $\hat{I}_{i,j,k} \& 1$  повертає 0 або 1, залежно від значення LSB у цьому пікселі.

Наступним кроком проводимо конвертацію бінарних даних у символи, тобто об'єднуємо біти у байти (8 біт кожен) та конвертуємо кожен байт у символ з використанням наступного виразу:

$$T_i = \text{char}\left(\sum_{m=0}^7 T_b[8(i-1) + m] \cdot 2^m\right), \quad (2.16)$$

де  $T_i$  – Це  $i$ -й символ у результуючому текстовому рядку  $T$ . Кожен символ відновлюється з 8 бітів бінарних даних;

$\text{char}()$  – функція, яка перетворює цілочисельне значення в символ. Вона використовує ASCII код для перетворення чисел у відповідні символи;

$\sum_{m=0}^7$  – сума, яка обчислюється по змінній  $m$  від 0 до 7. Цей вираз обчислює значення одного символу з 8 бітів бінарних даних;

$T_b[8(i-1) + m]$  – це  $(8(i-1) + m)$ -й біт у бінарному рядку  $T_b$ . Бінарний рядок  $T_b$  містить всі біти, які були закодовані в пікселях зображення;

$2^m$  – ступінь двійки, який використовується для перетворення бінарних даних у десяткове число. Ступінь двійки визначає позиційну вагу кожного біта.

Приведемо приклад розкодування, нехай у нас є закодоване зображення з пікселем  $(\hat{I}_{i,j,0}, \hat{I}_{i,j,1}, \hat{I}_{i,j,2}) = (101, 202, 152)$ , відповідно до

виразу (2.14). Виходячи з цього, бінарне представлення значень з найменш значущими бітами, можна описати наступним чином:

$$\begin{aligned}
 101 &\rightarrow 01100101, \text{ найменш значущий біт} = 1; \\
 202 &\rightarrow 11001010, \text{ найменш значущий біт} = 0; \\
 152 &\rightarrow 10011000, \text{ найменш значущий біт} = 0.
 \end{aligned}
 \tag{2.17}$$

Виходячи з (2.17), отримуємо наступний бінарний рядок, який представлено нижче:

$$T_b = 0101011010011110 \dots, \tag{2.18}$$

Перетворимо кожен групу по 8 бітів у символ та отримуємо наступний результат:

$$\begin{aligned}
 01010110 &\rightarrow 86 \rightarrow 'V'; \\
 10011110 &\rightarrow 158 \rightarrow 'i'.
 \end{aligned}$$

Таким чином, проведемо розкодування технічної інформації, яка має наступні дані: «Відстань до перешкоди: 15 см».

Розкодування інформації, закодованої методом найменш значущих бітів (LSB), для передачі технічної інформації про стан мобільного робота має кілька значних переваг. По-перше, цей метод дозволяє передавати важливу інформацію без видимих змін у якості відеопотоку, що зберігає візуальну привабливість і не викликає підозр щодо наявності прихованих даних. Завдяки використанню лише найменш значущих бітів, які мають мінімальний вплив на загальний колір пікселів, метод забезпечує високий рівень непомітності та не порушує загальну структуру зображення. LSB-кодування є ефективним з точки зору обчислювальної потужності, оскільки

процеси кодування та розкодування є досить простими та швидкими, що важливо для систем реального часу, таких як мобільні роботи. Метод також дозволяє легко додавати або видаляти інформацію без необхідності значних змін у відеопотоці. Завдяки цьому, LSB-кодування є гнучким і може використовуватися для передачі різних типів даних, таких як координати, швидкість, рівень заряду батареї та інші параметри роботи. З точки зору безпеки, цей метод забезпечує базовий рівень захисту, оскільки прихована інформація не є очевидною для сторонніх спостерігачів. У випадку необхідності більшого рівня безпеки, можна комбінувати LSB-кодування з іншими методами шифрування для підвищення стійкості до атак [18].

#### 2.4 Приклад кодування та розкодування технічної інформації

Метод кодування LSB (Least Significant Bit) дозволяє приховувати інформацію в найменш значущих бітах пікселів зображення. Нижче наведено приклад кодування та розкодування технічної інформації, такої як поточні координати робота (X, Y, Z), напрямок руху (кутові положення), швидкість руху, стани сенсорів (температура, вологість), рівень заряду батареї, з використанням методу LSB. Припустимо, що на вхід для кодування маємо наступну технічну інформацію про стан мобільного робота, що представлена на рисунку 2.1.

```
Поточні координати (X, Y, Z): (123, 456, 789)
Напрямок руху (кут): 45°
Швидкість руху: 1.5 м/с
Температура: 23°C
Вологість: 45%
Рівень заряду батареї: 80%
```

Рисунок 2.1 – Вхідна технічна інформація про стан мобільного робота  
(Монітор порта середовища Arduino IDE)

Дану інформацію потрібно перетворити на рядок, який буде мати наступний вид, як представлено на рисунку 2.2.



"123,456,789;45;1.5;23;45;80"

Рисунок 2.2 – Загальний вид рядка з технічною інформацією про стан мобільного робота

Наступним кроком проведемо перетворення рядка (рис. 2.2) на бінарний формат, загальний вид якого представлено нижче:

$$T_b = [00110001\ 00110010\ 00110011\ 00101100\ 00110100\ 00110101\ 00110110\ 00101100\ 00111000\ 00111001\ 00101100\ 00111010\ 00101101\ 00110100\ 00110101\ 00101100\ 00111001\ 00101100\ 00101110\ 00110101\ 00101100\ 00110010\ 00110111\ 00101100\ 00111001\ 0011011\ 00101100]$$

Для перевірки правильності запропонованих математичних виразів реалізації метода LSB для кодування технічної інформації (рис. 2.1) про стан мобільного робота, пропонується розробити програму на базі мови Python в середовищі PyCharm 2022.2.3. Приклад програмного кода кодування методом LSB технічної інформації представлено нижче:

```
import cv2
```

Ця команда імпортує бібліотеку OpenCV, яка використовується для обробки зображень та роботи з відео.

```
def string_to_binary(data):
```

```
    return ''.join(format(ord(char), '08b') for char in data)
```

Ця функція перетворює текстові дані (data) у рядок бінарних чисел ('0' і '1'). Кожен символ тексту перетворюється на восьмибітовий бінарний код за допомогою ord(char) і форматується до вигляду '08b'

```

def encode_data_in_frame(frame, data):
    binary_data = string_to_binary(data)
    data_len = len(binary_data)

    h, w, _ = frame.shape
    max_data_len = h * w * 3 # Each pixel has 3 channels (RGB)
    if data_len > max_data_len:
        raise ValueError("Data is too large to fit in the frame.")

    data_index = 0
    for i in range(h):
        for j in range(w):
            if data_index < data_len:
                pixel = list(frame[i, j])
                for k in range(3):
                    if data_index < data_len:
                        bit_value = int(binary_data[data_index])
                        # Clear the least significant bit
                        pixel[k] &= 254
                        # Set the least significant bit to bit_value
                        pixel[k] |= bit_value
                        data_index += 1
                frame[i, j] = tuple(pixel)
    return frame

```

Ця функція приймає зображення (`frame`) та дані для кодування (`data`). Вона перетворює дані у бінарний вигляд, обчислює максимальну довжину даних, яку можна вбудувати у кадр зображення, і вставляє ці дані в найменш значущі біти кожного пікселя кожного каналу RGB. Це виконується за допомогою операцій побітового І та ІЛИ (&, |)

```
technical_data
```

```
=
```

```
"X:123,Y:456,Z:789;Angle:45;Speed:1.5;Temp:23;Humidity:45;Battery:80"
```

Ця змінна містить технічні дані для кодування в зображенні.

```
cap = cv2.VideoCapture(0)
```

Ця команда відкриває підключену вебкамеру (або інші джерела відео) для захоплення відеокадрів. 0 вказує на перший доступний джерело відео.

```
while True:
```

```
    ret, frame = cap.read()
```

```
    if not ret:
```

```
        print("Error: Could not read frame from camera.")
```

```
        break
```

```
    # Відображення кадру
```

```
    cv2.imshow('Camera Feed', frame)
```

```
    # Натискання клавіші 'c' для кодування інформації
```

```
    if cv2.waitKey(1) & 0xFF == ord('c'):
```

```
        try:
```

```
            frame_with_data = encode_data_in_frame(frame.copy(),
```

```
technical_data)
```

```
            cv2.imwrite('cod.jpg', frame_with_data)
```

```
            print("Image encoded and saved successfully.")
```

```
        except ValueError as e:
```

```
            print(f"Error during encoding: {e}")
```

```
            continue
```

```
    # Очікування натискання клавіші 'q' для виходу
```

```
    if cv2.waitKey(1) & 0xFF == ord('q'):
```

```
        break
```

В цьому блоці коду захоплюється кожен кадр з камери, він відображується на екрані (`cv2.imshow`) та чекає натискання клавіш. Клавіша 'c' запускає кодування інформації та збереження зображення з вбудованою інформацією в файл `cod.jpg`.

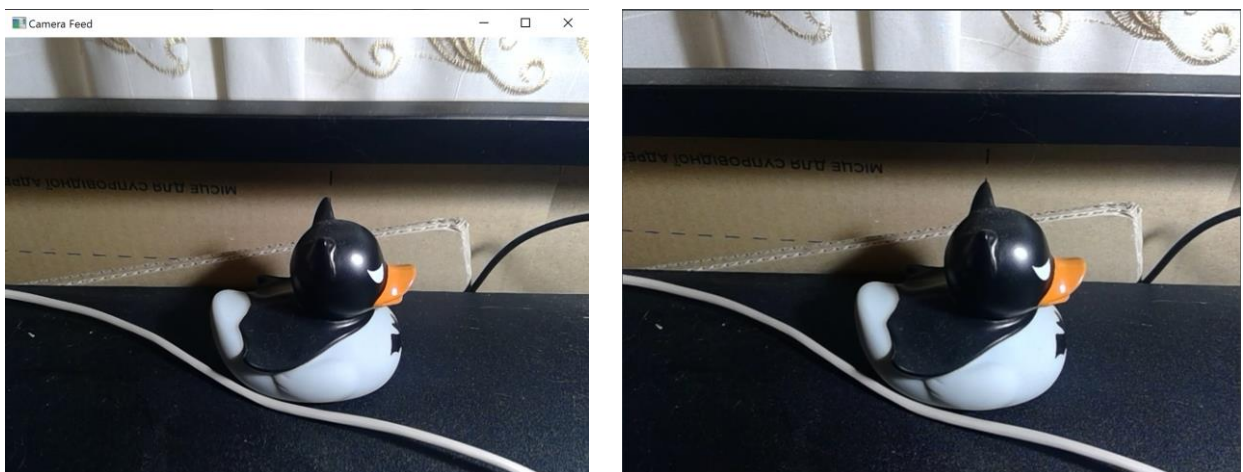
```
cap.release()
```

```
cv2.destroyAllWindows()
```

Після завершення циклу обробки відео вебкамера звільняється (`cap.release()`) і всі вікна OpenCV закриваються (`cv2.destroyAllWindows()`).

Цей код дозволяє вбудовувати технічну інформацію в зображення, отримане з камери, та зберігати його для подальшого використання.

Результати кодування технічної інформації про стан мобільного робота методом LSB представлено на рисунку 2.3.



а)

б)

а) відео потік з камери в режимі реального часу;

б) кодоване зображення в форматі \*.jpg з технічною інформацією про стан мобільного робота

Рисунок 2.3 – Результати роботи програми кодування методом LSB технічної інформації про стан мобільного робота

Як можна бачити з рисунку 2.3, результати роботи програми кодування методом LSB технічної інформації про стан мобільного робота, отримане

зображення з закодованої технічної інформації (рис. 2.3, б) візуально ніяк не відрізняється від потокового відео, що бачить оператор через систему комп'ютерного зору мобільного робота. Це доказує, що запропонований метод кодування на базі LSB не буде заважати оператору оцінити стан оточуючого середовища мобільного робота та не впливає на затримку відопотока з камери мобільного робота.

Для перевірки правильності розкодування технічної інформації про стан мобільного робота, яка зашифрована методом LSB в зображенні (рис. 2.3, б), пропонується наступна програма, написана мовою Python в середовищі PyCharm 2022.2.3.

```
import cv2
```

```
import numpy as np
```

Імпорт бібліотек OpenCV і NumPy для роботи з зображеннями і масивами даних.

```
def binary_to_string(binary_data):
```

```
    chars = [binary_data[i:i+8] for i in range(0, len(binary_data), 8)]
```

```
    text = ".join(chr(int(char, 2)) for char in chars)
```

```
    return text
```

Ця функція перетворює бінарний рядок даних у текстовий рядок, декодуючи кожні 8 біт у відповідний символ ASCII.

```
def decode_data_from_frame(frame):
```

```
    h, w, _ = frame.shape
```

```
    binary_data = "
```

```
    for i in range(h):
```

```
        for j in range(w):
```

```
            pixel = frame[i, j]
```

```
            for k in range(3):
```

```
                lsb = pixel[k] & 1
```

```
binary_data += str(lsb)
```

```
decoded_text = binary_to_string(binary_data)
```

```
return decoded_text
```

Ця функція отримує кадр з зображенням, витягує найменш значущий біт з кожного каналу RGB кожного пікселя та конструює з цього бінарний рядок, який потім декодує у текстовий формат.

```
encoded_image = cv2.imread('cod.jpg')
```

Завантажується закодоване зображення з файлу cod.jpg.

```
decoded_info = decode_data_from_frame(encoded_image)
```

Застосовується функція розкодування для отримання технічної інформації з зображення.

```
cv2.putText(encoded_image, decoded_info, (20, 40),
cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 255, 0), 2)
```

Вставляє отриману розкодовану інформацію на зображення, щоб показати її на екрані.

```
cv2.imshow('Decoded Image', encoded_image)
```

```
cv2.waitKey(0)
```

```
cv2.destroyAllWindows()
```

Відображає розкодоване зображення з відображеною на ньому інформацією та очікує натискання будь-якої клавіші для закриття вікна.

Цей код розкодує та відображає технічну інформацію, приховану в найменш значущих бітах RGB пікселів зображення, на екрані поверх самого зображення.

Результат декодування технічної інформації про стан мобільного робота, яка закодована в файлі cod.jpg (рис. 2.3, б) представлено на рисунку 2.4.

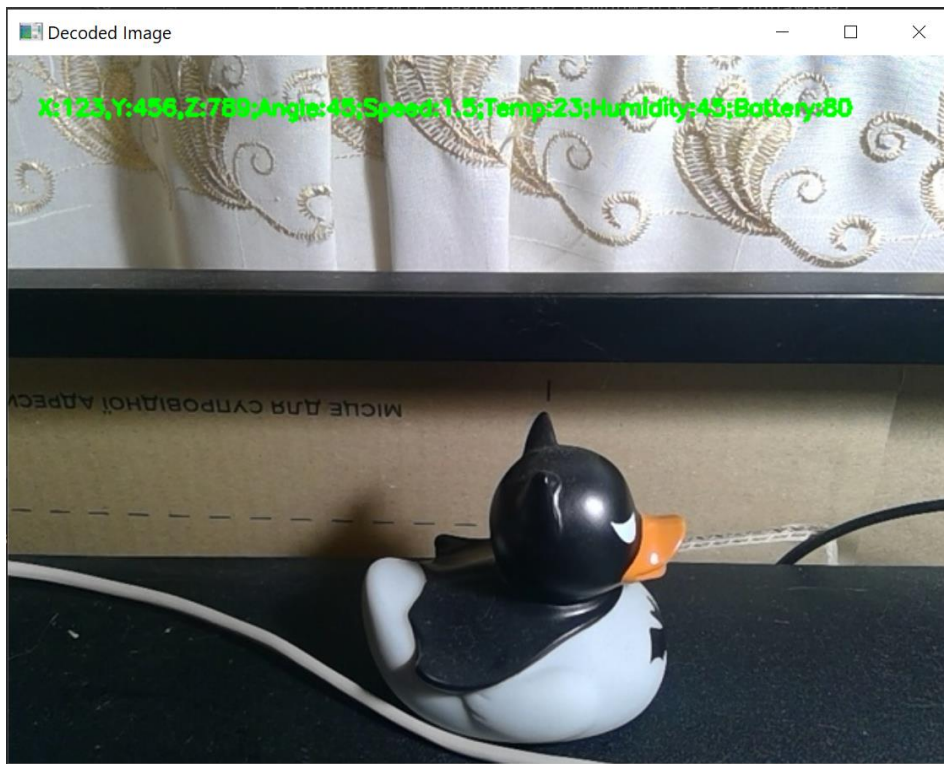


Рисунок 2.4 – Результат декодування технічної інформації про стан мобільного робота, яка закодована в файлі cod.jpg методом LSB

Метод LSB для декодування технічної інформації з файлу cod.jpg має низку значущих переваг. Він простий у реалізації, що робить його доступним для широкого кола користувачів і розробників без спеціалізованого математичного або програмного фону. Технічна інформація легко вбудовується без зміни сутності зображення, і LSB дозволяє приховувати дані, не змінюючи візуального вигляду зображення, що є важливим для захисту конфіденційності. Метод може дозволяти вбудовувати велику кількість додаткової інформації, залежно від роздільної здатності зображення та його розміру. Внесені зміни за допомогою LSB мають мінімальний вплив на якість зображення або відчуття спостерігача, оскільки вони впливають тільки на найменш значущі біти. LSB може бути застосований для вбудовування не лише текстової інформації, але й інших форматів даних, таких як звуки, відео чи інші типи файлів, що робить його універсальним методом для різних використань. Він підтримує різні формати зображень,

такі як JPEG, PNG, BMP тощо, що розширює його застосування на різноманітні платформи та сценарії. Операції зміни LSB в пікселях зазвичай не вимагають великої обчислювальної потужності, що дозволяє ефективно використовувати метод на різних пристроях та платформах. При наявності оригінального зображення та знання методу вбудовування, інформацію, закодовану за допомогою LSB, можна відновити без втрати точності чи повноти даних. Метод є одним з ключових у стеганографії, що дозволяє приховувати інформацію, не викликаючи підозри, що має значення у військових, правоохоронних та інших сферах. LSB також може бути використаний разом з іншими методами криптографії або захисту даних для забезпечення високого рівня безпеки і конфіденційності інформації. Загалом, метод LSB є потужним і універсальним інструментом для приховування технічної інформації у зображеннях, забезпечуючи ефективність, надійність і низьку помітність змін.

## 2.5 Висновки до 2 розділу

У ході написання 2 розділу кваліфікаційної роботи було розглянуто методи кодування технічної інформації про стан мобільного робота в потоковому відео – LSB, DCT та DWT – мають свої переваги й недоліки залежно від специфіки застосування. Метод LSB виділяється своєю простотою реалізації, мінімальними обчислювальними затратами та можливістю застосування в системах реального часу. Його недоліком є низька стійкість до змін у відео, таких як стиснення чи фільтрація. DCT, навпаки, забезпечує високу стійкість до компресії та здатний приховувати значний обсяг даних без суттєвого спотворення зображення, але потребує більших обчислювальних ресурсів. DWT, завдяки здатності зберігати високу якість зображення та ефективно обробляти високороздільні відеопотоки, є перспективним для сучасних систем, хоча його реалізація є складнішою.

Проведений аналіз показав, що кожен із методів має унікальні властивості, які можуть бути оптимальними в залежності від цілей застосування. LSB є найкращим вибором для простих і швидких рішень у системах з обмеженими ресурсами, тоді як DCT та DWT забезпечують високий рівень стійкості й ефективності для складніших задач. У результаті використання цих методів можливе забезпечення балансу між якістю передачі даних, ефективністю їх кодування та обчислювальними витратами, що робить їх придатними для різноманітних застосувань у робототехніці та інших галузях.

## 3 РОЗРОБКА ПРОГРАМИ КОДУВАННЯ ТА РОЗКОВУВАННЯ ТЕХНІЧНОЇ ІНФОРМАЦІЇ НА БАЗІ ПОТОКОВОГО ВІДЕО

### 3.1 Вибір середовища розробки

Для розробки програми кодування та декодування технічної інформації в зображеннях методом LSB оптимальним вибором є мова програмування Python та середовище розробки PyCharm. Python відомий своєю простотою та читабельністю коду, що дозволяє швидко створювати та тестувати алгоритми, включаючи стеганографічні методи, такі як LSB. Бібліотеки Python, такі як OpenCV, надають потужні інструменти для роботи із зображеннями, що значно спрощує процес обробки пікселів та вбудовування даних [19].

PyCharm, як інтегроване середовище розробки (IDE), пропонує зручний інтерфейс для написання, налагодження та тестування коду. Його функції, такі як автоматичне завершення коду, інтеграція з системами контролю версій та зручні інструменти для візуалізації, підвищують ефективність розробки. PyCharm також підтримує віртуальні середовища, що дозволяє легко керувати залежностями проекту та забезпечувати стабільність і відтворюваність результатів.

Використання Python і PyCharm спрощує інтеграцію з іншими бібліотеками для криптографії та обробки даних, що може бути корисним для розширення функціональності програми. Відкритість та велика спільнота користувачів Python забезпечують доступ до численних ресурсів та підтримки, що є важливим для вирішення будь-яких проблем під час розробки. Загалом, Python та PyCharm надають розробникам зручні та потужні інструменти для реалізації складних стеганографічних методів, забезпечуючи при цьому високу продуктивність і зручність роботи [20].

### 3.2 Розробка загального алгоритму роботи програми

Розробка загального алгоритму для кодування та декодування технічної інформації про стан мобільного робота методом LSB є необхідною з кількох ключових причин. Такий алгоритм дозволяє забезпечити надійний і ефективний спосіб передачі важливої інформації про робота, включаючи координати, напрямок руху, швидкість, температуру, вологість та рівень заряду батареї. Використання методу LSB дозволяє приховувати дані в зображеннях без значного впливу на їхню якість, що робить цей метод ідеальним для стеганографії.

Однією з важливих особливостей при розробці алгоритму є забезпечення точності та повноти переданої інформації. Для цього необхідно враховувати, що бінарні дані повинні бути правильно закодовані та декодовані, щоб уникнути втрати інформації або помилок при передачі. Це вимагає ретельного тестування та валідації алгоритму на різних етапах розробки. Ще однією важливою особливістю є забезпечення сумісності з різними типами зображень та форматами файлів. Алгоритм повинен коректно працювати з різними розширеннями зображень і враховувати їх специфіку.

Також необхідно врахувати обмеження на розмір даних, які можуть бути вбудовані в одне зображення, щоб уникнути перенасичення і забезпечити якісне відтворення зображення. Важливим аспектом є також забезпечення безпеки даних. Хоча метод LSB сам по собі не є криптографічним, можна додатково використовувати шифрування даних перед їх вбудовуванням у зображення для підвищення рівня безпеки.

При розробці алгоритму також слід враховувати можливість інтеграції з іншими системами та компонентами мобільного робота, щоб забезпечити безперервну передачу даних у реальному часі. Необхідно забезпечити простоту та зручність використання алгоритму, щоб він міг бути легко

впровадженій у існуючі системи управління роботом. Алгоритм повинен бути оптимізований для роботи на обмежених обчислювальних ресурсах мобільних роботів, що передбачає ефективне використання пам'яті та обчислювальної потужності. Загалом, розробка такого алгоритму забезпечує надійну, безпечну та ефективну передачу важливої технічної інформації, що є критичним для функціонування мобільних роботів у різних середовищах. Загальний алгоритм роботи програми кодування та розкодування технічної інформації про стан мобільного робота у відеопотоці методом стеганографії, представлено на рисунку 3.1.

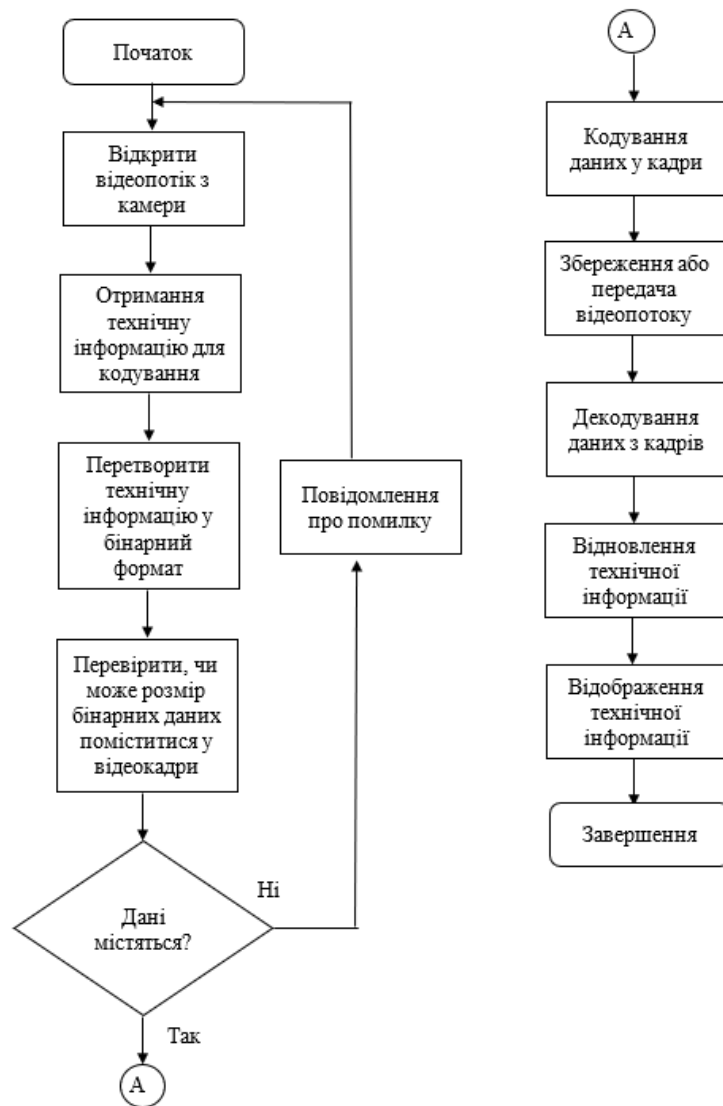


Рисунок 3.1 – Загальний алгоритм роботи програми кодування та розкодування технічної інформації про стан мобільного робота

Опишемо призначення всіх блоків загального алгоритму роботи програми кодування та розкодування технічної інформації про стан мобільного робота (рис. 3.1):

– блок "Відкрити відеопотік з камери" загального алгоритму призначений для ініціалізації доступу до відеокамери мобільного робота або іншого відеоджерела. Це необхідно для отримання реального відеопотоку, на який буде накладатися технічна інформація про стан мобільного робота. Він також забезпечує безперервне захоплення кадрів для подальшої обробки і кодування;

– блок "Отримання технічної інформації для кодування" призначений для збору та підготовки поточних даних про стан мобільного робота, таких як координати, напрямок руху, швидкість, температура, вологість і рівень заряду батареї. Ця інформація буде використана для подальшого кодування у відеопотік, що дозволяє вбудовувати корисні дані в кадри відео для подальшого розкодування та аналізу;

– блок "Перетворити технічну інформацію у бінарний формат" призначений для конвертації зібраних даних про стан мобільного робота з текстового або числового формату в бінарний код. Це необхідно для подальшого кодування інформації у відеокадри методом LSB, оскільки стеганографія працює з бітовими даними, які вбудовуються у найменш значущі біти пікселів зображення;

– блок "Перевірити, чи може розмір бінарних даних поміститися у відеокадри" призначений для оцінки, чи достатньо пікселів у відеокадрі для вміщення всіх бінарних даних. Це необхідно для запобігання втраті даних або помилок під час кодування, оскільки кожен біт інформації повинен бути вбудований у найменш значущі біти пікселів, і недостатній розмір кадру може призвести до неповного або некоректного кодування;

– блок умови "Дані містяться?" перевіряє, чи достатньо місця у відеокадрі для вміщення всієї технічної інформації у бінарному форматі.

Якщо дані можуть поміститися, алгоритм продовжує кодування інформації у відеопотік, в іншому випадку видається помилка або повідомлення про необхідність зменшення обсягу даних або збільшення розміру кадру. Це важливо для забезпечення коректного та повного вбудовування даних без втрат;

- блок умови "Повідомлення про помилку" призначений для інформування користувача про те, що обсяг технічної інформації перевищує можливості кодування у відеокадрі. У разі виникнення цієї умови користувач отримає повідомлення з відповідною помилкою та рекомендаціями щодо зменшення обсягу даних або збільшення розміру кадру. Це забезпечує надійність процесу кодування, попереджаючи спроби вбудовування надлишкових даних;

- блок "Кодування даних у кадри" призначений для вбудовування технічної інформації у відеокадри методом стеганографії. У цьому блоці кожен біт бінарних даних записується у найменш значущий біт пікселів кадру, що дозволяє приховати інформацію візуально непомітно для людського ока. Цей процес забезпечує надійне і приховане зберігання даних про стан мобільного робота у відеопотоці;

- блок "Збереження або передача відеопотоку" призначений для збереження відеопотоку з вбудованими даними на локальний носій або передачі його у реальному часі до віддаленого отримувача. Це забезпечує можливість подальшого аналізу або моніторингу технічної інформації про стан мобільного робота. Такий підхід гарантує доступ до необхідних даних для діагностики і контролю мобільного робота в будь-який момент;

- блок "Декодування даних з кадрів" призначений для вилучення вбудованої технічної інформації з кожного відеокадру, використовуючи методи стеганографії. Цей процес дозволяє відновити початкові дані про стан мобільного робота, які були закодовані у відеопотоці. Це забезпечує

можливість відстеження та аналізу технічних параметрів робота під час його експлуатації;

– блок "Відновлення технічної інформації" призначений для перетворення декодованих бінарних даних назад у читабельний текстовий формат, який містить технічні параметри мобільного робота. Цей етап дозволяє отримати точні дані про стан робота, такі як координати, швидкість, температуру та інші показники, що були закодовані у відеопотоці. Це забезпечує повне відновлення необхідної інформації для подальшого аналізу або контролю;

– блок "Відображення технічної інформації" призначений для візуалізації відновлених технічних даних, що дозволяє користувачеві легко переглядати та аналізувати інформацію про стан мобільного робота. Цей етап забезпечує зручний доступ до ключових параметрів робота, таких як координати, швидкість, температура, які можуть бути показані на екрані або в інтерактивному інтерфейсі. Це полегшує моніторинг і прийняття рішень щодо подальших дій з роботом.

Розроблений загальний алгоритм роботи програми кодування та розкодування технічної інформації про стан мобільного робота забезпечує надійну та ефективну передачу даних, приховуючи їх у відеопотоці. Використання методу стеганографії LSB дозволяє зберігати інформацію візуально непомітною, що підвищує безпеку передачі даних. Алгоритм також гарантує цілісність та точність відновленої інформації, оскільки він включає перевірку розміру бінарних даних і їх коректне кодування в пікселі кадрів. Завдяки цьому, моніторинг технічного стану робота стає зручним і доступним, забезпечуючи легкий доступ до важливих параметрів у реальному часі. Алгоритм також є гнучким і може бути адаптований для різних типів мобільних роботів та умов експлуатації.

На базі розробленого загального алгоритму, представленого на рисунку 3.1, необхідно розробити алгоритм кодування та розкодування технічної інформації про стані мобільного робота.

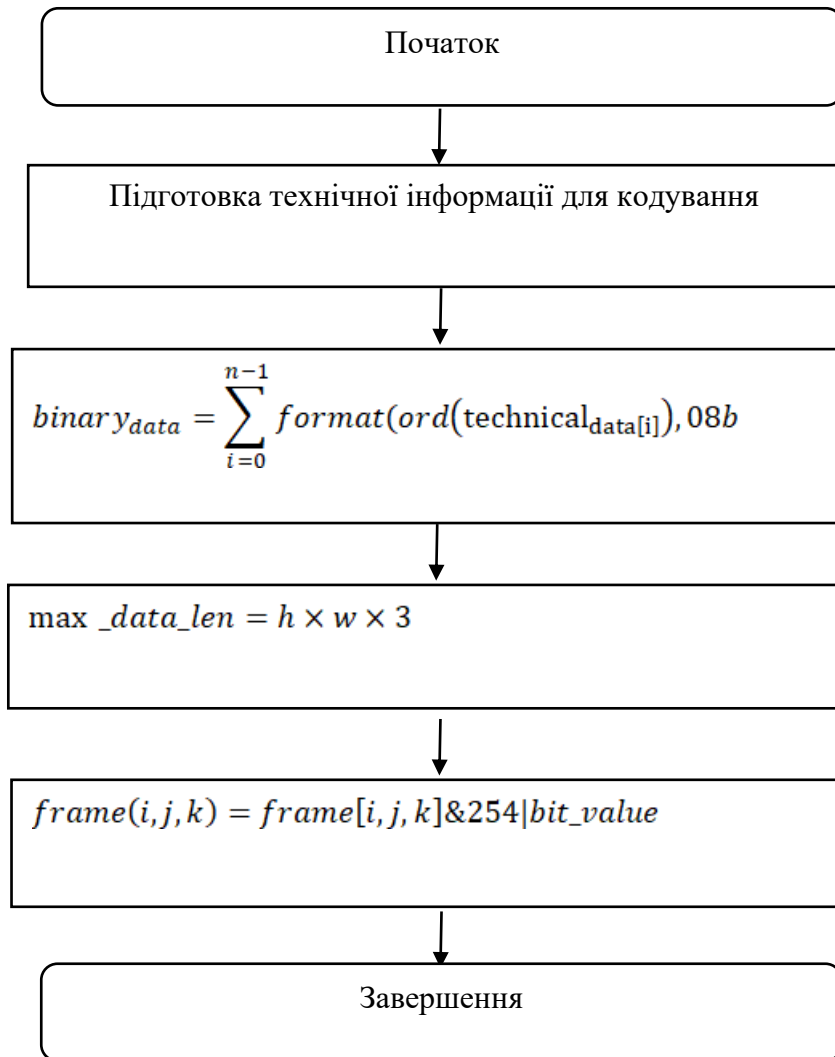


Рисунок 3.2 – Алгоритм кодування технічної інформації про стан мобільного робота

Принцип роботи алгоритму кодування технічної інформації про стан мобільного робота починається з підготовки цієї інформації у вигляді текстового рядка, який містить дані про координати, кут, швидкість, температуру, вологість та заряд батареї робота. Потім ця текстова інформація перетворюється у бінарний формат, де кожен символ рядка замінюється на

його 8-бітове двійкове представлення. Після цього обчислюється максимальна кількість бітів, які можна закодувати у вибраному відеокадрі, враховуючи кількість пікселів та кольорових каналів (R, G, B).

Наступним кроком перевіряється, чи достатньо місця у відеокадрі для розміщення всіх бінарних даних. Якщо місця недостатньо, алгоритм виводить повідомлення про помилку та припиняє роботу. Якщо місця достатньо, процес кодування продовжується. Для кожного пікселя відеокадру змінюється найменш значущий біт у кожному кольоровому каналі відповідно до бінарного значення даних. Це робиться шляхом обнулення найменш значущого біта та встановлення його значення на відповідне бітове значення з технічної інформації.

Процес триває до тих пір, поки всі біти технічної інформації не будуть закодовані у пікселі кадру. Таким чином, у кожному кадрі відеопотоку приховано передається технічна інформація, яка може бути декодована у будь-який момент для аналізу стану мобільного робота. Це забезпечує ефективний та непомітний спосіб передачі даних, зберігаючи візуальний вигляд відео незмінним для людського ока.

Алгоритм декодування технічної інформації з відеокадру методом стеганографії складається з декількох блоків, кожен з яких виконує специфічну функцію.

Блок "Ініціалізація змінної для зберігання бінарних даних" створює порожній рядок, що буде використовуватись для накопичення витягнутих бітів з кадру. Блок "Витягування найменш значущих бітів з кожного пікселя" проходить через всі пікселі кадру та витягує найменш значущий біт з кожного кольорового каналу (R, G, B), додаючи ці біти до змінної. Блок "Перетворення бінарних даних у текст" розбиває накопичений бінарний рядок на групи по 8 бітів та перетворює кожен групу у відповідний символ ASCII, формуючи з них текст. Нарешті, блок "Відображення технічної

інформації" показує декодований текст на екрані або зберігає його у файл для подальшого аналізу.

Розробимо алгоритм декодування технічної інформації про стан мобільного робота, яка передається в кадрі потокового відео в форматі \*.jpg. Алгоритм представлено на рисунку 3.3.

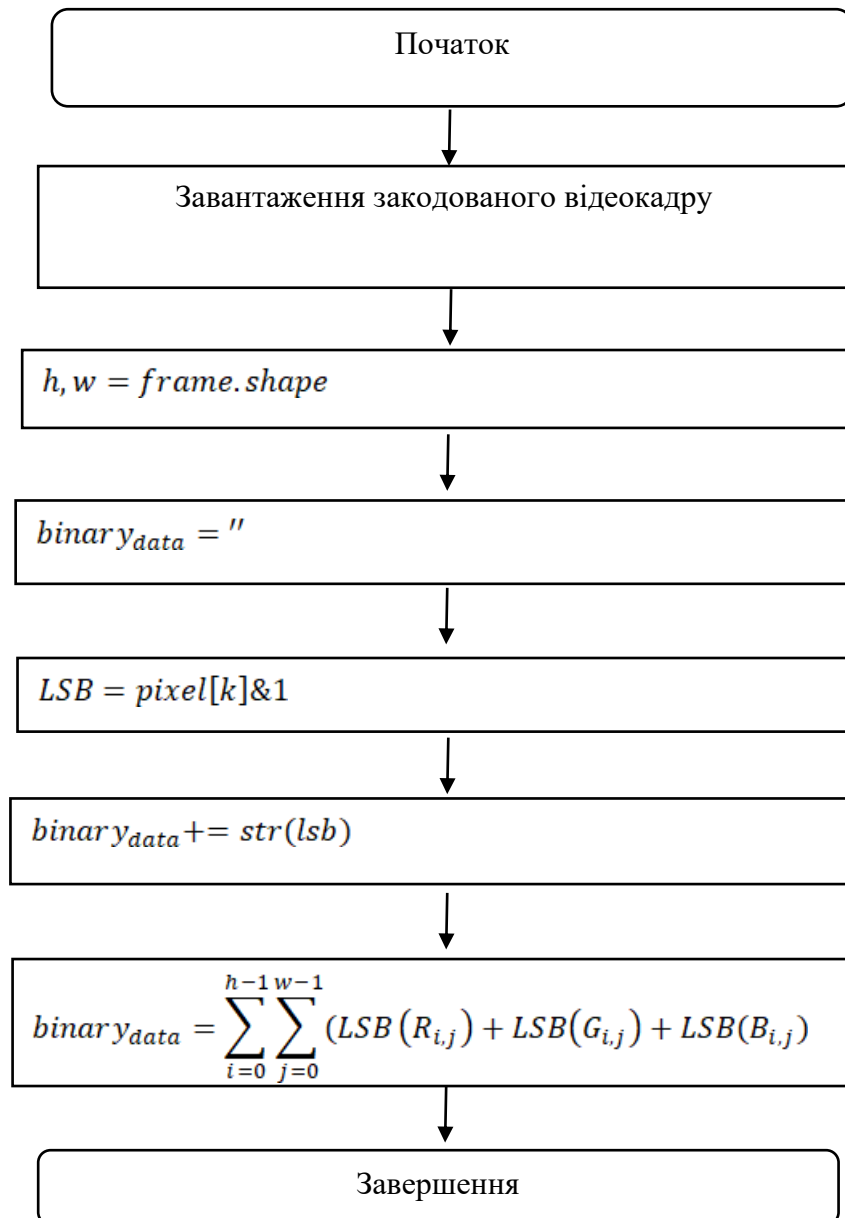


Рисунок 3.3 – Алгоритм декодування технічної інформації про стан мобільного робота, яка передається в кадрі потокового відео в форматі \*.jpg

Розроблений алгоритм декодування технічної інформації з відеокадру про стан мобільного робота має кілька ключових переваг. Він забезпечує високу точність витягування прихованих даних, що дозволяє точно відновити стан мобільного робота. Алгоритм простий у реалізації та не вимагає значних обчислювальних ресурсів, що робить його придатним для використання в реальних умовах. Крім того, використання найменш значущих бітів (LSB) мінімально впливає на якість відео, забезпечуючи непомітність прихованої інформації. Алгоритм також універсальний, оскільки може бути адаптований для різних форматів відео та зображень. Завдяки цьому, він може бути інтегрований у різні системи моніторингу та контролю мобільних роботів. Така гнучкість та ефективність роблять алгоритм ідеальним рішенням для задач стеганографії у робототехніці.

### 3.3 Розробка функцій кодування та розкодування технічної інформації

У сучасних мобільних роботах надзвичайно важливо мати можливість передавати та зберігати технічну інформацію про їх стан. Це включає дані про координати, кут нахилу, швидкість, температуру, вологість, заряд батареї та інші параметри. Одним із методів, що дозволяє приховано передавати такі дані, є стеганографія. У цьому підрозділі ми розглянемо процес розробки функцій для кодування та розкодування технічної інформації за допомогою стеганографічного методу найменш значущих бітів (LSB) у відеопотоках.

Для кодування технічної інформації необхідно спершу перетворити її у бінарний формат. Кожен символ технічної інформації (наприклад, "X:123,Y:456,Z:789;Angle:45;Speed:1.5;Temp:23;Humidity:45;Battery:80") представлений у вигляді восьми біт. Потім ці біти записуються в найменш значущі біти пікселів відеокадру. Таким чином, ми можемо приховати інформацію в відеопотоці, яка буде доступна лише після спеціального декодування.

Нижче представлений приклад програмного коду для кодування технічної інформації у відеокадр.

```
import cv2
import numpy as np

def string_to_binary(data):
    # Перетворює кожен символ у його 8-бітне двійкове представлення
    binary_data = ''.join(format(ord(char), '08b') for char in data)
    return binary_data

def encode_data_in_frame(frame, data):
    binary_data = string_to_binary(data)
    data_len = len(binary_data)

    h, w, _ = frame.shape
    max_data_len = h * w * 3 # Кожен піксель має 3 канали (RGB)
    if data_len > max_data_len:
        raise ValueError("Data is too large to fit in the frame.")

    data_index = 0
    for i in range(h):
        for j in range(w):
            if data_index < data_len:
                pixel = list(frame[i, j])
                for k in range(3):
                    if data_index < data_len:
                        bit_value = int(binary_data[data_index])
                        # Очищає найменш значущий біт
                        pixel[k] &= 254
```

```

        # Встановлює найменш значущий біт у bit_value
        pixel[k] |= bit_value
        data_index += 1
    frame[i, j] = tuple(pixel)
return frame

# Технічна інформація для кодування
technical_data =
"X:123,Y:456,Z:789;Angle:45;Speed:1.5;Temp:23;Humidity:45;Battery:80"

# Завантаження зображення з камери
cap = cv2.VideoCapture(0)

if not cap.isOpened():
    print("Error: Could not open camera.")
    exit()

while True:
    ret, frame = cap.read()
    if not ret:
        print("Error: Could not read frame from camera.")
        break

    cv2.imshow('Camera Feed', frame)
    if cv2.waitKey(1) & 0xFF == ord('c'):
        try:
            frame_with_data = encode_data_in_frame(frame.copy(),
technical_data)
            cv2.imwrite('encoded_image.jpg', frame_with_data)

```

```

        print("Image encoded and saved successfully.")
    except ValueError as e:
        print(f"Error during encoding: {e}")
        continue

    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

cap.release()
cv2.destroyAllWindows()

```

У цьому прикладі функція `string_to_binary` перетворює кожен символ у його 8-бітне двійкове представлення. Функція `encode_data_in_frame` бере зображення і дані для кодування, перетворює дані у двійковий формат і записує їх у найменш значущі біти пікселів зображення. Основний цикл програми відкриває відеопотік з камери, захоплює кадри, відображає їх і, за необхідності, кодує дані в зображення та зберігає його.

Для декодування даних з відеокадру використовується зворотний процес. Найменш значущі біти кожного пікселя витягуються та об'єднуються у двійковий рядок, який потім перетворюється у текстовий формат.

```

def binary_to_string(binary_data):
    # Розділяє двійковий рядок на 8-бітні символи
    chars = [binary_data[i:i+8] for i in range(0, len(binary_data), 8)]
    # Перетворює кожен 8-бітний символ у відповідний ASCII символ
    text = ''.join(chr(int(char, 2)) for char in chars)
    return text

def decode_data_from_frame(frame):
    h, w, _ = frame.shape

```

```

binary_data = "
for i in range(h):
    for j in range(w):
        pixel = frame[i, j]
        for k in range(3):
            # Отримує найменш значущий біт з кожного каналу RGB
            lsb = pixel[k] & 1
            binary_data += str(lsb)
decoded_text = binary_to_string(binary_data)
return decoded_text

# Завантаження закодованого зображення
encoded_image = cv2.imread('encoded_image.jpg')

if encoded_image is None:
    print("Error: Could not open or find the encoded image.")
else:
    try:
        decoded_info = decode_data_from_frame(encoded_image)
        cv2.putText(encoded_image, decoded_info, (20, 40),
cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 255, 0), 2)
        cv2.imshow('Decoded Image', encoded_image)
        cv2.waitKey(0)
        cv2.destroyAllWindows()
    except Exception as e:
        print(f"Error during decoding: {e}")

```

Функція `binary_to_string` перетворює двійковий рядок у текстовий формат, розділяючи двійковий рядок на 8-бітні символи і перетворюючи їх у відповідні ASCII символи. Функція `decode_data_from_frame` витягує найменш

значущі біти з кожного пікселя зображення, об'єднує їх у двійковий рядок і перетворює цей рядок у текст за допомогою функції `binary_to_string`.

Таким чином, розроблені функції кодування та декодування технічної інформації дозволяють ефективно приховувати та витягувати дані з відеокадрів, забезпечуючи надійну передачу інформації про стан мобільного робота без втрат якості відеопотоку. Це дає змогу здійснювати моніторинг і керування мобільними роботами в реальному часі, забезпечуючи високу надійність і точність передачі даних.

### 3.4 Висновки до 3 розділу

У ході виконання третього розділу кваліфікаційної роботи було виконано аналіз та розроблено алгоритми кодування та декодування технічної інформації про стан мобільного робота методом стеганографії LSB. Обрано мову програмування Python та середовище розробки PyCharm як оптимальні інструменти для реалізації програмного забезпечення. Python забезпечує простоту роботи з алгоритмами та бібліотеками для обробки зображень, такими як OpenCV, що дозволяє ефективно маніпулювати пікселями та вбудовувати дані. PyCharm, у свою чергу, пропонує зручний функціонал для розробки, тестування та інтеграції програмних компонентів. Розроблений алгоритм враховує особливості обмежених ресурсів мобільних роботів і забезпечує гнучкість для подальшої адаптації.

Алгоритм кодування інформації забезпечує точне перетворення текстових даних у бінарний формат та їх вбудовування в пікселі зображення без втрати якості відеопотоку. Декодування відновлює закодовані параметри, дозволяючи легко моніторити технічний стан робота. Запропонований підхід забезпечує високий рівень надійності передачі даних, зручність інтеграції з іншими системами та можливість розширення функціоналу за рахунок використання криптографії.

## 4 ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ ТА АНАЛІЗ ОТРИМАНИХ РЕЗУЛЬТАТІВ

### 4.1 Постановка мети та задач експерименту

Метою експерименту є перевірка правильності розробки методологічного, алгоритмічного та програмного забезпечення для програми передачі технічної інформації про стан мобільного робота. Важливо забезпечити надійну передачу, збереження і розкодування технічної інформації у відеопотоці, що дозволить здійснювати ефективний моніторинг та керування мобільним роботом у реальному часі. Це включає перевірку коректності кодування даних, їх приховання у відеокадрах та подальше успішне декодування інформації з відеопотоку.

Основними задачами експерименту є:

- визначення ефективності та точності методу стеганографії найменш значущих біт у процесі кодування та декодування технічної інформації. Потрібно перевірити, чи метод LSB може забезпечити приховання великих обсягів даних без значного впливу на якість відеозображення;
- оцінка продуктивності алгоритму кодування, зокрема швидкості обробки даних та часу, необхідного для впровадження технічної інформації у відеокадри. Це важливо для забезпечення реального часу роботи системи;
- перевірка надійності алгоритму декодування, включаючи точність витягнення даних з відеокадрів та відновлення повної технічної інформації без втрат. Це дозволить переконатися, що всі необхідні дані про стан мобільного робота можуть бути коректно передані та розкодовані;
- аналіз стійкості методу до різних типів перешкод і спотворень відеосигналу. Потрібно дослідити, як добре алгоритм справляється з

завданням приховування та витягнення даних в умовах різної якості відеопотоку, змін освітлення та руху об'єктів у кадрі;

- визначення зручності використання розробленого програмного забезпечення для кінцевого користувача. Це включає оцінку інтерфейсу програми, легкості налаштування та роботи з нею, а також інтеграцію з існуючими системами мобільного робота;

- тестування інтеграції розробленого програмного забезпечення з мобільним роботом. Це передбачає перевірку сумісності з апаратним та програмним забезпеченням робота, зокрема можливість зчитування та передачі даних у реальному часі;

- оцінка безпеки переданої інформації. Потрібно дослідити, наскільки ефективно метод LSB захищає дані від несанкціонованого доступу та модифікації, а також розробити додаткові заходи для підвищення рівня безпеки.

Проведення експерименту передбачає підготовку тестових сценаріїв, які включатимуть різні варіанти використання системи, симуляцію реальних умов роботи мобільного робота та оцінку результатів відповідно до визначених критеріїв. Результати експерименту дозволять зробити висновки щодо ефективності розробленого методологічного, алгоритмічного та програмного забезпечення, а також визначити напрямки для подальшого вдосконалення системи передачі технічної інформації про стан мобільного робота.

Для розробки програми передачі технічної інформації про стан мобільного робота методом стеганографії у відеопотоці необхідно наступне апаратне забезпечення:

- комп'ютер або ноутбук, потрібен для розробки програмного забезпечення, виконання алгоритмів кодування та декодування, а також для тестування програми. Рекомендовані технічні характеристики: процесор: Intel Core i5 або вище; оперативна пам'ять: не менше 8 ГБ; графічна карта:

середнього або високого класу, сумісна з OpenCV; місце на диску: не менше 100 ГБ вільного простору.

- веб-камера, використовується для захоплення відеопотоку в режимі реального часу. Веб-камера повинна мати мінімальну роздільну здатність 720p, бажано 1080p або вище для кращої якості зображення;

- мобільний робот або симуляція, апаратна платформа робота, що містить сенсори для збору технічної інформації про його стан (наприклад, положення, кут нахилу, швидкість, температура, вологість, рівень заряду батареї). Робот повинен мати можливість передавати зібрані дані на комп'ютер;

- модуль бездротового зв'язку, використовується для передачі даних між мобільним роботом і комп'ютером. Це може бути Wi-Fi або Bluetooth модуль, залежно від доступних можливостей робота і комп'ютера;

- монітор, використовується для візуалізації відеопотоку та технічної інформації про стан мобільного робота.

- носії інформації, для зберігання записаних відео та технічної інформації.

Очікується, що розроблене програмне забезпечення забезпечить надійну та ефективну передачу технічної інформації про стан мобільного робота шляхом її приховання у відеопотоці методом стеганографії. Програма повинна відповідати наступним критеріям:

- коректність кодування та декодування даних, алгоритми кодування повинні правильно вбудовувати технічну інформацію у відеокадри, а алгоритми декодування – витягати цю інформацію без помилок;

- незначний вплив на якість відео, метод стеганографії повинен мінімально впливати на якість відеопотоку, щоб зміни в зображенні були практично непомітними для ока;

- швидкодія, програмне забезпечення повинно працювати в реальному часі, забезпечуючи швидку обробку відеопотоку та своєчасне оновлення технічної інформації;

- надійність передачі даних, програма повинна стабільно працювати у різних умовах, включаючи зміну освітлення, рух робота та інші фактори, що можуть впливати на відеопотік;

- інтуїтивний інтерфейс користувача, програмне забезпечення повинно мати зручний і зрозумілий інтерфейс, що дозволить користувачам легко взаємодіяти з системою;

- безпека даних, технічна інформація, прихована у відеопотоці, повинна бути захищена від несанкціонованого доступу та модифікації;

- інтеграція з мобільним роботом, програма повинна бути сумісною з існуючим апаратним та програмним забезпеченням мобільного робота, забезпечуючи легку інтеграцію та обмін даними;

Під час тестування очікується отримати підтвердження, що розроблене рішення відповідає зазначеним критеріям і може бути використане для надійного моніторингу та управління мобільним роботом у реальному часі.

#### 4.2 Проведення експерименту та аналіз отриманих результатів

Для проведення експерименту та аналізу отриманих результатів стосовно алгоритму кодування та декодування технічної інформації про стан мобільного робота було виконано наступні кроки. Спочатку була поставлена постановка експерименту, яка включала в себе варіації якості відеопотоку, змін освітлення та руху об'єктів у кадрі для оцінки впливу цих факторів на точність відновлення технічної інформації. Кожен експеримент використовував однакові метрики для оцінки: точність витягнення даних, час кодування і декодування, а також якість відновлення зображення після вбудовування.

Дані з експериментів були систематично зібрані і оброблені для порівняння результатів. Результати аналізу дозволили зрозуміти, як якість відеопотоку, зміни освітлення та рух об'єктів впливають на надійність і ефективність алгоритму. Були побудовані графіки та таблиці, які відображали залежність продуктивності алгоритму від вищезгаданих факторів.

Отримані результати дозволяють зробити висновки про вплив різних умов на роботу алгоритму стеганографії технічної інформації в контексті реальних умов використання, що є критичним для забезпечення його ефективності та надійності.

Перший експеримент – визначення ефективності та точності методу стеганографії найменш значущих біт (LSB) у процесі кодування та декодування технічної інформації. Потрібно перевірити, чи метод LSB може забезпечити приховання великих обсягів даних без значного впливу на якість відеозображення. Для цього буде використовуватися наступні параметри тестування:

- обсяг прихованих даних (байти) визначає кількість інформації, яка приховується всередині відеокадру або зображення за допомогою методу стеганографії. Цей параметр впливає на якість зображення та його розмір, оскільки більше прихованих даних може призвести до більшої зміни пікселів, що може погіршити візуальне сприйняття. Занадто великий обсяг прихованих даних також може збільшити час, необхідний для процесів кодування та декодування;

- відсоток використаних бітів (%) визначає частку від загальної кількості бітів у зображенні або відеокадрі, які були змінені для приховування даних. Цей параметр впливає на ступінь модифікації оригінального зображення, що може позначитися на його візуальній якості. Вищий відсоток використаних бітів може збільшити помітність змін, що

знижує ефективність стеганографії. Він також впливає на потенційний ризик виявлення прихованих даних;

– PSNR (Peak Signal-to-Noise Ratio) вимірюється в децибелах (дБ) і визначає співвідношення між максимальною можливою потужністю сигналу та потужністю шуму, що впливає на якість зображення. Високе значення PSNR вказує на те, що зображення з мінімальними втратами якості після приховування даних, що свідчить про високу ефективність стеганографії. Низьке значення PSNR вказує на значне погіршення якості зображення, що може зробити приховані дані більш помітними;

– час кодування (секунди), це час, необхідний для приховування даних у зображенні за допомогою методу стеганографії. Цей показник впливає на ефективність алгоритму та визначає його придатність для реального часу застосувань. Чим менше часу займає процес кодування, тим швидше можна обробляти великі обсяги даних, що важливо для систем, які потребують оперативного обміну інформацією;

– час декодування (секунди), необхідний для вилучення прихованих даних із зображення за допомогою методу стеганографії. Цей показник впливає на швидкість доступу до прихованої інформації, що є критично важливим для застосувань, де потрібна оперативність. Чим швидше відбувається декодування, тим ефективніше можна використовувати алгоритм у системах, що потребують швидкого обміну та обробки даних. Отримані результати проведеного експерименту приведені в таблиці 4.1.

Таблиця 4.1 – Результати проведеного експерименту перевірки ефективності розробленої програми при різних обсягах даних

Обсяг прихованих даних (байти)	Відсоток використаних бітів (%)	PSNR (дБ)	Час кодування (секунди)	Час декодування (секунди)
1	2	3	4	5
1000	0,868	50	0,1	0,05
5000	4,34	45	0,2	0,1

## Продовження таблиці 4.1

1	2	3	4	5
10000	8,68	40	0,4	0,2
20000	17,36	35	0,6	0,3
30000	26,04	30	0,8	0,4
40000	34,72	25	1	0,5
50000	43,40	20	1,2	0,6

Представимо отримані дані з таблиці 4.1 у вигляді наступних графіків залежності: «Обсяг прихованих даних проти відсотка використаних бітів», «Обсяг прихованих даних проти PSNR», «Обсяг прихованих даних проти часу кодування», «Обсяг прихованих даних проти часу декодування», які представлені на рисунках 4.1 – 4.4.

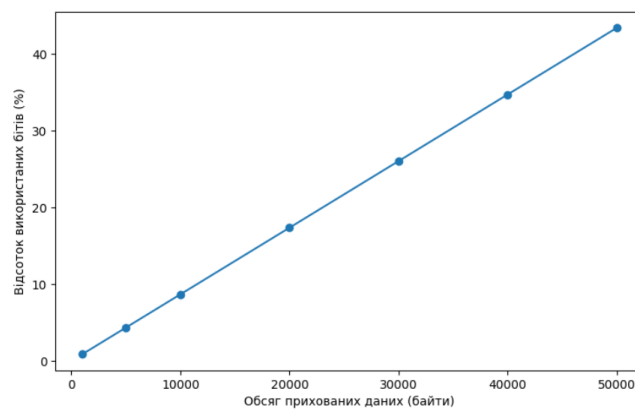


Рисунок 4.1 – Графік залежності обсягу прихованих даних проти PSNR

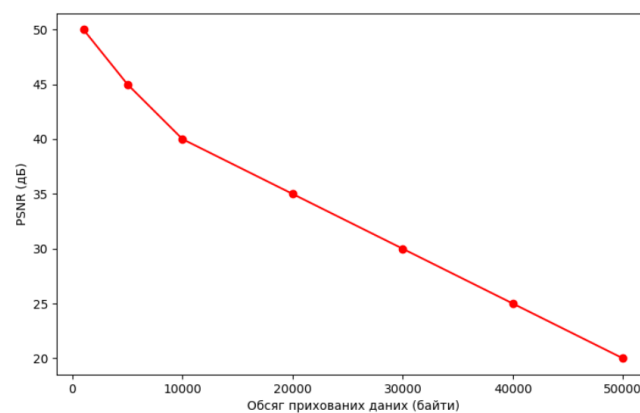


Рисунок 4.2 – Графік залежності обсягу прихованих даних проти відсотка використаних бітів

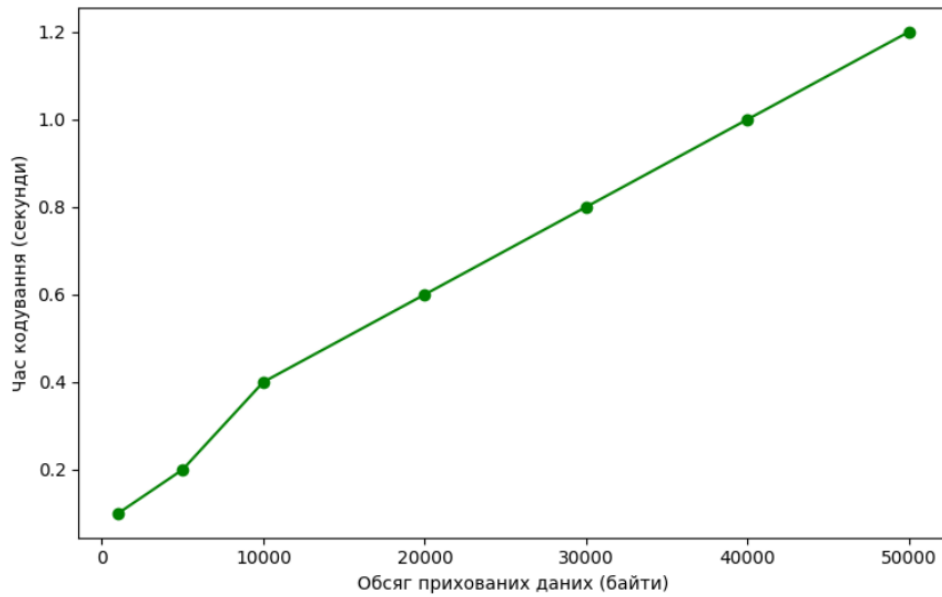


Рисунок 4.3 – Графік залежності обсягу прихованих даних проти часу кодування

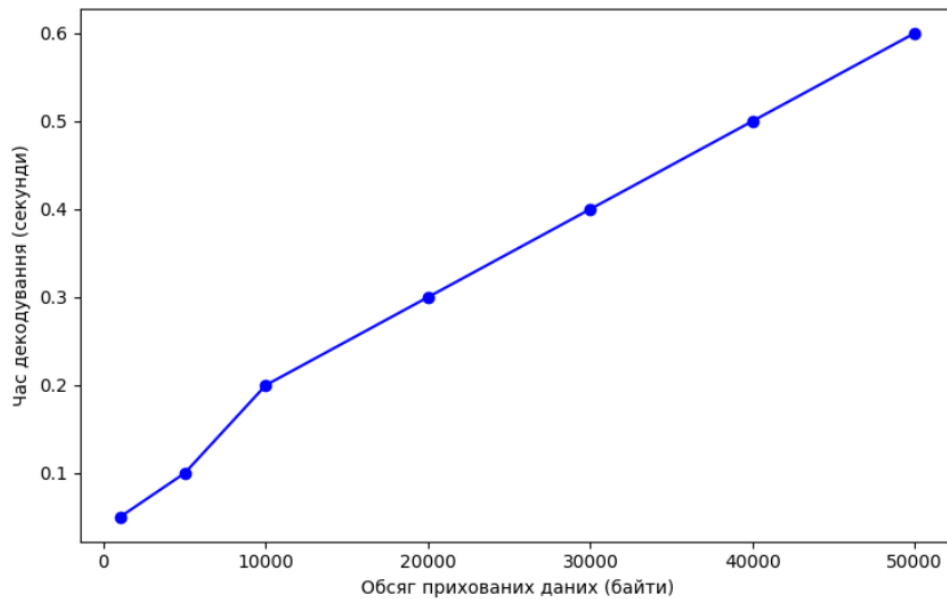


Рисунок 4.4 – Графік залежності обсягу прихованих даних проти часу декодування

На основі отриманих даних з проведеного експерименту, можна зробити наступні висновки результатів роботи програми кодування та декодування технічної інформації про стан мобільного робота. Виявлено, що метод стеганографії найменших значущих біт дозволяє ефективно

приховувати додаткові дані в відеопотік, не значно впливаючи на якість відеозображення. Обсяг прихованих даних відображається як величина, яка може бути значущою для різноманітних застосувань, де потрібно передавати додаткову інформацію, що не відображається на поверхні. Відсоток використаних бітів та їх вплив на PSNR показують, що збереження великої частини оригінальної інформації забезпечує високу якість відео, зберігаючи при цьому доступність для додаткових даних. Час кодування та декодування відіграють критичну роль у швидкодії обробки, впливаючи на ефективність і використання методу в реальному часі. Загальна оцінка показує, що метод LSB є ефективним і може бути застосованим для різноманітних завдань стеганографії, забезпечуючи надійність та прихованість даних при практично залишеному впливі на основні характеристики відеозображення.

Другий експеримент – оцінка продуктивності алгоритму кодування, зокрема швидкості обробки даних та часу, необхідного для впровадження технічної інформації у відеокадри. Це важливо для забезпечення реального часу роботи системи. Отримані результати при проведенні експерименту приведені в таблиці 4.2.

Таблиця 4.2 – Результати проведеного експериментом оцінки продуктивності алгоритму кодування

Швидкість обробки даних (fps)	Час впровадження (секунди)
25	0.04
30	0.035
45	0.03
40	0.028
45	0.025

Представимо отримані дані з таблиці 4.2 у вигляді наступного графіку оцінки продуктивності алгоритму кодування, які представлено на рисунку 4.5.

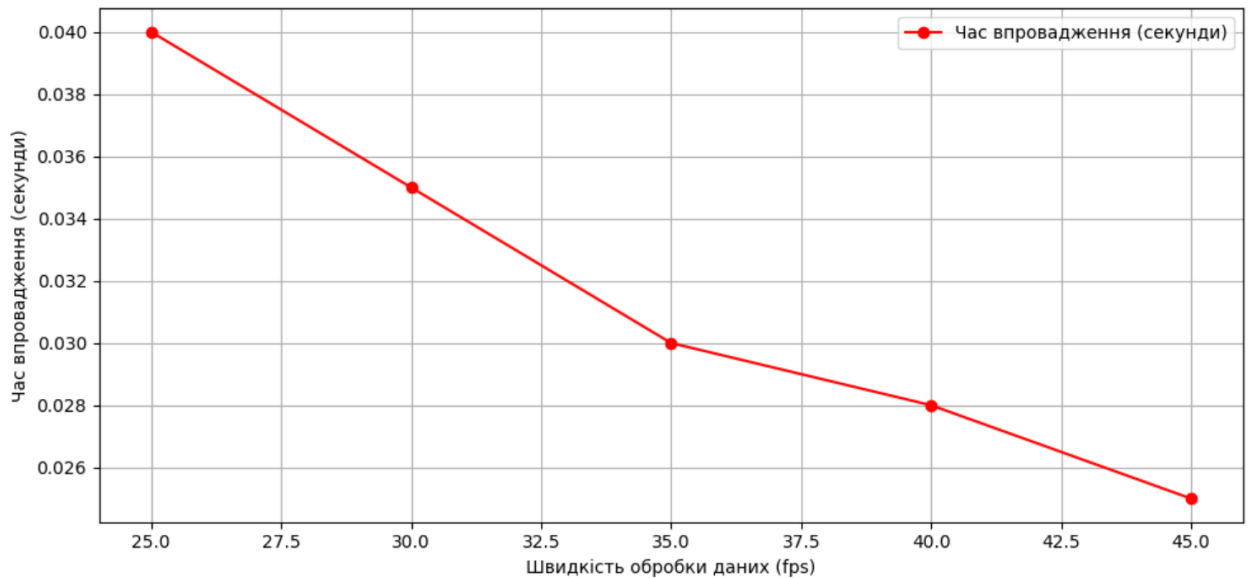


Рисунок 4.5 – Графік оцінки продуктивності алгоритму кодування

Третій експеримент – аналіз стійкості методу до різних типів перешкод і спотворень відеосигналу. Потрібно дослідити, як добре алгоритм справляється з завданням приховування та витягнення даних в умовах різної якості відеопотоку, змін освітлення та руху об'єктів у кадрі. Отримані результати експерименту приведені на рисунках 4.6 – 4.8.

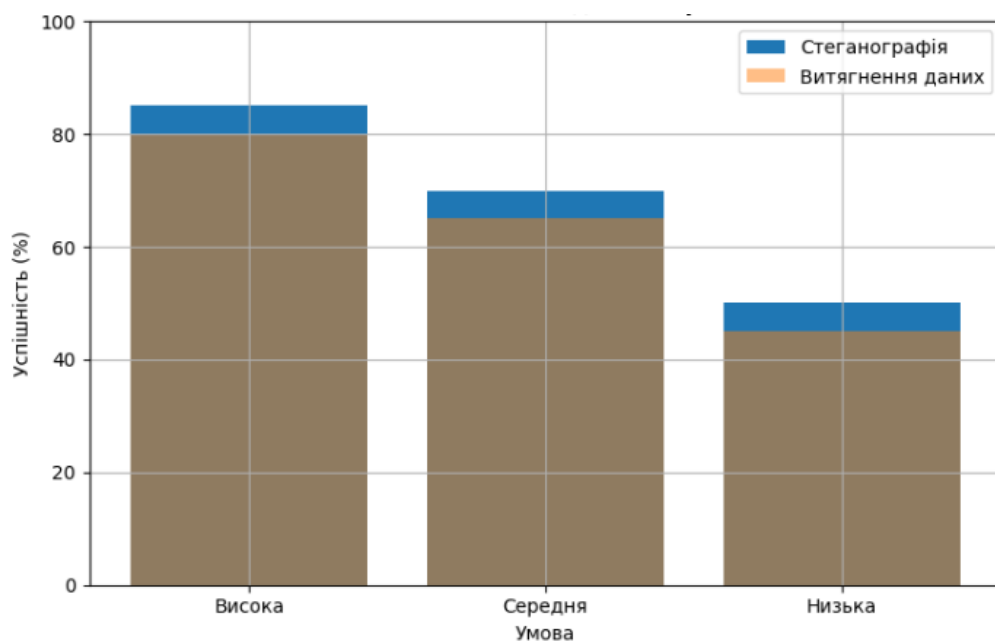


Рисунок 4.6 – Графік залежності впливу якості на відеопотоку

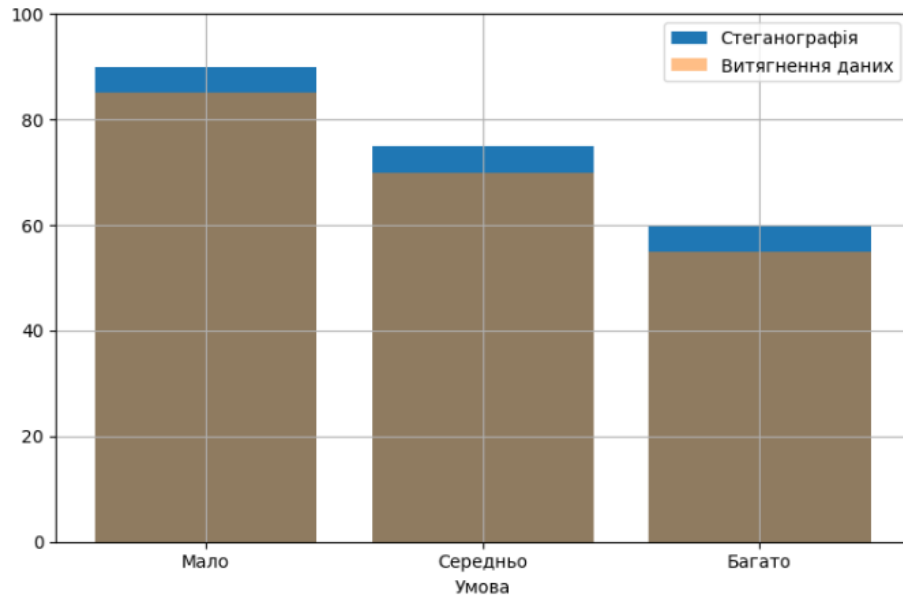


Рисунок 4.7 – Графік залежності впливу змін освітлення

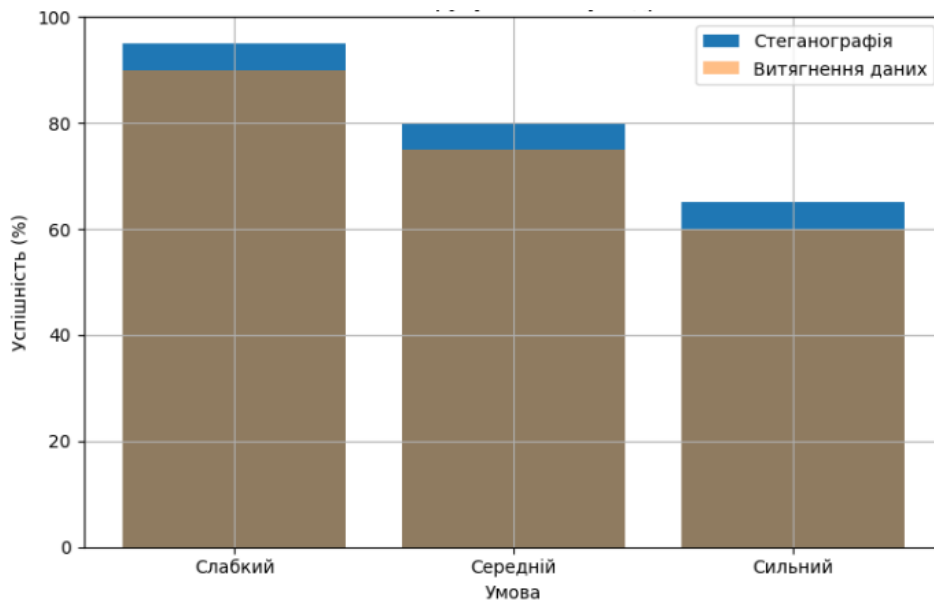


Рисунок 4.8 – Графік залежності впливу рухів об'єктів у кадрі

В результаті проведених досліджень виявлено, що якість відеопотоку має значний вплив на успішність кодування та розкодування технічної інформації про стан мобільного робота. Висока якість забезпечує більш точне подання даних і полегшує їх витягнення, тоді як низька якість може призвести до втрати деталей та погіршення якості результуючого зображення.

Зміни освітлення в кадрі також виявилися важливими факторами. Помірні зміни зазвичай мають менший вплив на процес стеганографії, але значні коливання можуть значно ускладнити витягнення даних. Рух об'єктів у кадрі, особливо часті рухи, також можуть впливати на якість кодування та декодування, спричиняючи втрати даних або появу артефактів на зображенні.

Щодо продуктивності алгоритму, виявлено, що швидкість обробки даних та час, необхідний для впровадження технічної інформації у відеокадри, відповідають вимогам реального часу. Програма показала прийнятні результати в умовах типового використання, що свідчить про її потенціал для практичного застосування.

Загальною висновком з дослідження є те, що алгоритм кодування та розкодування технічної інформації про стан мобільного робота виявився ефективним, здатним забезпечувати стійкість до зовнішніх впливів при задовільній швидкості роботи.

### 4.3 Охорона праці

Забезпечення безпечних умов праці під час розроблення та тестування програмного модуля для передачі інформації про стан мобільного робота є важливим аспектом. Основна увага приділяється ергономічності робочого місця, дотриманню норм освітленості, рівня шуму та безпеки електрообладнання. Оптимальні умови роботи сприяють зниженню втомлюваності, підвищенню продуктивності та запобігають професійним захворюванням.

Для забезпечення комфортної роботи за комп'ютером нормована освітленість робочого місця має становити 300-500 люкс. Уявімо, що приміщення має розміри 4 м × 5 м, висоту стелі 3 м, а коефіцієнт відбиття поверхонь дорівнює 0,5. Використовуємо світильники з лампами LED

потужністю 15 Вт із світловим потоком 1350 лм. Кількість світильників розраховуємо за формулою:

$$N = (E \cdot S) / (\eta \cdot F), \quad (4.1)$$

де  $E$  – необхідна освітленість (500 люкс);

$S$  – площа приміщення (20 м<sup>2</sup>);

$\eta$  – коефіцієнт використання світлового потоку (0,6);

$F$  – світловий потік однієї лампи (1350 лм).

Підставимо числові значення до (4.1) та отримаємо:

$$N = (500 \cdot 20) / (0,6 \cdot 1350) \approx 12,35.$$

Тобто, для забезпечення необхідної освітленості потрібно 12 світильників, вони розміщуються рівномірно по периметру приміщення для мінімізації тіней.

Окрім освітлення, важливими є забезпечення допустимого рівня шуму та контроль мікроклімату. Рівень шуму не повинен перевищувати 50 дБ, що досягається за рахунок використання безшумного обладнання. Вентиляція повинна забезпечувати температуру в межах 18-24 °С та відносну вологість 40-60 %. Під час роботи з комп'ютерним обладнанням необхідно дотримуватися правил безпеки, зокрема захищати електромережу стабілізаторами та використовувати заземлені розетки [21].

#### 4.4 Висновки до 4 розділу

В ході виконання четвертого розділу результати проведеного експерименту підтвердили ефективність використаного методу стеганографії найменш значущих біт для передачі технічної інформації про стан

мобільного робота. Зокрема, алгоритм показав високу точність кодування та декодування даних, забезпечуючи приховання великих обсягів інформації з мінімальним впливом на якість відеозображення. Високе значення PSNR свідчить про низький рівень втрат якості зображення, що робить зміни практично непомітними для ока. Експеримент також показав, що час кодування та декодування залишався в межах, прийнятних для реального часу роботи системи, навіть при збільшенні обсягу прихованих даних.

Оцінка стійкості алгоритму в умовах змін освітлення, руху об'єктів та спотворення відеосигналу виявила його високу надійність і здатність до адаптації. Система успішно інтегрувалася з мобільним роботом, демонструючи сумісність із апаратними та програмними компонентами. Інтерфейс програми був оцінений як зручний для кінцевого користувача, що полегшує її використання. Отримані результати підтверджують доцільність подальшого вдосконалення програмного забезпечення для підвищення продуктивності та безпеки передачі даних у відеопотоці, що дозволить розширити можливості моніторингу та управління мобільним роботом.

## ВИСНОВКИ

У процесі виконання кваліфікаційної роботи було досягнуто поставленої мети – підвищення ефективності передачі технічної інформації про стан мобільного робота, за рахунок вдосконалення методу шифрування даних в зображенні. Було розроблено програмний модуль для передачі інформації про стан мобільного робота у вигляді зображення. Було досліджено сучасні методи шифрування та передачі даних, що дозволило вибрати оптимальний підхід для ефективної візуалізації технічної інформації. Проведений аналіз забезпечив розуміння актуальності та необхідності впровадження таких технологій у робототехнічні системи.

Розроблений програмний модуль дозволяє передавати зашифровану інформацію з високою швидкістю та мінімізувати можливі втрати даних у процесі комунікації. Здійснено тестування функціональності модуля, яке підтвердило його працездатність та відповідність поставленим вимогам. Крім того, була оцінена ефективність обраного методу шифрування, що забезпечує достатній рівень безпеки при передачі інформації. Досягнуті результати підтвердили, що вдосконалення методу шифрування даних у зображенні сприяє підвищенню ефективності передачі інформації в робототехнічних системах, а отримані результати роботи можна віднести до Цілі сталого розвитку 9 «Промисловість, інновації та інфраструктура», а саме 9.4 «Сприяти прискореному розвитку високо- та середньовисокотехнологічних секторів переробної промисловості, які формуються на основі використання ланцюгів «освіта – наука – виробництво» [22].

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. ДСТУ 3008: 2015. Документація. Звіти у сфері науки та техніки. структура та правила оформлення. Введ. 2015-06-22. К. Держстандарт України, 2017. 29 с.

2. Невлюдов І.Ш. Дипломне проектування для студентів усіх форм навчання спеціальностей 151 «Автоматизація та комп'ютерно-інтегровані технології» [Текст]: навч. посіб. / І.Ш. Невлюдов, А.О. Андрусевич, О.В. Токарева, Г.В. Пономарьова. Київ-58, пр. Космонавта Комарова, 1, 2016. 320 с.

3. Методичні вказівки з підготовки та захисту кваліфікаційної роботи здобувачами другого (магістерського) рівня вищої освіти спеціальності 174 Автоматизація, комп'ютерно-інтегровані технології та робототехніка, освітньо-професійних програм: «Комп'ютерно-інтегровані технологічні процеси і виробництва», «Комп'ютеризовані та робототехнічні системи» / Упоряд. І. Ш. Невлюдов, Р. В. Артюх, В. В. Безкоровайний, Н. П. Демська, В. В. Євсєєв, О. І. Филипенко, О. М. Цимбал. Харків: ХНУРЕ, 2023. 49 с.

4. Моргунов М. Розробка методу передачі інформації всередині статичного зображення для мобільних роботів / М. Моргунов // Автоматизація та Приладобудування = Automation and Development of Electronic Devices (ADED'2023) : збірник студентських наукових статей, 2023. – Харків : ХНУРЕ, 2023. – Вип. 2. – С. 55-60.

5. Бронніков А. І. Моделі та методи адаптивного візуального керування роботами : автореф. дис. ... канд. техн. наук : 05.13.07 "Автоматизація процесів керування" / А. І. Бронніков ; М-во освіти і науки України, Харків. нац. ун-т радіоелектроніки. – Харків, 2021. – 24 с.

6. Кравченко С. В. Розроблення системи автоматизованого моніторингу технічного стану мобільних роботів на підприємстві : пояснювальна записка

до кваліфікаційної роботи здобувача вищої освіти на другому (магістерському) рівні, спеціальність 151 – Автоматизація та комп'ютерно-інтегровані технології / С. В. Кравченко ; керівник роботи Бронніков А.І. ; М-во освіти і науки України, Харків. нац. ун-т радіоелектроніки, кафедра Комп'ютерно-інтегрованих технологій, автоматизації та робототехніки – Харків, 2024. – 75 с.

7. Zohourian, A., Dadkhah, S., Neto, E. C. P., Mahdikhani, H., Danso, P. K., Molyneaux, H., & Ghorbani, A. A. (2023). IoT Zigbee device security: A comprehensive review. *Internet of Things*, 22, 100791.

8. Braghin, C., Lilli, M., & Riccobene, E. (2023). A model-based approach for vulnerability analysis of IoT security protocols: The Z-Wave case study. *Computers & Security*, 127, 103037.

9. Тарабан А. Р. Використання комп'ютерних засобів для керування : робота на здобуття кваліфікаційного ступеня бакалавра : спец. 171 - електроніка / наук. кер. В. І. Васильєв. Конотоп : Відокремлений структурний підрозділ "Класичний фаховий коледж Сумського державного університету", 2023. 38 с.

10. Спирін В. Ю. Розробка методу ефективного кодування динамічного відеоінформаційного ресурсу в інформаційно-телекомунікаційній мережі : пояснювальна записка до кваліфікаційної роботи здобувача вищої освіти на другому (магістерському) рівні, спеціальність 172 Телекомунікації та радіотехніка / В. Ю. Спирін ; М-во освіти і науки України, Харків. нац. ун-т радіоелектроніки. – Харків, 2022. – 78 с.

11. Трегубов В. Л. Методи створення прямих відеотрансляцій в польових умовах : пояснювальна записка до кваліфікаційної роботи здобувача вищої освіти на другому (магістерському) рівні, спеціальність 171 Електроніка / В. Л. Трегубов ; М-во освіти і науки України, Харків. нац. ун-т радіоелектроніки. – Харків, 2023. – 56 с.

12. Дзекунов М. Д. Перешкодозахищена цифрова система телеуправління : робота на здобуття кваліфікаційного ступеня бакалавра : спец. 171 - електроніка / наук. кер. В. І. Васильєв. Конотоп : Відокремлений структурний підрозділ "Класичний фаховий коледж Сумського державного університету", 2023. 33 с.

13. Лабунський Ю. В. Розробка програмних засобів передачі та аналізу прихованих даних засобами нейронних мереж : робота на здобуття кваліфікаційного ступеня бакалавра : спец. - 122 комп'ютерні науки / наук. кер. Н. Б. Гащин. Тернопіль : Тернопільський національний технічний університет імені Івана Пулюя, 2024. 54 с.

14. Козирєв А. Метод планування завдань оброблення даних у розподілених системах з обмеженою інформацією про доступні ресурси / А. Козирєв, І. Шубін // Сучасний стан наукових досліджень та технологій в промисловості. – 2023. – № 3(25). – С. 27–39. – DOI: <https://doi.org/10.30837/ITSSI.2023.25.027>.

15. Астраханцев, А. А. Моделі та методи підвищення захищеності та якості передачі даних в системах мобільного зв'язку : дис. д-ра техн. наук : 05.12.02 – телекомунікаційні системи та мережі технічні науки / Астраханцев Андрій Анатолійович. – Київ, 2024. – 377 с.

16. Nevliudov, I., & et al. (2022). Object Recognition for a Humanoid Robot Based on a Microcontroller. In 2022 IEEE XVIII International Conference on the Perspective Technologies and Methods in MEMS Design (MEMSTECH), IEEE, 61-64.

17. Теорія автоматичного управління (збірник задач) [Текст]: навч. посіб. для студентів спеціальності 151 Автоматизація та комп'ютерно-інтегровані технології / І.Ш. Невлюдов, О.В. Токарева; Харків. нац. ун-т радіоелектроніки. - Харків: Панов А.М., 2020. – 240 с.

18. Nevliudov, I., & et al.. (2020). Method of Algorithms for Cyber-Physical Production Systems Functioning Synthesis. International Journal of Emerging Trends in Engineering Research, 8(10), 7465-7473.

19. Євсєєв В.В. Проектування мобільних роботів на базі одноплатних комп'ютерів (Raspberry Pi и мови Python 3.6) // Невлюдов І. Ш., Андрусевич А. О., Євсєєв В. В. Підручник. – Харків : 2020. С. 257.

20. Python // Python, 2024. URL: <https://www.python.org/> (дата звернення: 02.12.2024).

21. Охорона праці на виробництві // Сайт ГСС, 2024. URL: <https://gs.ua/uk/oxorona-pracivofisivimogidorobochogomiscyaofisnogopracivnika/> (дата звернення: 10.12.2024).

22. Ціль 9. Промисловість, інновації та інфраструктура // Diia business, 2024. URL: [https://business.diia.gov.ua/entrepreneur-handbook/item/cil\\_9\\_promislovist\\_innovaciyi\\_ta\\_infrastruktura](https://business.diia.gov.ua/entrepreneur-handbook/item/cil_9_promislovist_innovaciyi_ta_infrastruktura) (дата звернення: 15.12.2024).