Хмельницький національний університет
Факультет програмування та комп'ютерних і телекомунікаційних систем
Кафедра комп'ютерних наук та інформаційних технологій

ДИПЛОМНА РОБОТА
МАГІСТРА

на тему *Інформаційна технологія маркування та перевірки автентичності медичних зображень*

Галузь знань *12 – Інформаційні технології*
Шифр і назва галузі знань

Спеціальність *122 – Комп'ютерні науки*
Шифр і назва спеціальності

Виконав: студент *2* курсу, група *КНм–19–1* _____ *В.В. Мостовий*
Підпис Ініціали, прізвище

Керівник: *к.т.н., доцент кафедри КНІТ* _____ *О.А. Пасічник*
Підпис Ініціали, прізвище

Нормоконтроль: *к.т.н., доцент кафедри КНІТ* _____ *Р.О. Багрій*
Підпис Ініціали, прізвище

До захисту допускаю:
Зав. кафедри КНІТ, д.т.н., професор _____ *О.В. Бармак*
Підпис Ініціали, прізвище
7 12 2020 р.

Хмельницький 2020

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет *програмування та комп'ютерних і телекомунікаційних систем*

Кафедра *комп'ютерних наук та інформаційних технологій*

Освітній ступінь *магістр*

Галузь знань *12 – Інформаційні технології*

Спеціальність *122 – Комп'ютерні науки*

ЗАТВЕРДЖУЮ

Завідувач кафедри *комп'ютерних наук та інформаційних технологій*

_____

(підпис)

д.т.н., професор О.В. Бармак

«7» 9 *2020* року

ЗАВДАННЯ

НА ДИПЛОМНУ РОБОТУ МАГІСТРА

1. Тема дипломної роботи магістра: *«Інформаційна технологія маркування та перевірки автентичності медичних зображень»*

2. Завдання видано студенту *Мостовому Владу Вікторовичу*

(прізвище, ім'я, по батькові)

3. Керівник роботи *д.т.н., доцент Пасічник Олександр Анатолійович*

(прізвище, ім'я, по батькові)

4. Затверджені наказом університету від «9» 9 *202* р. № 22

5. Зміст пояснювальної записки (перелік задач) та вихідні дані:

Мета роботи – реалізації інформаційної технології маркування та перевірки автентичності медичних зображень но основі ЦВЗ з покращеним рівнем захисту та програмна реалізація відповідної інформаційної технології. для дослідження практичної ефективності інформаційної технології маркування та перевірки автентичності медичних зображень. В якості методів впровадження додаткового захисту використати методи криптографії, сегментації, стеганографії. Запропонована інформаційна технологія має дозволяти маркувати вхідні зображення цифровим водяним знаком та перевіряти автентичність маркованих зображень.

Реферат

Дипломна робота магістра присвячена розробці інформаційної технології маркування та перевірки автентичності медичних зображень но основі ЦВЗ з покращеним рівнем захисту.

Актуальність теми. Потреба застосування та впровадження в медичні інформаційні системи технології маркування та перевірки автентичності медичних даних обумовлення стрімким ростом кількості відповідних суб'єктів, таких як: приватні та державні лікарні, медичні центри, лабораторії, госпіталів та їх діяльності в МІС, а саме розповсюдження медичних даних, які повинні бути захищеними для кінцевого їх отримувача, тобто автентичним .

Маркування є розповсюдженим явищем в усіх галузях людської діяльності та в більшості з них є обов'язковим та стандартизованим. Не зважаючи на те, що в недалекому минулому термін маркування вживався більше по відношенню до фізичних об'єктів, таких як: деталі, товари, продукти харчування, зараз з розвитком інформаційний технологій цей термін став актуальний для різних інформаційних систем.

Інформаційні медичні системи, здебільшого використовують застаріле програмне забезпечення, яке з кожним роком все складніше підтримувати або взагалі не підтримується розробниками на даний момент. Відповідно, застарілість ПЗ створює проблему низького рівня захисту відповідних систем в порівняні з теперішнім рівнем розвитку технологій. Найбільш не захищені є данні які передаються через сервіси цих систем що становить загрозу не тільки приватності, а що більше важливо – можливої фальсифікації чи модифікації цих даних третіми особами, що викликає втрату цілісності самих даних.

Це і призвело до необхідності дослідження та розробки відповідної ІС на основі існуючих методів та їх модифікації з подальшим впровадженням в медичні інформаційні системи.

Мета і задачі роботи. Мета роботи полягає у реалізації інформаційної технології маркування та перевірки автентичності медичних зображень но основі ЦВЗ з покращеним рівнем захисту.

Для досягнення поставленої мети визначенні наступні задачі дослідження:

- Провести аналіз існуючих методів технологій та рішень маркування та перевірки автентичності медичних зображень на основі ЦВЗ;
- Удосконалення існуючих методів маркування та перевірки автентичності медичних зображень на основі ЦВЗ у рамка покращення рівня захисту;
- Розробити інформаційну технологію маркування та перевірки автентичності медичних зображень за допомогою отриманих моделей та методів
- Виконати експериментальну перевірку маркування та перевірки автентичності медичних зображень

Об'єкт дослідження. Процес маркування та перевірки автентичності медичних зображень.

Предмет дослідження. Моделі методи підходи та засоби інформаційної технології маркування та перевірки автентичності медичних зображень

Методи дослідження. Для розв'язання поставлених задач використовуються основні положення методу аналіз даних обробки зображень; для удосконалення методів маркування та перевірки автентичності медичних зображень методами криптографія, стеганографії, сегментації для реалізації інформаційної технології – методології проектування ІС та об'єктно орієнтований та функціональний підходи.

Наукова новизна. В результаті проведеної роботи були отримані такі результати:

- удосконалено існуючі методи маркування та перевірки автентичності медичних зображень на основі ЦВЗ у рамках покращення рівня захисту;

Практичне значення отриманих результатів. В результаті виконання дипломної роботи магістра реалізоване відповідне програмне забезпечення, яке підтвердило вірність запропонованих положень. Застосування ІТ дає можливість

маркувати та перевіряти автентичність медичних зображень з підвищеним ступенем захисту.

Апробація результатів дипломної роботи. Основні наукові та практичні результати опубліковані в фаховому науковому виданні:

– стаття на тему «Сегментація медичних зображень» у журналі «Вісник Хмельницького Національно Університету»

– а темою дипломної роботи магістри виконано одну наукову публікацію [36].

Структура та обсяг роботи. Дипломна робота магістра складається з завдання, реферату, змісту, переліку скорочень, вступу, 4 розділів, висновків, переліку посилань із 36 найменувань та 3 додатків. Загальний обсяг дипломної роботи магістра становить 142 сторінок, з них 86 сторінка основного тексту та 55 сторінок додатків. У роботі наведено 64 рисунки та 1 таблиця.

Ключові слова: інформаційна технологія, маркування зображень, автентичність зображень.

Зміст

Перелік скорочень

Скорочення, термін, позначення	Пояснення
МІС	Медична інформаційна система
ЦВЗ	Цифровий водяний знак
ІС	Інформаційна система
ПП	Програмний Продукт
АЗ	Апаратне забезпечення
ПЗ	Програмне забезпечення
ІТ	Інформаційна Технологія

Вступ

Дипломна робота магістра присвячена розробці інформаційної технології маркування та перевірки автентичності медичних зображень но основі ЦВЗ з покращеним рівнем захисту.

Актуальність теми. Потреба застосування та впровадження в медичні інформаційні системи технології маркування та перевірки автентичності медичних даних обумовлення стрімким ростом кількості відповідних суб'єктів, таких як: приватні та державні лікарні, медичні центри, лабораторії, госпіталів та їх діяльності в МІС, а саме розповсюдження медичних даних, які повинні бути захищеними для кінцевого їх отримувача, тобто автентичним .

Маркування є розповсюдженим явищем в усіх галузях людської діяльності та в більшості з них є обов'язковим та стандартизованим. Не зважаючи на те, що в недалекому минулому термін маркування вживався більше по відношенню до фізичних об'єктів, таких як: деталі, товари, продукти харчування, зараз з розвитком інформаційний технологій цей термін став актуальний для різних інформаційних систем.

Інформаційні медичні системи, здебільшого використовують застаріле програмне забезпечення, яке з кожним роком все складніше підтримувати або взагалі не підтримується розробниками на даний момент. Відповідно, застарілість ПЗ створює проблему низького рівня захисту відповідних систем в порівняні з теперішнім рівнем розвитку технологій. Найбільш не захищені є данні які передаються через сервіси цих систем що становить загрозу не тільки приватності, а що більше важливо – можливої фальсифікації чи модифікації цих даних третіми особами, що викликає втрату цілісності самих даних.

Це і призвело до необхідності дослідження та розробки відповідної ІС на основі існуючих методів та їх модифікації з подальшим впровадженням в медичні інформаційні системи.

Мета і задачі роботи. Мета роботи полягає у реалізації інформаційної технології маркування та перевірки автентичності медичних зображень но основі ЦВЗ з покращеним рівнем захисту.

Для досягнення поставленої мети визначенні наступні задачі дослідження:

− Провести аналіз існуючих методів технологій та рішень маркування та перевірки автентичності медичних зображень на основі ЦВЗ;

− Удосконалення існуючих методів маркування та перевірки автентичності медичних зображень на основі ЦВЗ у рамка покращення рівня захисту;

− Розробити інформаційну технологію маркування та перевірки автентичності медичних зображень за допомогою отриманих моделей та методів

− Виконати експериментальну перевірку маркування та перевірки автентичності медичних зображень

Об’єкт дослідження. Процес маркування та перевірки автентичності медичних зображень.

Предмет дослідження. Моделі методи підходи та засоби інформаційної технології маркування та перевірки автентичності медичних зображень

Методи дослідження. Для розв’язання поставлених задач використовуються основні положення методу аналіз даних обробки зображень; для удосконалення методів маркування та перевірки автентичності медичних зображень методами криптографія, стеганографії, сегментації для реалізації інформаційної технології – методології проектування ІС та об’єктно орієнтований та функціональний підходи.

Наукова новизна. В результаті проведеної роботи були отримані такі результати:

− удосконалено існуючі методи маркування та перевірки автентичності медичних зображень на основі ЦВЗ у рамках покращення рівня захисту;

Практичне значення отриманих результатів. В результаті виконання дипломної роботи магістра реалізоване відповідне програмне забезпечення, яке

підтвердило вірність запропонованих положень. Застосування ІТ дає можливість маркувати та перевіряти автентичність медичних зображень з підвищеним ступенем захисту.

Апробація результатів дипломної роботи. Основні наукові та практичні результати опубліковані в фаховому науковому виданні:

– стаття на тему «Сегментація медичних зображень» у журналі «Вісник Хмельницького Національно Університету»

– за темою дипломної роботи магістри виконано одну наукову публікацію [36];

Структура та обсяг роботи. Дипломна робота магістра складається з завдання, реферату, змісту, переліку скорочень, вступу, 4 розділів, висновків, переліку посилань із 36 найменувань та 3 додатків. Загальний обсяг дипломної роботи магістра становить 142 сторінок, з них 86 сторінка основного тексту та 55 сторінок додатків. У роботі наведено 64 рисунки та 1 таблиця.

Ключові слова: інформаційна технологія, маркування зображень, автентичність зображень.

Розділ 1

Аналіз сучасного стану проблеми маркування та перевірки автентичності медичних зображень

1.1 Аналіз предметної області

В теперішній час з розвитком інформаційних технологій будь–яка галузь діяльності людини адаптується та покращується відповідно до технічного прогресу та розвитку технологій. Саме розвиток та прогрес в області комп'ютерних наук допомагає покращенню, пришвидшенню та вдосконаленню процесів. Інновації з сфери ІТ не оминули і медичну сферу, як основну із сфери людської діяльності, яка безпосередньо зв'язана життям людини, її комфортом та безпекою.

Зараз медична сфера являє собою саме тією галуззю, яка обробляє велику кількість персоналізованих даних які потребують особливого захисту та рівня безпеки. В медичних інформаційних системах визначення «інформаційної безпеки» пов'язаної з даними означає їх цілісність, автентичність, доступність та конфіденційність.

Медичні дані умовно можна розділити на такі категорії [1]:

– медичні зображення (рентгени, знімки магнітно–резонансної томографії, ультра–звукові знімки, тощо) (рис. 1.1) .

– документи (результати аналізів, заключення, рецепти, рахунки, тощо);

– мультимедійні дані спеціалізованих форматів (дані які не можливо зчитати, отримати без спеціалізованого АЗ або спеціалізованого ПЗ)

– інше;

Найбільш незахищеним типом даних є зображення, адже вони передаються найчастіше, як між медичними закладами (для консультації між спеціалістами), так і для передачі пацієнту. В свою чергу документи та мультимедійні дані не настільки часто фігурують в передачі між різними адресантами.

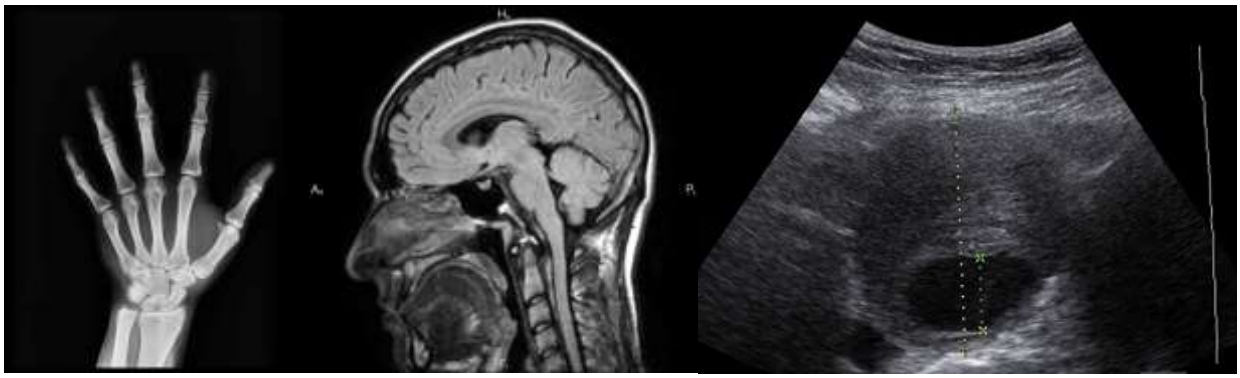


Рисунок 1.1 – Кінцевий результат сегментації: (а) – рентген руки; (б) – МРТ мозку; (в) – ультразвуковий знімок черевної аорти [2, 3, 4];

Варто зауважити, що для передачі медичних зображень та даних існує окремий стандарт DICOM (Digital Imaging and Communications in Medicine) [5]. Його принцип полягає у передачі інформації у форматі набору даних, тобто файл рентгенівського знімка може містити ідентифікаційний код пацієнта а також інший набір додаткових даних, що робить його дещо схожим до передачі простих зображень які теж можуть зберігати набір деяких даних про себе (метаданих). Розглядаючи цей стандарт для передачі даних була виявлена низка недоліків для прикладного використання [6]. Основним недоліком, який не є особливістю ПЗ чи АЗ це розповсюдження цього стандарту в світовій практиці (рис 1.2). Така «популярність» цього стандарту обумовлена в першу чергу тим, що для впровадження його в ІС медичного закладу потрібне відповідне АЗ. Також DICOM як метод для безпечної передачі даних було виявленню ряд негативних факторів, а саме:

– жорстка прив’язка до апаратного забезпечення, а саме сканери, сервери, принтери, тощо;

– можливість вводу великої кількості необов’язкових полів створює проблеми неузгодженості даних або їх неправильності [7];

– також цей формат файлу допускає виконувати програмний код і може містити зловмисне програмне забезпечення [8];

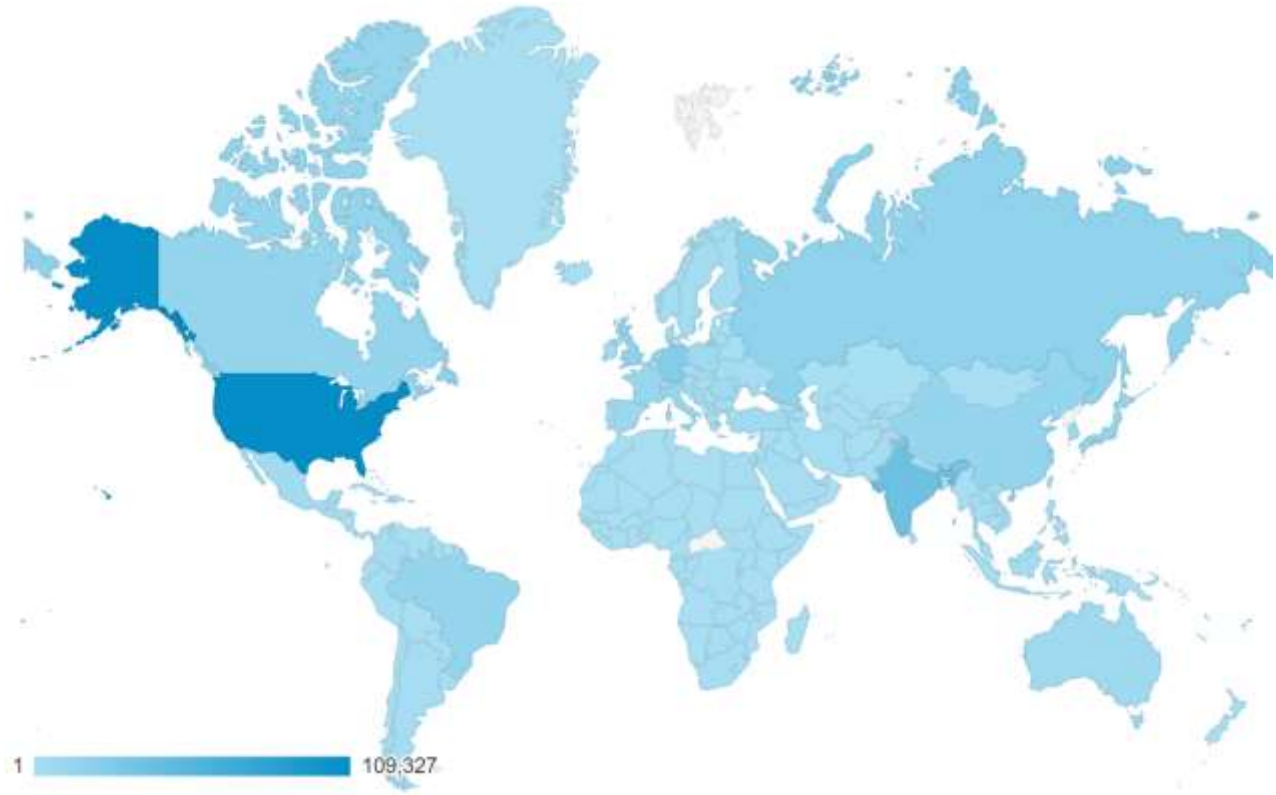


Рисунок 1.2 – Карта використання стандарту DICOM [9]

Тому відповідно виникає необхідність захисту медичних зображень від втручання та можливого спотворення при їх передачі, та відповідної перевірки при отримані цих даних, також відмовитись від використання, модифікації чи покращення існуючого стандарту DICOM, опираючись на недоцільність його використання (рис. 1.2), але розглянути концепцію цього стандарту, а саме передачу додаткової інформації разом з зображенням для подальшого використання в ІС .

Найбільш привабливим та розповсюдженим методом маркування для захисту зображень є метод нанесення цифрового водяного знаку (рис. 1.3). Використання цього методу допомагає досягнути цілей як в комплексі так і окремо:

- захист від несанкціонованого копіювання;
- вистежування джерела розповсюдження інформації;
- перевірка автентичності (перевірка оригінальності)

– інше;

Рисунок 1.3 – Приклад зображення з ЦВЗ [10]

При застосуванні методу ЦВЗ ми стикаємось з низкою обмежень при роботі з такого виду даних. Найбільш вагоме обмеження яке накладається на реалізацію маркування це – можливе спотворення початкових даних та подальша неможливість їх правильного трактування, тобто втрата цілісності інформації з першоджерела.

Для уникнення цього обмеження одне з можливих рішень це – застосування ЦВЗ який є невидимий для людського сприйняття, тобто не сприяє втраті цілісності візуальної інформації після безпосереднього процесу маркування, але зчитуватись та оброблятись відповідними методами ІС [11].

Типи цифрового знаку представлені на рисунку 1.4.

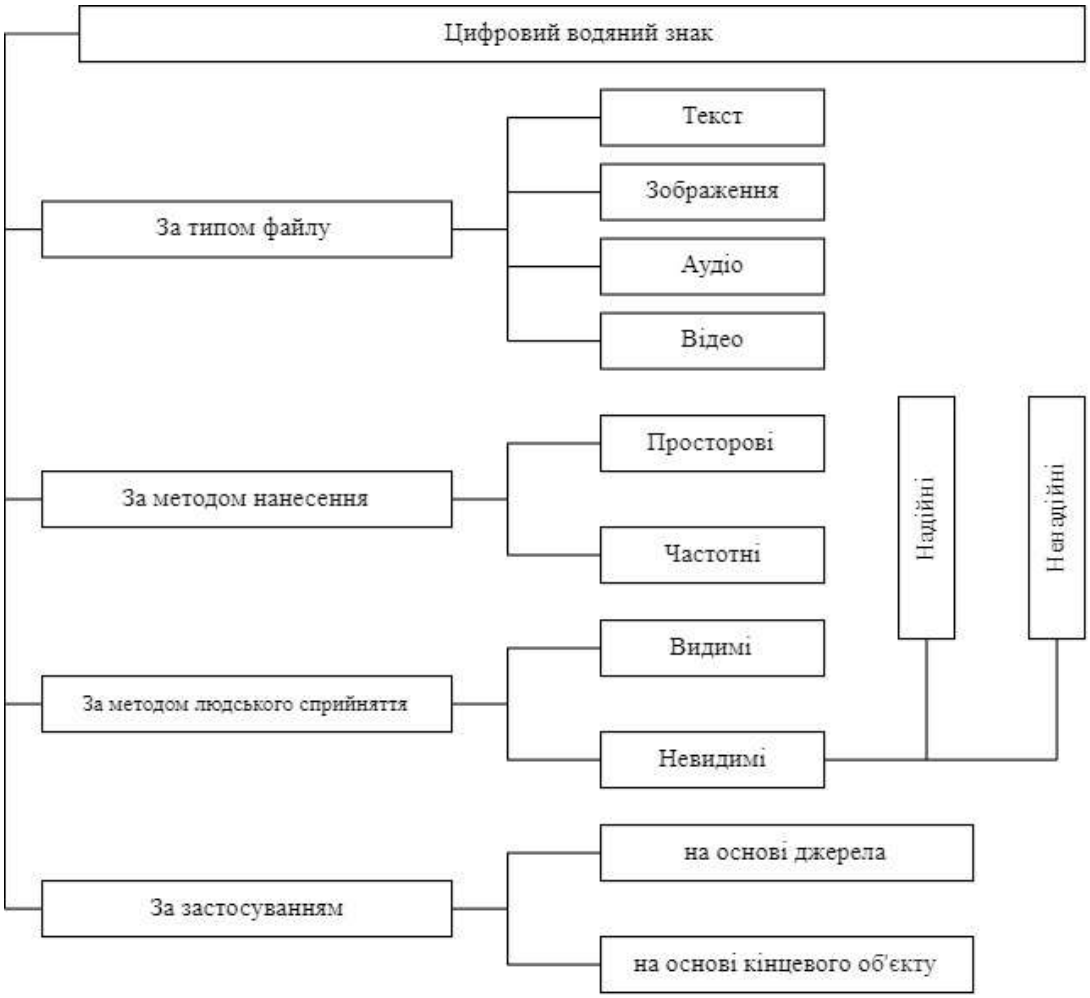


Рисунок 1.4 – Типи цифрового водяного знаку

Очевидно, що одного методу ЦВЗ не достатньо для повного захисту та впровадження цілісності та автентичності даних. Тому варто розглянути маркування в об’єднанні з іншими методами захисту даних. Аналіз інших методів допоможе зрозуміти та виявити можливе об’єднання технік та технологій які в комплексі представлять найкращий результат.

Концепція ЦВЗ тісно пов’язана з двома іншими областями безпечного та зашифрованого способу передачі інформації, а саме з стеганографією та криптографією.

Криптографія – це спосіб надсилання повідомлення або інформації у захищеному форматі, яка може бути розшифрована при наявності відповідного ключа [12].

Але не зважаючи на те, що зашифроване повідомлення захищене під час передачі його по мережі, як тільки воно буде розшифроване воно більше не захищене і це є головний недолік криптографії в порівнянні з ЦВЗ. Більше того більшість криптографічних методів досить складні та представляють слабкий захист авторських прав [13].

Розглядаючи стеганографію [14] в порівнянні з криптографією, то їх методи суттєво відрізняються. При передачі даних або їх зберігання за допомогою методів стеганографії здається враження ще немає ніякого зашифрованого повідомлення, таким чином третя особа принципово не може розшифрувати повідомлення, бо не знає про факт його існування. Що робить його дещо схожим з методом невидимого нанесення ЦВЗ.

Незважаючи на деяку схожість між стеганографією та деякими типами ЦВЗ, є деякі відмінності між ними, а саме:

– Як було описано вище, завдання стеганографії це приховання факту наявності секретного повідомлення чи інформації незалежно де ця інформація буде схована, коли як в ЦВЗ інформація та об’єкт де вона прихована пов’язані [15].

– Стеганографія передбачає невидиме зображення, в той час як ЦВЗ може бути як видимим так і невидимим, але так як ми визначили з методом нанесення ЦВЗ можна вважати, що ця відмінність нівелюється [15].

– Головна мета стеганографії – приховати повідомлення таким чином у об’єкті так, щоб можливий зловмисник не міг його виявити, в той час як головна ЦВЗ – вбудовувати дані в об’єкт таким чином, щоб вони не могли бути видалені або замінені зловмисником [14].

Зважаючи на данні переваги та недоліки, можна зробити висновок що ЦВЗ найкращий вибір для реалізації безпеки медичних зображень, але дані можуть бути зашифровані безпосередньо перед вставкою ЦВЗ або навіть кодування даних безпосередньо в ЦВЗ, що створює додатковий шар захисту. Тому є доцільним розглянути можливість об’єднання цих методів.

Розглядаючи криптографію, стенографію, варто згадати сегментацію зображень, як метод аналізу вхідного зображення.

Сегментація – представляє собою процес відокремлення деяких частин зображення та виділення потрібних ліній та кривих для подальшого їх аналізу [16].

Найбільш розповсюджений метод в задачах комп'ютерного зору [16], тому використовується для таких задач як: розпізнавання об'єктів на фото, розпізнавання обличь, розпізнавання відбитків пальців, тощо. Також цей процес широко застосовуються для медичних зображень де допомагає з низкою завдань, таких як [17]:

– планування операцій;
– віртуальна симуляція операції;
– вивчення анатомічної структури органів;
– діагностування різного типу патологій (рис. 1.5);

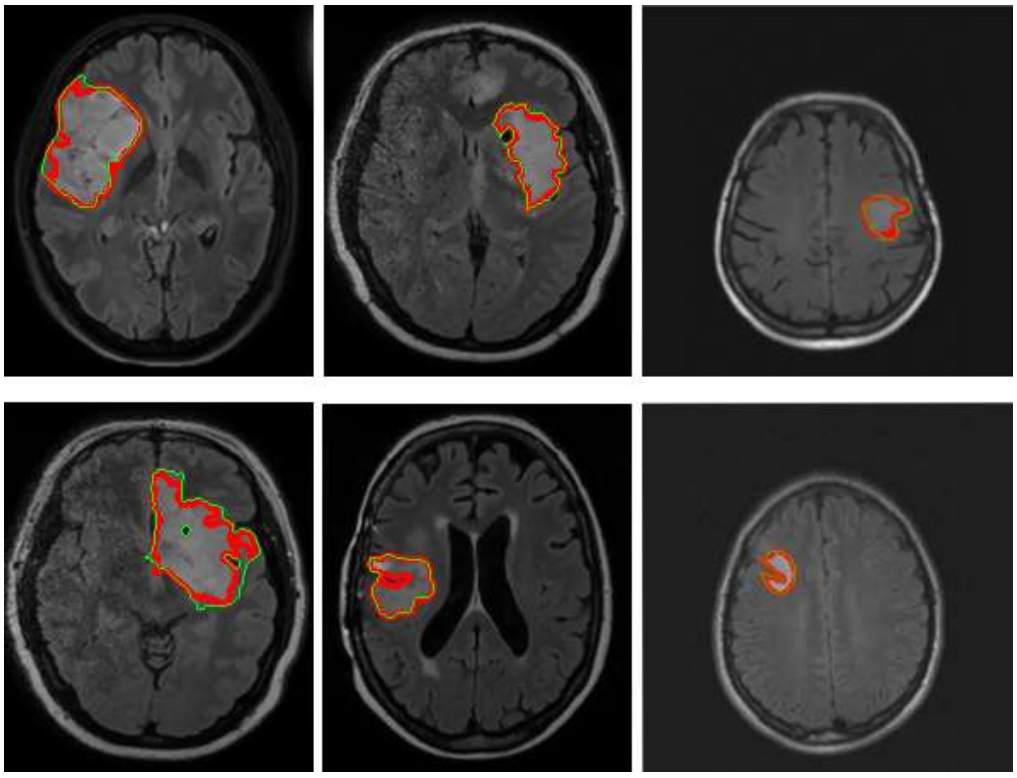


Рисунок 1.5 – Приклад сегментації магнітно–резонансної томографії мозку [18]

Опираючись на розповсюджене використання цього методу для медичних зображень варто прийняти до уваги, що нанесення ЦВЗ не повинно також вплинути на результати трактування не тільки спеціаліста, а й за допомогою автоматичних методів комп'ютерного зору.

Також, цей метод може бути об'єднаний з відповідними методами стеганографії та криптографії, що допоможе створити свого роду «трьох шаровий» рівень захисту при правильному та доцільному об'єднані цих підходів, а саме для визначення областей інтересів (region of interest або ROI), що дає змогу для різного типу маніпуляцій (зчитування, запис, кодування, шифрування, тощо) з відповідними областями для подальшого об'єднання з ЦВЗ, можливо покращить або виправить недоліки та слабкі місця при використані ЦВЗ.

Життєвий цикл може бути описаний наступним чином (рис. 1.6).

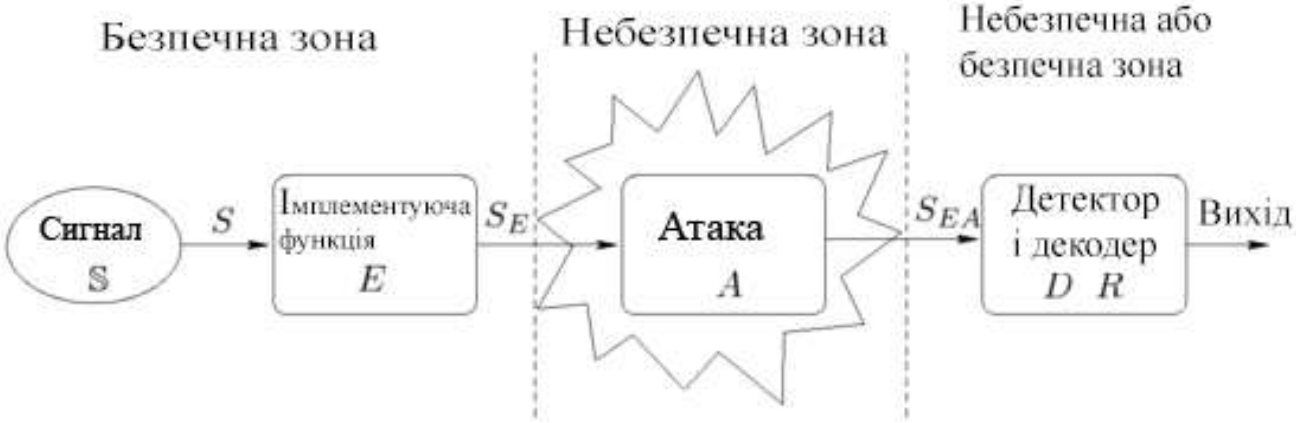


Рисунок 1.6 – Схема життєвого циклу ЦВЗ

Спочатку в сигнал – джерело S в довіреному середовищі наноситься водяний знак за допомогою функції E. В результаті нанесення цього сигналу отримуємо сигнал S_E. Наступний крок це – розповсюдження сигналу через мережу або будь–яким іншим способом. Під час розповсюдження цього сигналу на нього може буте здійсненна атака. У отриманого сигналу S_{EA} ЦВЗ може бути потенційно змінений або знищений. На наступному етапі функція D намагається ідентифікувати деякий водяний знак ω, а в свою чергу функція R розпізнати або

розшифрувати із сигналу деяке секретне повідомлення. Цей процес може бути здійснений зловмисником.

Отже, відповідно до представленої інформації розуміємо що є необхідність в модифікації існуючих методів, можливого їх об’єднання для створення ІС, яка буде корегувати недоліки існуючих алгоритмів з мінімальними вимогами як до середовища використання цієї ІС та апаратного забезпечення.

1.2 Аналіз сучасних наукових робіт застосування цифрового водяного знаку для медичних зображень

Існує безліч публікацій та наукових робіт про різні види застосування ЦВЗ, як для зображень так і для інших типів даних. Так як маркування ЦВЗ – є комплексним завданням яке об’єднує безліч сфер таких як: математика, програмування, криптографія, стенографія тощо, виділити фундаментальні праці не є можливим та доцільним. Основна маса публікацій належить зарубіжним авторам, що говорить про те, що тема за кордоном, а саме: США, Європі та країнах Азії є більше популярною та актуальною, так як в цих частинах світу більш розповсюджена боротьба з несанкціонованим копіюванням та розповсюдженням мультимедійних даних, тобто захист авторських прав. Незважаючи на велику кількість зарубіжних публікацій про маркування медичних зображень за існують також вітчизняні публікації на цю тематику [19,20,21,22,23].

Зарубіжні публікацій в більшості мають рекомендаційний характер щодо методу нанесення ЦВЗ та технічної реалізації з відповідними розрахунками для подальшого впровадження в ІС [24,25,26]. Вітчизняні публікаціях в більшості носять описовий характер явища ЦВЗ та можливості впровадження в ІС [19,20,21].

Якщо розглядати саме маркування або верифікацію чи автентифікацію медичних зображень чи даних, то вітчизняні автори пропонують різні методи реалізації, як на нанесення так і перевірки зображень чи даних.

Наприклад: Л.А. Кузнецова, О.О. Яковенко в роботі «Верифікація медичних даних» [27] при нанесені ЦВЗ спотворюють вхідне зображення, що унеможливлює його подальше трактування лікарем. В роботі «Дослідження методів реверсивних цифрових знаків для верифікації медичних зображень» [28] не було запропоновано однозначного вирішення проблеми спотворення вхідного зображення, а лише викладені припущення щодо оптимальної реалізації відповідних методів при деяких параметрах.

Поверхневий аналіз інших тез та публікацій вітчизняних авторів показав, що були висунуті припущення та розглянуті можливості реалізації різних методів та алгоритмів для впровадження ЦВЗ для медичних зображень, але не запропоновано безкомпромісного та однозначного результату, або впровадження прикладного ПЗ яке б задовільнило сучасні вимоги до ІС та прикладного використання.

Аналіз зарубіжних публікацій показав, що в них висвітлюються більш прикладні підходи, які більш спрямовані на впровадження в ІС та розробки алгоритмів для впровадження в ПЗ. Наприклад роботи [29,30,31,32,33] в свою чергу демонструють опис більш прикладних методів та побудованих відповідних моделей, також моделі перевіряються безліччю тестів, щодо їх доцільності, швидкодії, практичного застосування уже безпосередньо в ПЗ, але самого ПЗ побудованого на основі розроблених моделей не було розглянуто, та не приділено достатньої уваги саме до розробки відповідно ПЗ.

Основні аргументи та порівняння різних методів, які фігурують в цих роботах вказують на те, що метод маркування повинен бути невидимим для людського ока, щоб оригінальне зображення могло бути інтерпретоване лікарем правильно, відповідно до оригінального (вхідного) зображення.

Але в свою чергу задача перевірки автентичності, а саме передача секретного повідомлення чи інформації безпосередньо зав’язаного на ЦВЗ та їх подальшої перевірки не була детально розглянута та вивчена, що вказує на

потребу та актуальність подальшого вивчення та реалізацію відповідних методів, моделей та алгоритмів.

1.3 Дослідження інформаційного забезпечення для маркування та перевірки автентичності медичних даних

На сьогоднішній день існує велика кількість ПЗ для різного типу медичних закладів. В більшості це ПЗ є комплексними рішеннями (рис 1.7) в які входять, так звані «пакети» різних послуг, які можна поділити на такі основні категорії:

– зберігання даних про пацієнтів, лікарів, препаратів, техніку, хвороби;

– ведення графіку запису до лікарів, консультацій, процедур, тощо;

– звітність будь–якого роду як фінансова так й медична;

– електронний кабінет пацієнта з можливістю взаємодії з лікарем та доступною інформацією в кабінеті(результати аналізів, рецепти, документи);

– електронний кабінет лікаря з відповідним функціоналом для взаємодії з потрібними департаментами та службами, заповнення звітності, робочим графіком, тощо;

– електрона реєстратура для реєстрації звернень пацієнтів;

– система управління відносин з клієнтами (CRM), як для лікарів, так і для інших працівників (відділу маркетингу, працівників реєстратури);

– платіжні системи, як для клієнта так і для медичного закладу (оплата рахунків посередникам, лабораторія, працівникам, тощо);

Можна привести як приклад наступні ПП: «Intelligent Medical Software», «lifeIMAGE», «Collective Minds Platform», «PaleBlue», «VEPRO PACS/EMR», «Medesk», «MEDODS», «МЕДМИС», «Инфоклиника», «Renovatio», тощо.

Також варто зауважити, що більшість ПЗ потрібно встановлювати безпосередньо на робочу машину, що позбавляє його в гнучкості використання з усіх можливих платформ, а також збільшує його вартість та неможливість його інтеграції в клініки та заклади, які не мають потрібного апаратного забезпечення

(комп'ютерів з потрібною потужністю, операційною системою, тощо). Також, це ускладнює, або унеможливлює взаємодію пацієнта та клініки, що є великим недоліком для такого роду системи.

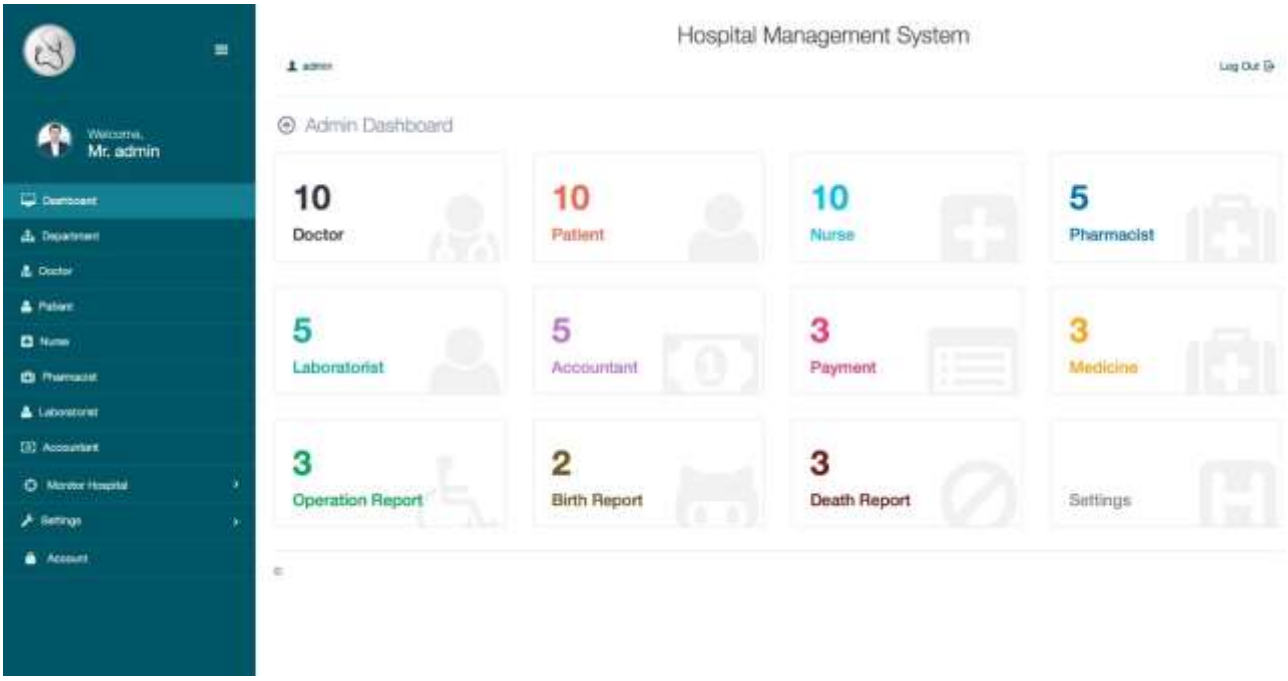


Рисунок 1.7 – Приклад дизайну та функціоналу медичного комплексного ПЗ [34]

Новітнє ПЗ, яке розроблено як веб–додаток (в форматі веб–сайту) є більш гнучким у використані так як може бути запущене на будь–якому пристрої який дозволяє відкривати веб сторінки, що дає змогу використовувати одну і ту ж ІС, як лікарю так і пацієнту в залежності від відповідних ролей.

В більшості випадків саме такі ПП відповідають за передачу будь–якого типу файлів між лікарями, пацієнтами та іншими користувачами. Встановити методи захисту цих медіа файлів не являється можливим без доступу до коду, так як більшість таких ПЗ являється платним та закритим від публічного доступу можна лише робити припущення щодо наявного захисту.

При аналізі існуючих програмних продуктів не було виявлено спеціальних додатків саме для маркування медичних зображень, можна зробити

припущення, що відповідна ІС з маркуванням та перевіркою медичних зображень могла би бути інтегрована в одне із комплексних рішень приведених вище.

Отже, на основі описаних ПП можна зробити висновок про доцільність розробки та покрашення існуючих МІС саме у веб середовищі, та реалізація ІС та ПП, які можна імплементувати у існуючі ПП.

1.4 Висновки до розділу та постановка задачі

Проведений аналіз літературних джерел засвідчує потребу застосування та впровадження в медичні інформаційні системи технології маркування та перевірки автентичності медичних даних обумовлення стрімким ростом кількості відповідних суб'єктів, таких як: приватні та державні лікарні, медичні центри, лабораторії, госпіталів та їх діяльності в МІС, а саме розповсюдження медичних даних, які повинні бути захищеними для кінцевого їх отримувача, тобто автентичним.

Маркування є розповсюдженим явищем в усіх галузях людської діяльності та в більшості з них є обов'язковим та стандартизованим. Не зважаючи на те, що в недалекому минулому термін маркування вживався більше по відношенню до фізичних об'єктів, таких як: деталі, товари, продукти харчування, зараз з розвитком інформаційний технологій цей термін став актуальний для різних інформаційних систем.

Інформаційні медичні системи, здебільшого використовують застаріле програмне забезпечення, яке з кожним роком все складніше підтримувати або взагалі не підтримується розробниками на даний момент. Відповідно, застарілість ПЗ створює проблему низького рівня захисту відповідних систем в порівняні з теперішнім рівнем розвитку технологій. Найбільш не захищені є данні які передаються через сервіси цих систем що становить загрозу не тільки приватності, а що більше важливо – можливої фальсифікації чи модифікації цих даних третіми особами, що викликає втрату цілісності самих даних.

Це і призвело до необхідності дослідження та розробки відповідної ІС на основі існуючих методів та їх модифікації з подальшим впровадженням в медичні інформаційні системи.

В результаті проведеного аналізу існуючих підходів сформульовані такі завдання дослідження роботи полягає у реалізації інформаційної технології маркування та перевірки автентичності медичних зображень но основі ЦВЗ з покращеним рівнем захисту:

- Провести аналіз існуючих методів технологій та рішень маркування та перевірки автентичності медичних зображень на основі ЦВЗ;
- Удосконалення існуючих методів маркування та перевірки автентичності медичних зображень на основі ЦВЗ у рамка покращення рівня захисту;
- Розробити інформаційну технологію маркування та перевірки автентичності медичних зображень за допомогою отриманих моделей та методів;
- Виконати експериментальну перевірку маркування та перевірки автентичності медичних зображень.

Розділ 2

Розробка інформаційної технології маркування медичних зображень

2.1 Загальний опис технології маркування та перевірки автентичності медичних зображень

Загальність послідовність технології маркування та перевірки автентичності медичних зображень включає наступні ключові етапи (рис. 2.1):

1) Отримання вхідного зображення
2) Сегментація вхідного зображення
3) Шифрування отриманих результатів сегментації
4) Отримання потрібних даних користувача.
5) Шифрування відповідних даних
6) Конкатенація результатів шифрування даних
7) Нанесення цифрового водяного знаку
8) Перевірка автентичності (наявності ЦВЗ та його оригінальність)

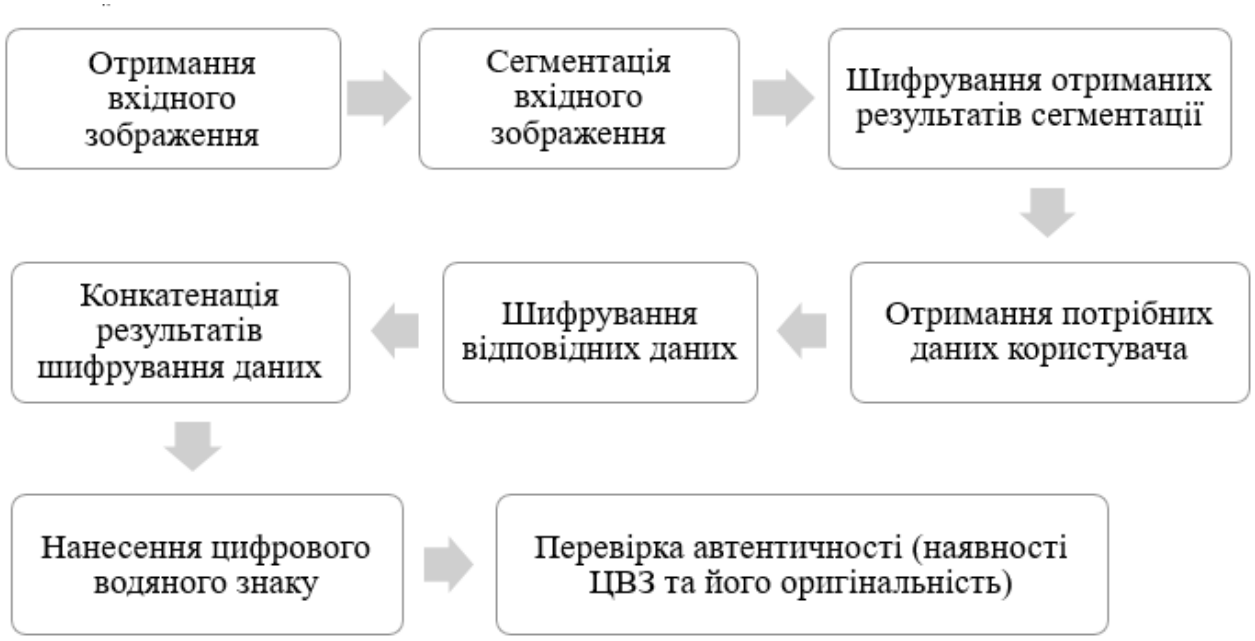


Рисунок 2.1 – схема ключових етапів технології маркування та перевірки автентичності медичних зображень.

2.2 Отримання вхідного зображення

Вхідні зображення, які отримуються від відповідного користувача системи (лікаря, рентгенолога, тощо), можна розділити відповідно до технології отримання медичного зображення, а саме (рис 1.9):

- Рентгенографія
- Магнітно–резонансна томографія (МРТ)
- Ядерна медицина
- Ультразвук
- Еластографія
- Тактильна візуалізація
- Фотоакустичні зображення
- Термографія
- Ехокардіографія

Рисунок 2.2 – Технології отримання медичного зображення

Вхідне зображення має наступні властивості (рис 2.4):

– Розмір зображення, який може виражатись в кількості пікселі по ширині та висоті (800 * 600px), так і як загальна кількість пікселів (2 000 000px);

– Кількість використовуваних кольорів або глибина кольору (ці характеристики мають наступну залежність: $N = 2^k$, де N кількість кольорів, k глибина)

– Колірний простір RGB, XYZ, CMYK та інші;

– Роздільна здатність зображення – величина, що визначає кількість точок (елементів растрового зображення) на одиницю площі (або одиницю довжини)

Рисунок 2.4 – Властивості вхідного зображення

Отже, оцінивши властивості медичних зображень на основі розглянутих прикладів (рис 2.3) та їх типи можна зробити висновок, що їх характеристики такі як глибина кольору, колірний простір є приблизно однаковими.

2.3 Сегментація вхідного зображення

Отримавши вхідне зображення, незважаючи на те, що медичні зображення мають схожі властивості, виконується перетворення зображення в чорно–біле (рис. 2.5), щоб нівелювати та відсікти вплив кольору при сегментації. В такому

зображені кожен піксель представляє інформацію лише про свою інтенсивністю забарвлення, відповідне перетворення можна представити наступним чином (2.1):

$$C_{\text{linear}} = \begin{cases} \dfrac{C_{\text{srgb}}}{12.92}, & \text{якщо } C_{\text{srgb}} \leq 0.04045 \\ \left(\dfrac{C_{\text{srgb}} + 0.055}{1.055}\right)^{2.4}, & \text{інакше} \end{cases} \quad (2.1)$$

Де, C_{srgb} представляє собою будь–який з трьох зжатих параметрів (R_{srgb} ,G_{srgb} , B_{srgb} , кожен в діапазоні [0;1]) і C_{linear} відповідає за значення лінійної інтенсивності (R_{linear} ,G_{linear} , B_{linear} , також кожен в діапазоні [0;1]).

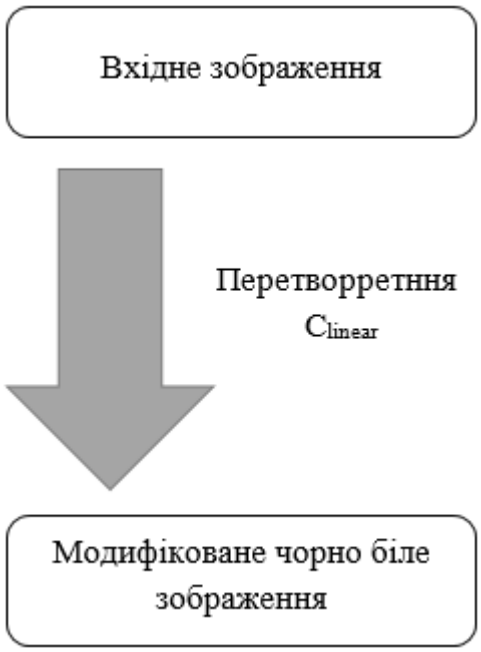


Рисунок 2.5 – Перетворення кольорового зображення в чорно–біле

Після того як зображення нормалізоване для сегментації, визначається область інтересів – найбільш важливу частину медичного зображення. Вона містить найцінніші дані в медичному зображенні і не повинна зазнавати змін в процесі передачі та нанесення ЦВЗ.

В даному випадку використовується метод росту регіону. Нехай отримано зображення $I(x, y)$ розміру 256 * 256 пікселів, потрібно визначити відповідну область інтересів з цього зображення, потрібний алгоритм дій це представлено наступним чином (рис 2.5):

1) Обчислюється градієнт вхідного зображення I для як для осі x (I_{Rx}), так й для осі y (I_{Ry}).

2) Після цього рахується вектор градієнта I^G, отримавши гібрид значень градієнта за допомогою наступного рівняння

$$I^G = \frac{1}{1 + \left(I_{Rx}^2 + I_{Ry}^2\right)} \tag{2.2}$$

3) Потім проходить заміна одиниці виміру значення вектору градієнта, яка зазвичай знаходяться в радіанах, на градуси, щоб досягти значень орієнтації

4) Розділення зображення на сітки G^i.

5) Встановлення порогу інтенсивності (*TIN*) та поріг орієнтації (*TOR*).

6) Для кожної сітки G^i повторюються подальші процеси на кроці 7, поки кількість сіток не досягне загальної кількості сіток для зображення

7) Розраховується гістограму H кожного пікселя в G^i.

8) Регулюється найбільш часта гістограма G^{ith} сітку і представити її у вигляді F^H.

9) Вибирається будь–який піксель відповідно до F^H та виконується його видалення як початкової точки, яка має інтенсивність IN_n та орієнтацію OR_n.

10) Розглянемо сусідній піксель, що має інтенсивність IN_n та орієнтацію OR_n.

11) Рахується різниця інтенсивності та орієнтації цих пікселів, p та n. В результаті отримаємо (2.2 – 2.3):

$$D_{IN} = \left\| IN_p - IN_n \right\| \tag{2.3}$$

$$D_{OR} = \left\| IN_p - IN_n \right\| \quad (2.4)$$

12) Якщо $D_{IN} \leq T_{IN}$ та $D_{OR} \leq T_{OR}$, додається послідовний піксель до області, і область росте; в іншому випадку виконується перехід до кроку 14.

13) Відбувається перевірка, чи всі пікселі додані до регіону. Якщо так, виконується перехід до кроку 6, а потім до кроку 14.

14) Повторно проходить оцінка області та виявляються нові початкові точки (seed points), і виконується процедуру з кроку 7.

15) Зупинка всієї процедури.

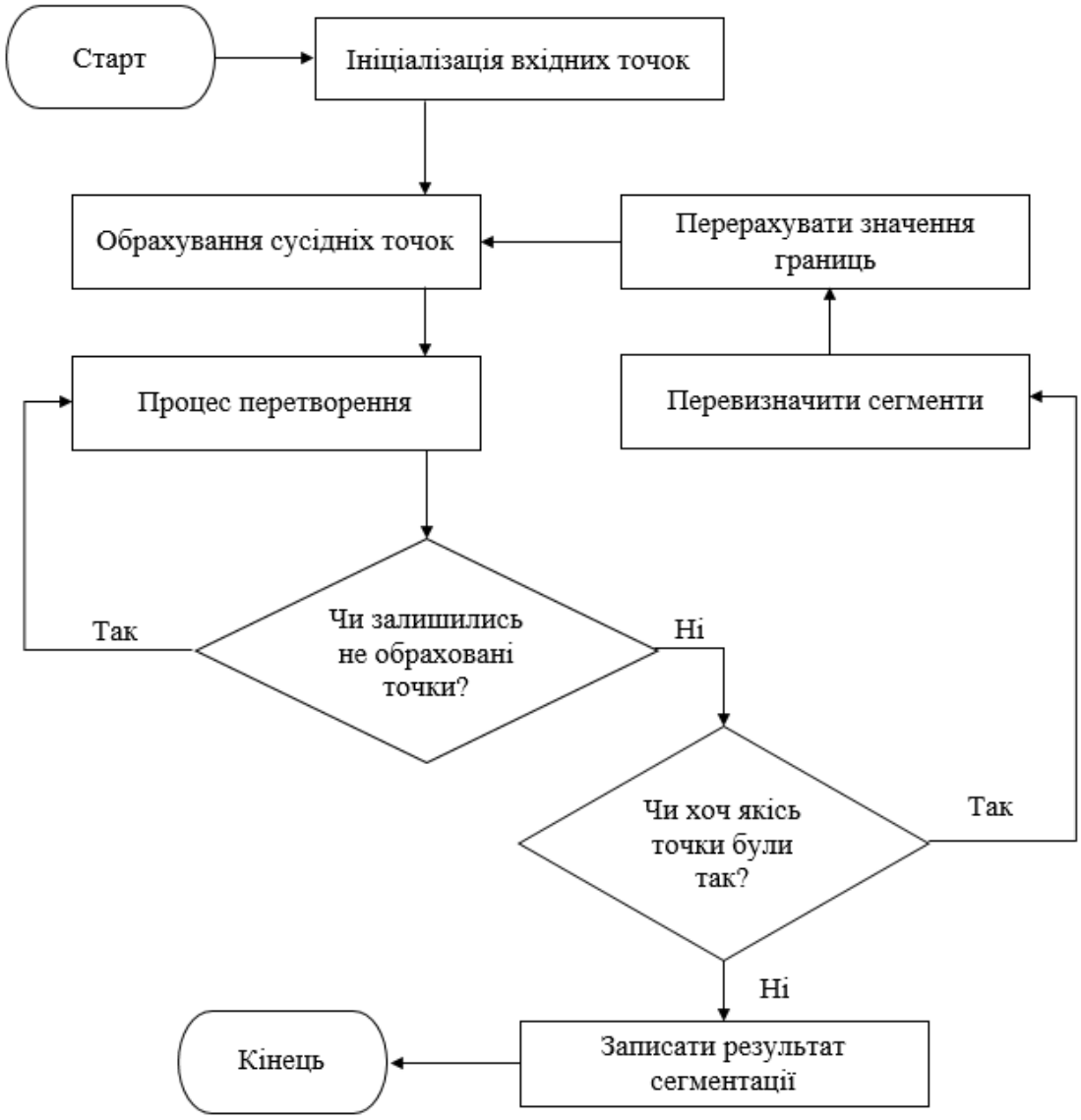


Рисунок 2.6 – Узагальнений алгоритм сегментації методом росту регіонів

За допомогою методу росту регіонів вхідні зображення сегментуються та визначаються відповідні області інтересів.

2.4 Шифрування отриманих результатів сегментації

Після того як вхідне зображення було просегментовано та були отримані області інтересі, даний етап можна представити наступним чином (рис 2.7):

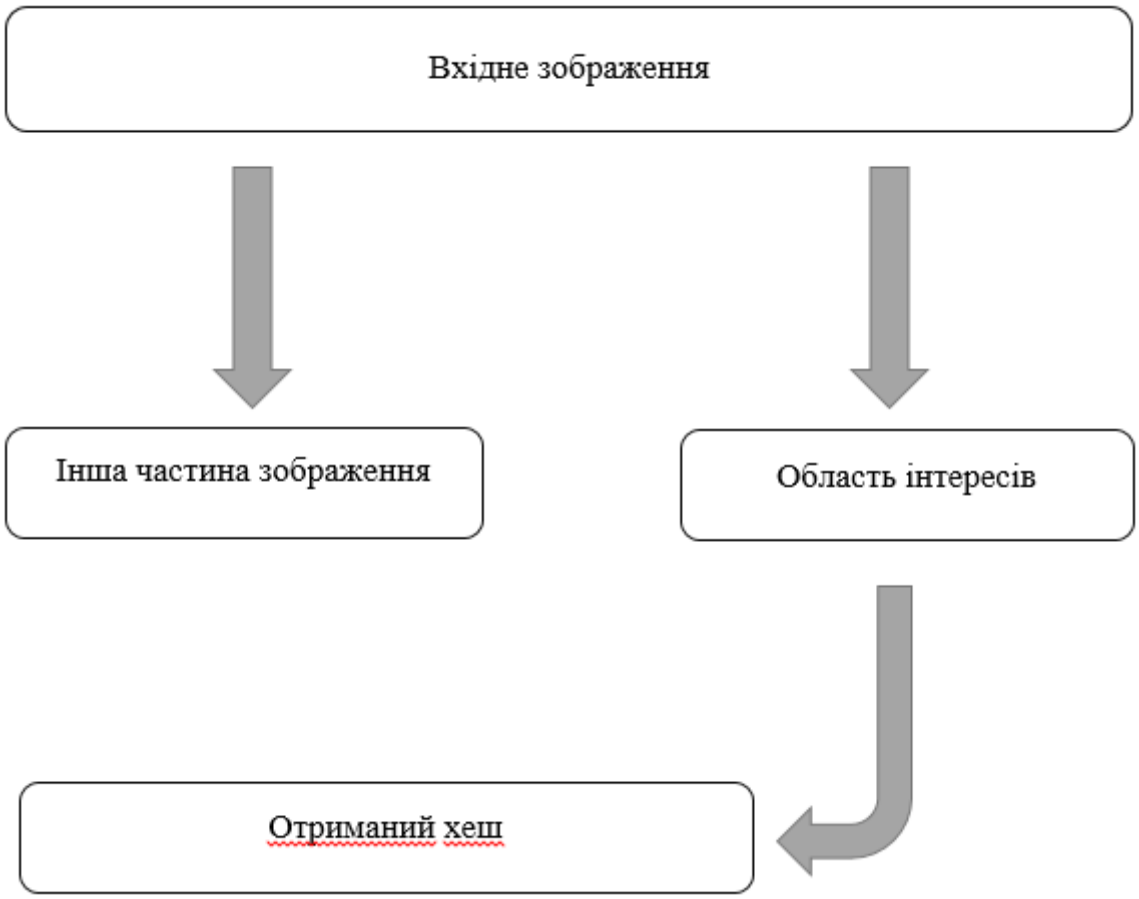


Рисунок 2.7 – Схема отримання хеш коду з області інтересів

Як представлено на малюнку (2.7) області інтересів а саме їх відповідні якісні характеристики (розмір та діаметр області сегментації, тощо) шифруються за алгоритмом SHA–256 (рис 2.7).

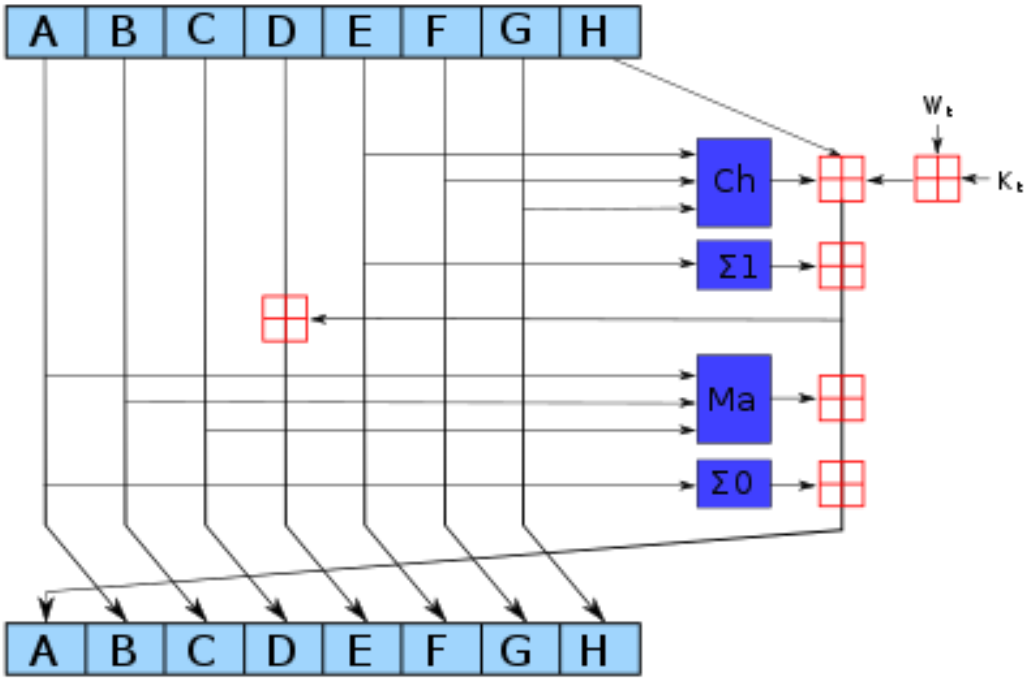


Рисунок 2.8 – Схема роботи алгоритму SHA–256

Блоки синього кольору представлені на рисунку 2.8 виконують наступні операції:

$$\mathrm{Ch}(E,F,G) = (E \wedge F) \oplus (\neg E \wedge G) \tag{2.5}$$

$$\mathrm{Ma}(A,B,C) = (A \wedge B) \oplus (A \wedge C) \oplus (B \wedge C) \tag{2.6}$$

$$\Sigma_0(A) = (A \ggg 2) \oplus (A \ggg 13) \oplus (A \ggg 22) \tag{2.7}$$

$$\Sigma_1(E) = (E \ggg 6) \oplus (E \ggg 11) \oplus (E \ggg 25) \tag{2.8}$$

Цей алгоритм шифрування дозволяє згенерувати унікальний код для кожного вхідного зображення, що дозволить використовувати відповідне значення області інтересів для її автентифікації та додати додатковий рівень захисту при нанесені ЦВЗ.

2.5 Отримання потрібних даних користувача.

Також для забезпечення впровадження додаткового рівня захисту для ЦВЗ та для кращої перевірки автентичності, а саме передачу корисного об'єму деякої інформації такої як (рис. 2.9):

- Завантажувач зображення (лікар або працівник з відповідним доступом)
- Дата та час завантаження зображення в систему
- Метадані зображення
- Діагноз або коментар лікаря
- Інформація про систему в яку було завантажено зображення (тип лікарні, назва лікарні, тощо)
- Інше

Рисунок 2.9 – Дані які шифруються для вставлення в ЦВЗ

Але передача цієї інформації разом з ЦВЗ у відкритому вигляді не є безпечним, тому запропоновано передавати ці дані в зашифрованому вигляді за допомогою алгоритмів еліптичної криптографії (ECC).

2.6 Шифрування медичних даних

Еліптична криптографія будується на еліптичних кривих над кінцевими полями, головна перевага цих методів в тому, що на даний момент є невідомим існування алгоритмів для вирішення завдань дискретного логарифмування, що означає високу надійність таких методів. Також дані методи підходять для використання відкритого та закритого ключа. Архітектура пропонує чотириступінчасту техніку для гарантування автентичності користувача (рис.2.10).

Рисунок 2.10 – Процес шифрування на основі ECC

Роботу даного алгоритму в цьому випадку можна розділити на такі основні кроки .

1) У цифровому підписі дані будуть здрібнені на кілька рядків, які називаються хешем повідомлення за допомогою алгоритму хешування.

2) Хеш повідомлення визначається за допомогою закритого ключа для забезпечення цифрового підпису.

3) Використовуючи алгоритм ECC, хеш повідомлення шифрується за допомогою відкритого ключа користувача.

4) Власник даних розшифрує цифровий підпис у дайджесті хеш повідомлення за допомогою відкритого ключа та перетворить текст шифру в звичайний текст цим приватним ключем, як показано рисунку 2.10.

Криптосистема еліптичної кривої працює над основі опису еліптичної кривої. Для поточних криптографічних розв'язувань еліптична крива – це плоска крива, яка охоплює точки, що задовольняють рівняння. Рівняння еліптичної кривої над полем *K***,** розраховується наступним чином (2.9):

$$x^3 = y^3 + ay + b \tag{2.9}$$

де *x, y* координати і *a, b* є елементами *K*.

Існує три фази процесу, тобто генерація ключа, шифрування та дешифрування.

Генерація ключів – це важливий процес, завдяки якому необхідно надати як відкритий, так і приватний ключ. Відправник запрограмує повідомлення у відкритому ключі власника розробки, а власник даних розшифрує за допомогою закритого ключа. В даний час обирається число f всередині діапазону m. Надати відкрий ключ можна використовуючи таке рівняння (2.10):

$$H = f \cdot q \tag{2.10}$$

де f – випадкове число, яке визначене в межах (від 1 до m –1);

q – точка на кривій, H – відкритий ключ;

f – приватний ключ.

Фаза шифрування полягає в наступному: нехай p – деяке делікатне повідомлення. Щоб охарактеризувати це повідомлення на кривій необхідно розглянути p до точки M на кривій E. Довільно обрано k з [1− (m−1)]. Зашифровані тексти будуть створені після шифрування; нехай вони будуть R_1 та R_2**.**

$$R_1 = k \cdot q \tag{2.11}$$

$$R_2 = M + k \cdot H \tag{2.12}$$

Дешифрування представляє собою такий процес: надіслане повідомлення M транслітерується як рівняння (2.13):

$$M = R_2 - f \cdot R_2 \tag{2.13}$$

Таким чином, конфіденційні дані шифруються, і авторизовані користувачі можуть отримувати інформацію на основі підключеного запиту. За потреби користувач також може отримати доступ до конфіденційної інформації, використовуючи дані доступу від власника даних.

2.7 Конкатенація результатів шифрування даних

Після отримання відповідних зашифрованих даних сегментації області інтересів та відправлених даних відбувається їх конкатенація (рис. 2.11).

Рисунок 2.11 – Конкатенація зашифрованих даних

Після безпосередньої конкатенації дані готові для нанесення на медичне зображення.

2.8 Нанесення цифрового водяного знаку

2.8.1 Методи нанесення ЦВЗ

В загальному методи нанесення цифрового водяного знаку діляться на дві групи основні (рис 2.12):

- просторові;
- частотні;

Рисунок 2.12 – Методи нанесення цифрового водяного знаку

2.8.2 Просторові методи нанесення ЦВЗ

Дані техніки вставляє інформацію водяного знаку у вхідне зображення, як визначено власником у просторовій або часовій області, використовуючи різні методи, включаючи алгоритми модифікації найменш значущих бітів (LSB), проміжні значущі біти (ISB) або алгоритми Patchwork , а також розширює спектр та кореляцію на основі алгоритмів.

Ці методи (рис 2.13) працюють безпосередньо на вихідних пікселях зображення. ЦВЗ знак можна вставити, маніпулюючи значеннями пікселів на основі логотипу або інформації про підпис, наданим користувачем.

У найбільш часто використовуваних конструкціях інтенсивність пікселів у відомих точках простору представляє зображення, де біт найнижчого порядку певних пікселів у кольоровому або сірому масштабі перевертається. Залежно від інтенсивності пікселів отриманий водяний знак може бути видимим або невидимим.

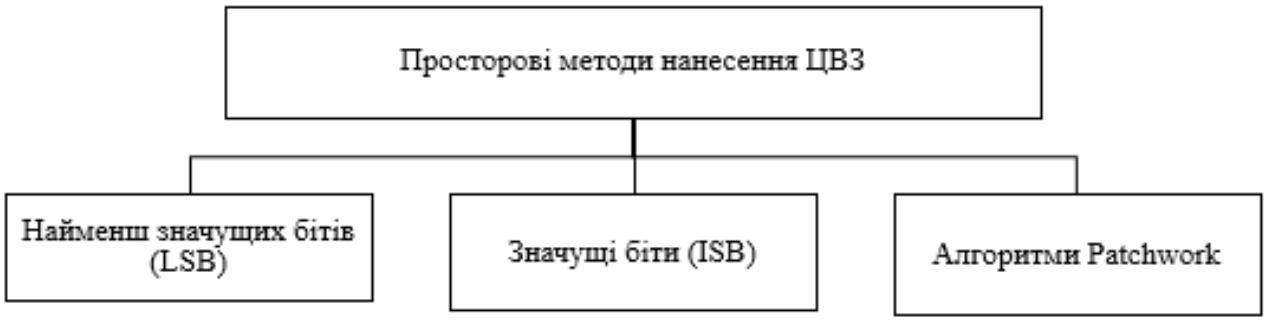


Рисунок 2.13 – Види просторових методів нанесення ЦВЗ

Дані підходи щодо методів просторового нанесення ЦВЗ є досить популярними завдяки їх оптимальному балансу між непомітністю, міцністю та ємністю, які є найважливішими вимогами будь–якої техніки водяного маркування.

Ці методи мають низьку складність, покращену ефективність та швидке виконання. Крім того, якість зображення з водяними знаками може контролюватися. Однак ці методи добре працюють лише в тому випадку, якщо зображення не піддається будь–яким шумам або людським модифікаціям.

Обрізання зображення можна використовувати для виключення водяного знаку, що є головною слабкою стороною просторового ЦВЗ. Ці методи вбудовують великий обсяг даних з точки зору ємності, але вставлені дані можуть бути легко виявлені різними атаками.

Крім того, невеликий предмет можна вставити кілька разів. Отже, єдиний збережений водяний знак буде вважатися досягненням, незважаючи на втрату більшої частини зображення через кілька атак.

2.8.3 Частотні методи нанесення ЦВЗ

Частотні методи нанесення ЦВЗ водяних знаків просторового домену є досить крихкими, оскільки ними можна легко маніпулювати. Ці методи набагато менш надійні проти різних типів атак, порівняно з алгоритмами просторового нанесення. Ці недоліки зосередили увагу на дослідженні методів нанесення водяних знаків трансформаційного домену, які приховують дані в просторі перетворення сигналу, а не в часі, більш ефективним способом.

Ця техніка перетворює зображення за допомогою заздалегідь визначеного перетворення, щоб представити зображення в частотній області. Потім вона вбудовує водяний знак, змінюючи коефіцієнти домену перетворення оригінального зображення, використовуючи різні перетворення, включаючи дискретне перетворення косинусів (DCT), дискретне перетворення Фур'є (DFT), дискретне перетворення вейвлетів (DWT), розкладання сингулярних значень (SVD) Трансформація Адамара, CAT, FFT, PHT та Френеля, серед інших.

Нарешті, він витягує водяний знак за допомогою правильного ключа, використовуючи зворотне перетворення. На рисунку 2.14 описана вищезазначена процедура.

Для відновлення вихідного сигналу в частотній області частотні компоненти повинні бути рекомбіновані, застосовуючи інформацію про фазовий зсув до кожної синусоїди зображення.

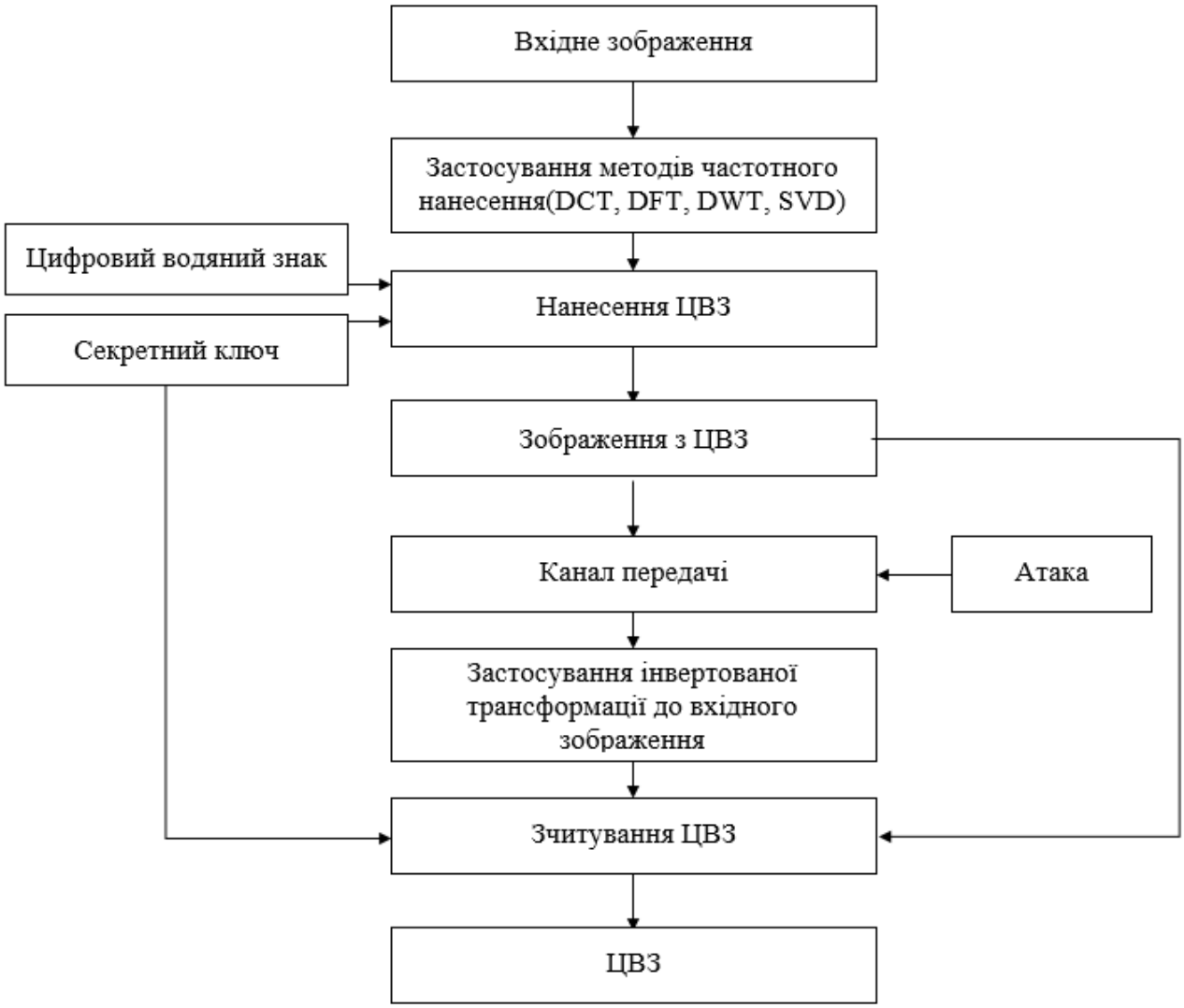


Рис 2.14 – Вбудовування та вилучення водяних знаків у домені перетворення.

2.8.4 Процес нанесення ЦВЗ (маркування)

Запропонований метод нанесення ЦВЗ (рис. 2.15) на медичне зображення базується на основі пройдених етапів в розділі 2. Тобто нанесення ЦВЗ забезпечується декількома методами шифрування та на основі гібридизації різних методів нанесення ЦВЗ.

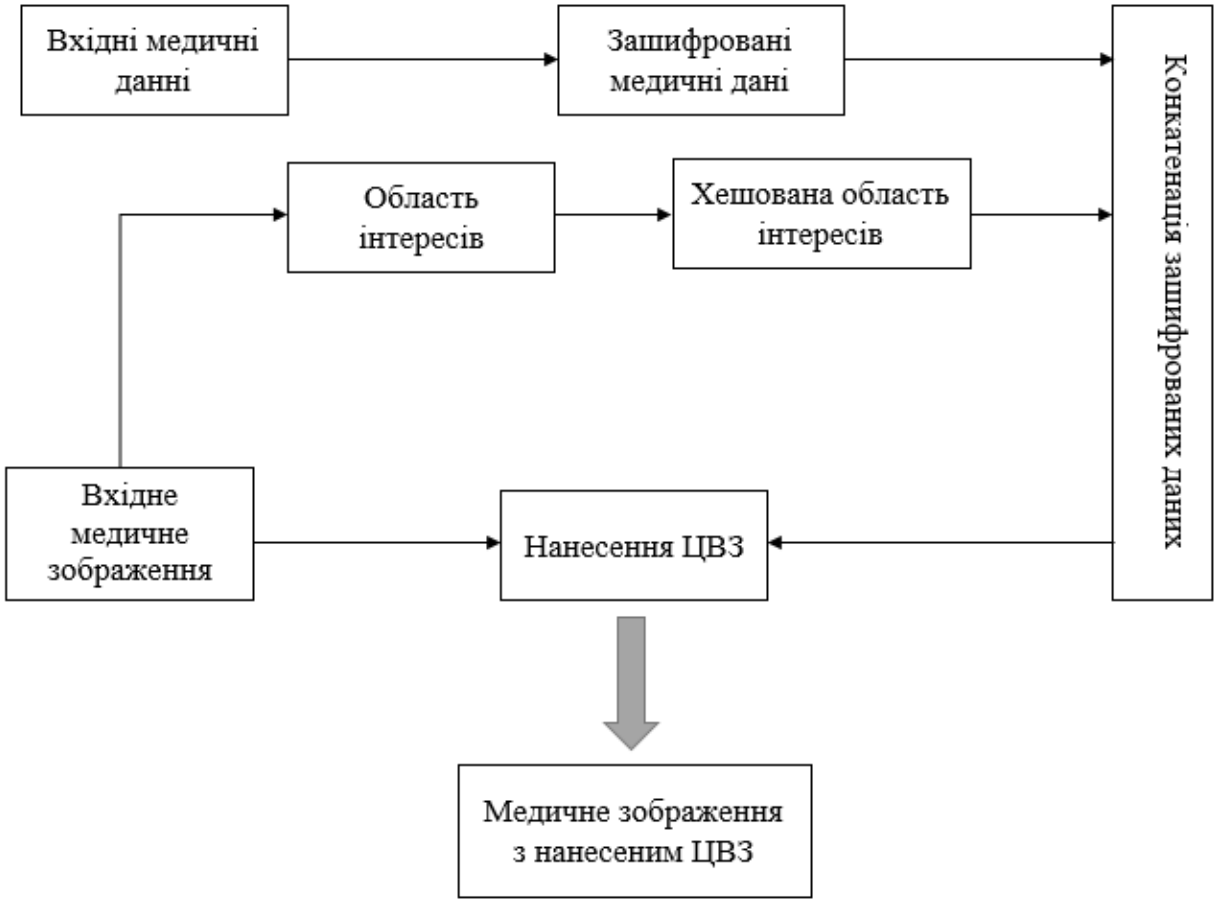


Рисунок 2.15 – Схема запропонованого методу нанесення ЦВЗ

Тобто повний процес нанесення ЦВЗ на медичне зображення складається з наступний кроків (рис 2.15):

1) Розглянемо вхідне зображення I розміром 256 × 256. Спочатку сегментується область ROI (I_{ROI}) із вхідного зображення за допомогою алгоритму RG (див. пункт 2.3)

2) Після процесу сегментації проводяться обчислення хеш–функції області інтересів ($Hash_{\text{ROI}}$) за допомогою *SHA–256* (див. пункт 2.6).

3) Потім розглядається інша частину вкладених даних, а саме медичні дані (EHR). Тут шифрується EHR, використовуючи алгоритм ECC. EHR складається з відповідних переданих даних. Використовуючи алгоритм ECC, отримується зашифрований EHR (E_{encrypt}).

4) Після процесу шифрування об'єднується інформація про зображення за допомогою EHR. Тут, конкатенації H_{ROI}, H_{EHR} та RH_{ROI}, ми отримуємо CON_I.

5) Потім генерується початкова випадкова матрицю *R*, використовуючи наступні кроки. Значення пікселів на вхідному зображенні *I* підсумовуються і позначаються як I_{seed}.(2.14):

$$I_{seed} = \sum_{i=1}^{n} \sum_{j=1}^{n} I_{ij} \tag{2.14}$$

Випадкова матриця *R* генерується з розміром вхідного зображення за допомогою генерації псевдовипадкової матриці з I_{seed} (2.15):

$$R = PRMG[R_{seed}]_{(2\times2)} \tag{2.15}$$

6) Потім генерується остаточна випадкова матрицю *RM* через *R*, використовуючи такі кроки: Значення 0,5 віднімають із сформованої випадкової матриці *R*, а отриману матрицю множать на 2. Кінцеву результуючу матрицю позначають як R_t. (2.16):

$$R_t = (R - 0.5) \times 2 \tag{2.16}$$

Нарешті, передбачувана випадкова матриця RM генерується з використанням генератора псевдовипадкових матриць з матрицею R_t як початковою величиною (2.17):

$$RM = PRMG[R_t]_{(2\times2)} \tag{2.17}$$

7) Потім C_{BS} маркується у вихідне зображення I . Нарешті, отримується зображення I_W з нанесеним ЦВЗ. Процес вбудовування наведено нижче.

Якщо біт цифрового водяного знаку дорівнює 0, це означає, що випадкова матриця RM множиться на силу вбудовування β і результуюча матриця додається до вихідного зображення I (2.18).

$$[I_M] = [I_M] + (\beta \cdot [RM]), \qquad \text{де } \beta = 2 \tag{2.18}$$

Якщо біт водяного знаку дорівнює 1, це означає, що жодна операція не виконується.

8) Кроки 6 і 7 повторюються доки не вбудуються всі пікселі цифрового водяного знаку. Початкова випадкова матриця R для кожної ітерації генерується з $PRMG$, з новими початковими точками I_{seed}.

2.9 Процес вилучення (перевірка автентифікації)

Під час процесу вилучення виконується операція, зворотна до операції маркування, для вилучення стисненого біта C_{BS} із підозрілого зображення з водяним знаком I_W та порівняння його з вхідним зображенням I та інформацією EHR. Для того, щоб виявити вихідну інформацію, спочатку алгоритм вилучення виконує ту ж операцію, що і алгоритм маркування. Спочатку вбудоване зображення з ЦВЗ I_W подається на вхід операції екстракції. Процес вилучення наведено, можна описати наступним чином:

1) У процесі автентифікації спочатку генерується випадкова матриця *RM*, використовуючи етапи 8 та 9 (див. 2.8.4). Відповідно до попереднього підрозділу, початкова випадкова матриця *R* для кожної ітерації генерується з *PRMG*, нового I_{seed}.

2) Потім обчисліть коефіцієнт кореляції R^{Cor} між зображенням із водяним знаком I_W та сформованою випадковою матрицею [RM], використовуючи таке рівняння (2.19):

$$R^{Cor} = \frac{\sum_m \sum_n (I_W^{mn} - I_W)(B_{mn} - B)}{\sqrt{(\sum_m \sum_n (I_W^{mn} - I_W)^2)(\sum_m \sum_n (B_{mn} - B)^2)}} \quad (2.19)$$

де I_W^{mn} являє собою зображення з водяним знаком, B_{mn} являє собою випадкову матрицю *RM*, $\overline{I_W}$ являє собою середнє значення I_W.

3) Розділити обчислене значення коефіцієнта кореляції R^{Cor} два і позначить результуюче значення , як R_Z (2.20).

$$R_Z = \frac{R^{Cor}}{2} \quad (2.20)$$

4) Повторіть кроки з 1 по 3 для розміру зображення з ЦВЗ і зберегти отримані значення R_Z в векторі VR_Z..

5) Потім обчислити середнє значення вектору VR_Z (2.21).

$$\overline{VR_Z} = \frac{\sum_{i=1}^{k} VR_Z^I}{k}, \text{де } k|VR_Z| \quad (2.21)$$

6) Порівняйте елементи вектору VR_Z із середнім значенням $\overline{VR_Z}$ для вилучення пікселів зображення водяного знаку. Якщо значення елемента перевищує середнє значення, витягнутий піксель зображення цифрового водяного

знаку дорівнює 0; в іншому випадку значення пікселя дорівнює 1. Вищеописаний процес описується наступним чином (2.22):

$$E_{BS}(x,y) = \begin{cases} 0, & VR_Z^i > \overline{VR_Z} \\ 1, & \text{інакше} \end{cases}, \text{де } n = |VR_Z| \qquad (2.22)$$

7) Розшифрувати бітовий потік E_{BS}. За допомогою методу арифметичного декодера отримують вихідний бітовий потік D_{BS}.

8) З вихідного бітового потоку D_{BS} витягніть H_{ROI}, H_{EHR}, RH_{ROI}

9) Перетворіть H_{EHR} у відповідне подання бітового потоку як B_{EHR}. Потім змініть B_{EHR} на оригінальне подання як E_{encrypt}.

10) Розшифруйте E_{encrypt} за допомогою K методом ECC, щоб отримати EHR.

11) Аналогічно, обчисліть хеш для H_{ROI}, щоб отримати вихідне зображення.

Висновки до розділу 2

Представлено інформаційну технологію маркування та перевірки автентичності медичних зображень, наведено загальну структуру інформаційної системи, базову послідовність дій, основні алгоритми та процедури нанесення цифрового водяного знаку з додатковими рівнями захисту. Забезпечується шляхом запровадження наведеного алгоритму, який поєднує різні криптографії, стенографії та безпосереднього нанесення ЦВЗ.

Розділ 3

Розробка методів та компонентів для інформаційної технології маркування та перевірки автентичності медичних даних

Умовно ІТ що розробляється можна поділити на 3 основні модулі (рис. 3.1):

– Обробка та аналіз вхідних даних
– Шифрування вхідних даних
– Перевірка автентичності (дешифрування)

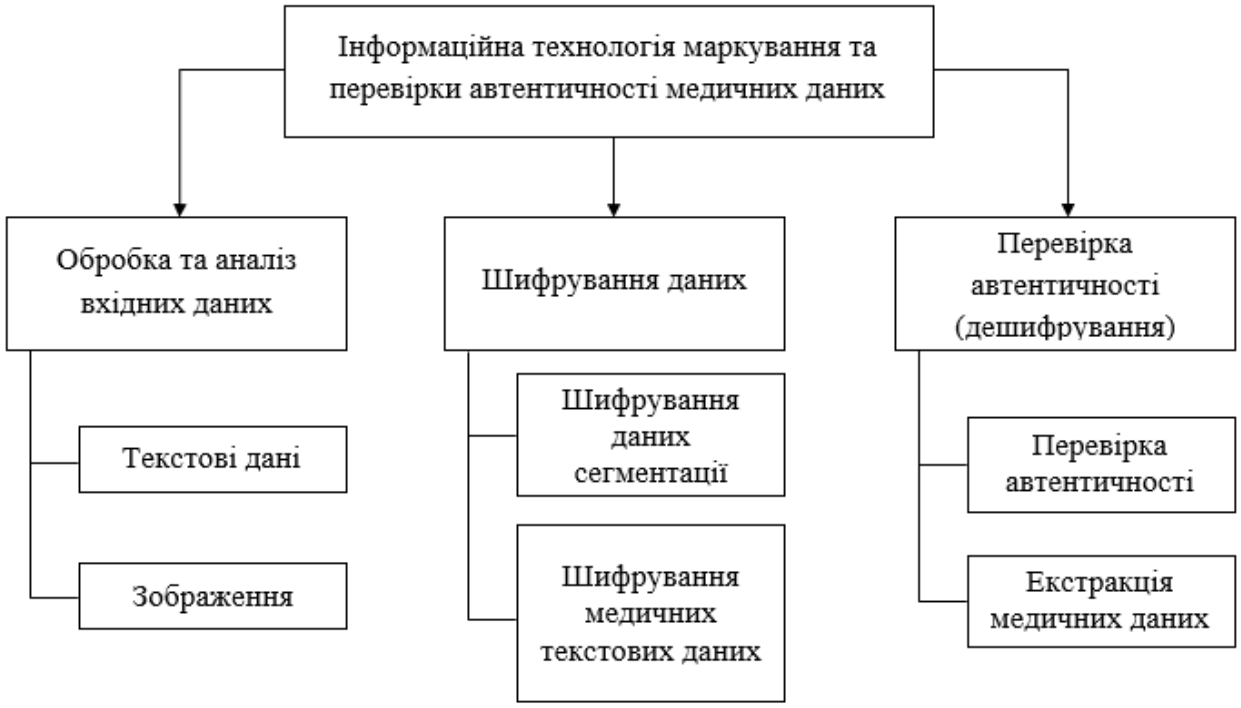


Рисунок 3.1 – Основні модулі інформаційної технології маркування та перевірки автентичності медичних даних

3.1 Розробка методів обробки вхідного медичного зображення інформаційної технології маркування та перевірки автентичності медичних зображень

Початок роботи з інформаційною системою починається з безпосереднього завантаження зображення в систему користувачем. Після завантаження зображення в систему постає задача його подальшого опрацювання. Інтерфейс вхідного зображення можна представити наступним чином:

Рисунок 3.1 – Інтерфейс вхідного зображення

В представленому інтерфейсі його характеристики можна представити наступним чином:

- «Name» – назва зображення
- «Width» – ширина зображення в пікселях
- «Height» – висота зображення в пікселях

– «Size» – розмір зображення в кілобайтах
– «Grayscale» – метод для перетворення вхідного зображення в чорно–біле

Метод «Grayscale» приймає безпосередньо інтерфейс «Image» та виконує відповідні перетворення з вхідним зображенням, які можна представити наступним чином (рис 3.2).

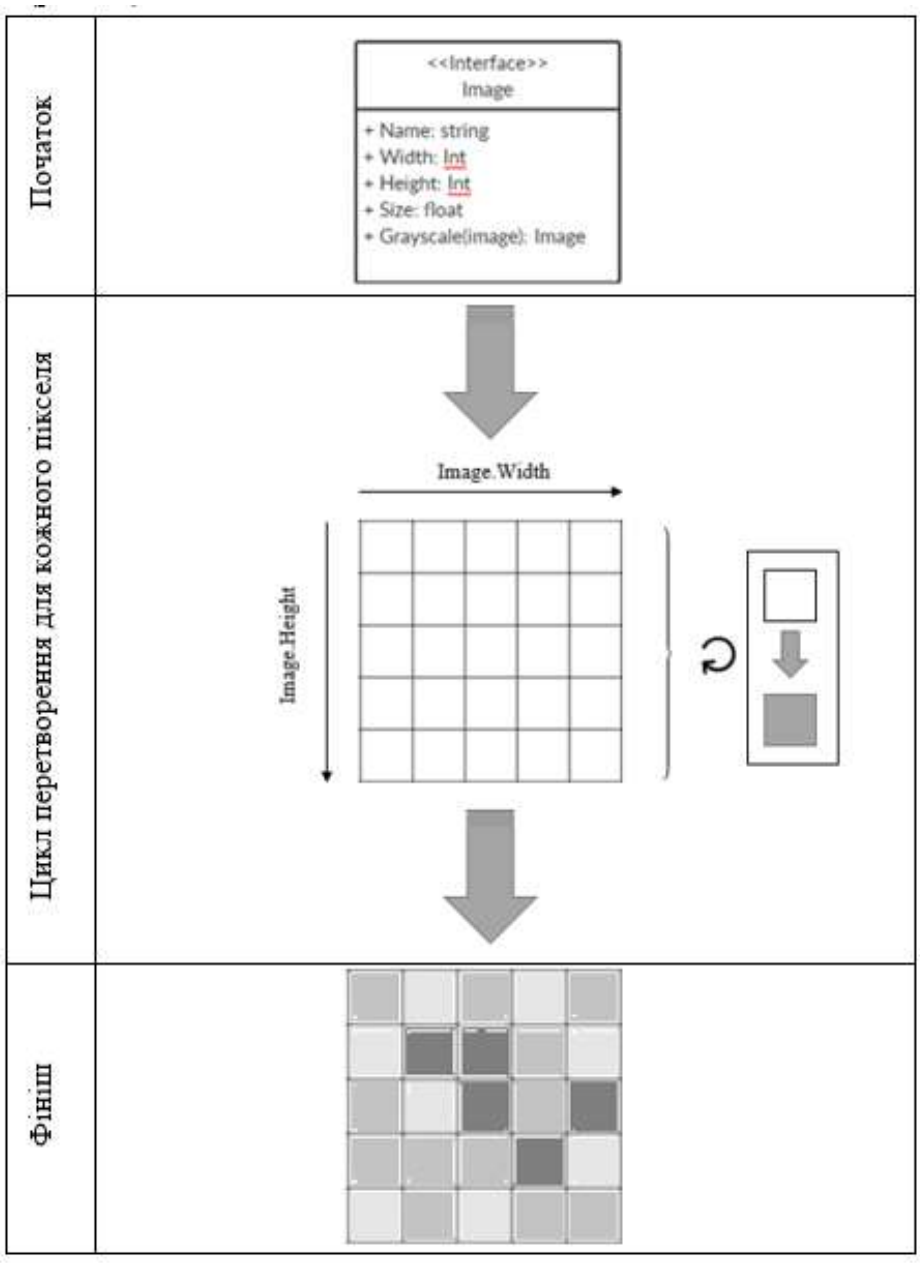


Рисунок 3.2 – Схематичне представлення роботи методу «Greyscale»

Тобто з інтерфейсу зображення використовуються відповідні параметри «Image.Width» та «Image.Height» для їх використання в якості «границь» для відповідного циклу перетворення. Безпосередньо цикл перетворення змінює вхідний колір на сірий з відповідним рівнем яскравості (в залежності від вхідного кольору).

Отримавши зображення, яке є результатом роботи функції «Grayscale», проводиться сегментація (рис 3.2).

Отримавши зображення в результати методу «Greyscale» виконується процес сегментації методом зростаючих регіонів.

Інтерфейс отриманих сегментованих даних представлено на рисунку 3.3, з відповідними значеннями:

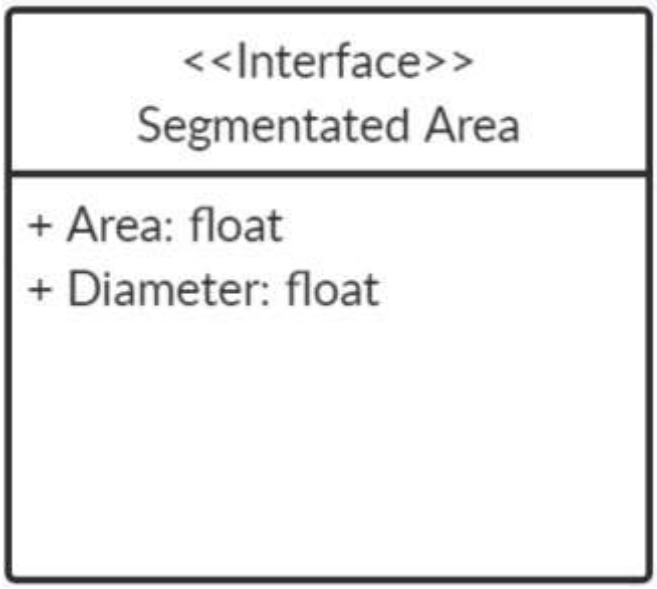


Рисунок 3.3 – Інтерфейс результату процесу сегментації (дані сегментації)

Відповідний процес сегментації можна представити наступним чином (рис. 3.4):

1) Отримується вхідне чорно–біле зображення
2) Виконується аналіз пікселів зображення
3) Виконується C_{linear} перетворення
4) Отримуються дані сегментації у відповідний інтерфейс (рис. 3.3)

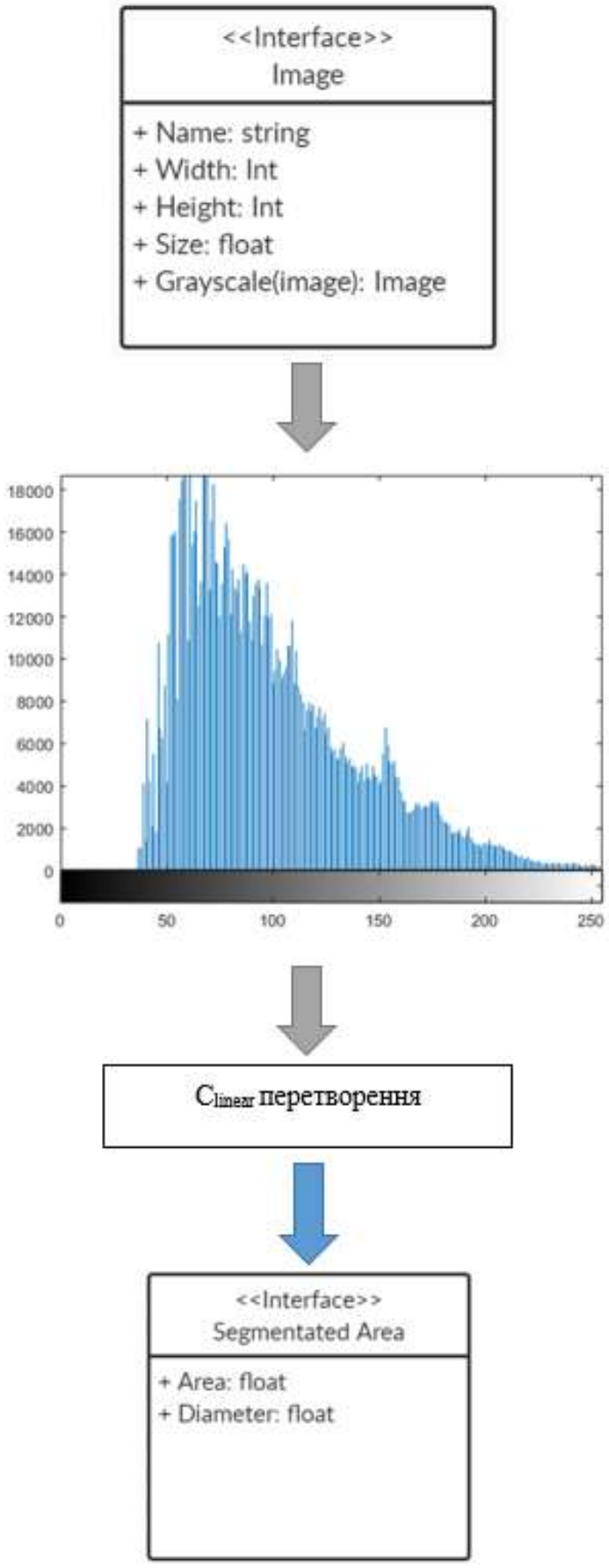


Рисунок 3.4 – Схема процесу отримання даних сегментації

3.2 Розробка методів опрацювання вхідних медичних даних користувача

Дані, які отримані з MIC можуть бути різними відповідно до потреб користувачів чи медичного закладу. В відповідній IT інтерфейс отриманих даних виглядає наступним чином (рис. 3.5):

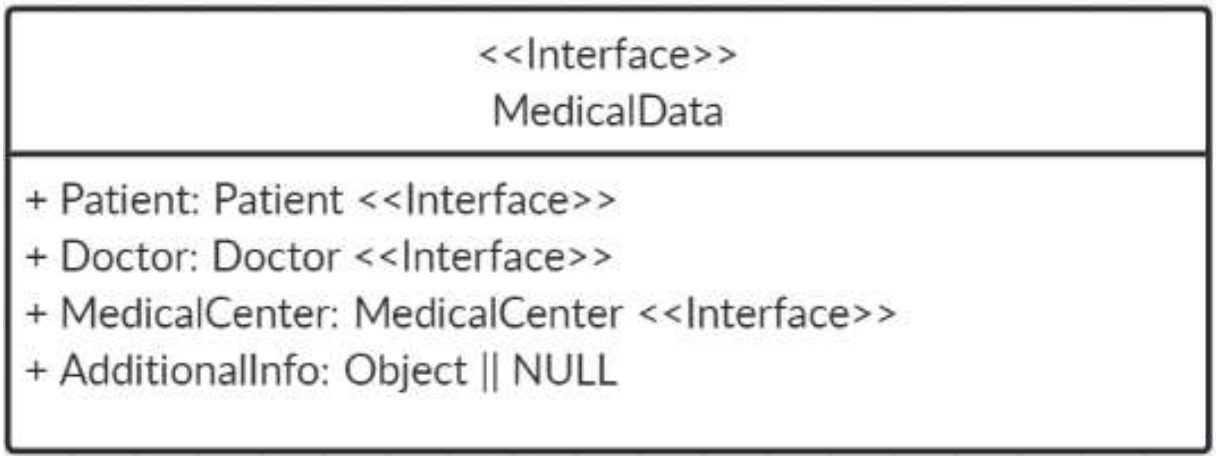


Рисунок 3.5 – Інтерфейс вхідних даних

– Поле «Patient» – виражено інтерфейсом «Patient» (рис. 3.6), який унаслідується від інтерфейсу «User» (рис 3.7);

– Поле «Doctor» – виражено інтерфейсом «Doctor» (рис 3.8);

– Поле «MedicalCenter» – виражено інтерфейсом «MedicalCenter» і представлено на рис 3.9;

– Поле «AdditionalInfo» – може бути довільним об’єктом з додатковими даними, або null.

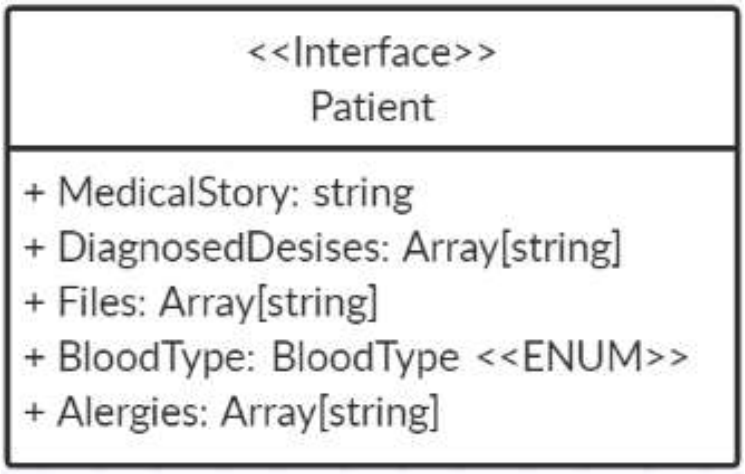


Рисунок 3.6 – Інтерфейс пацієнта

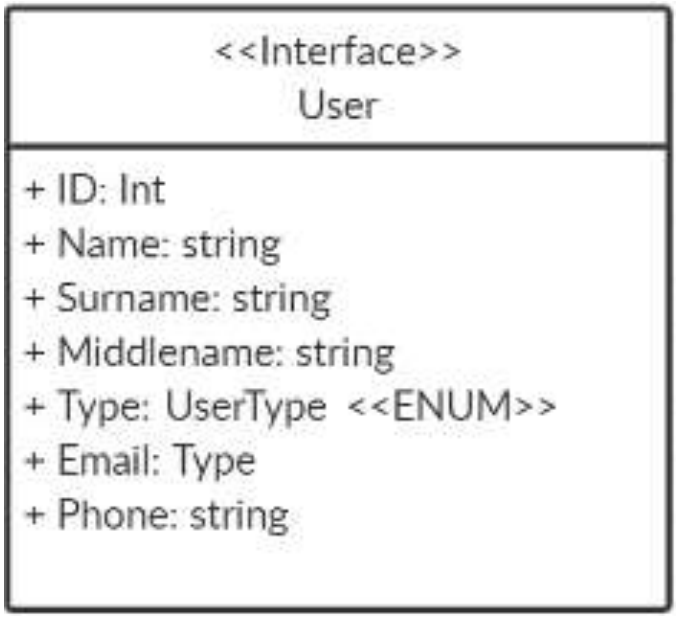


Рисунок 3.7 – Інтерфейс користувача

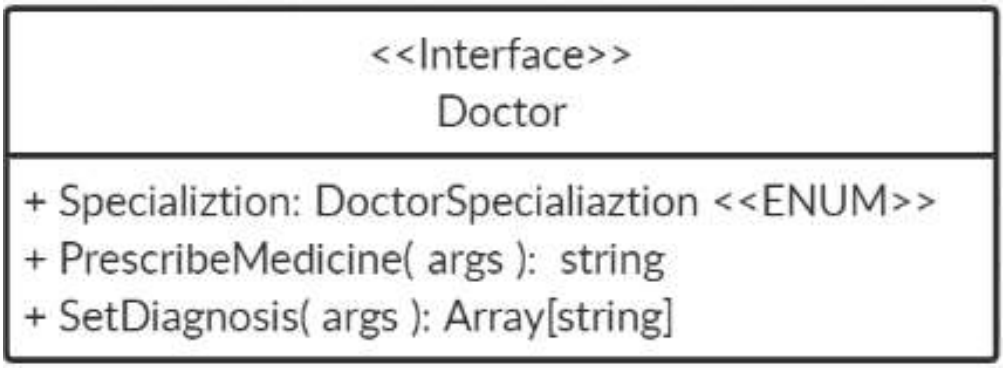


Рисунок 3.8 – Інтерфейс лікаря

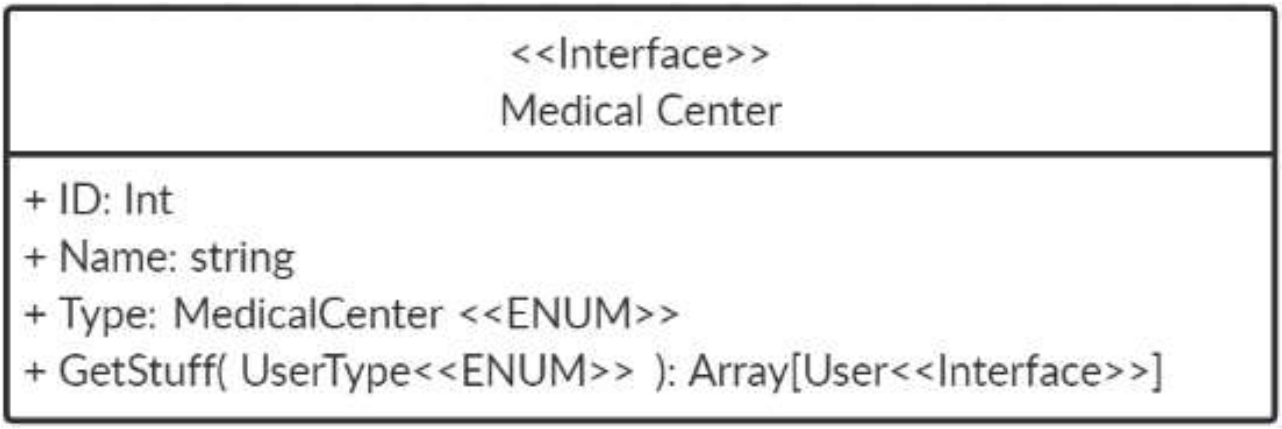


Рисунок 3.9 – Інтерфейс медичного закладу

Взаємозв’язки між основними інтерфейсами можна представити наступним чином (рис 3.10):

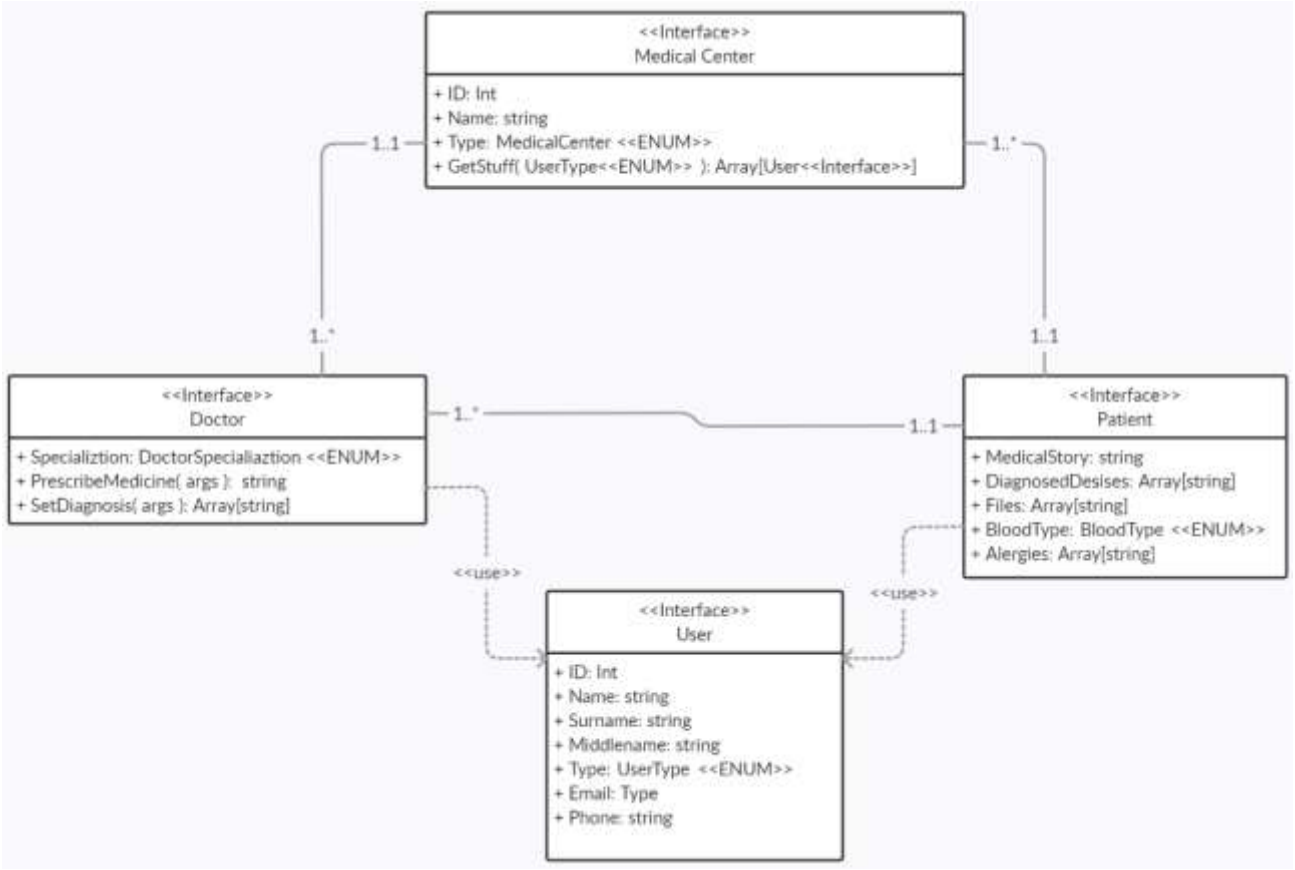


Рисунок 3.10 – Схема взаємозв’язків основних інтерфейсів

3.3 Розробка методів шифрування для вхідних медичних даних та зображень

При роботі з шифруванням даних перш за все генерується закритий ключ, який зберігається не сервері для дешифровки вхідних зашифрованих даних та відкритий ключ для користувача за допомогою якого відбувається «спілкування» по схемі «клієнт–сервер».

Для шифрування даних сегментації (рис 3.4) потрібно перетворити дані інтерфейсу в дані типу «string» та використати відповідний алгоритм шифрування SHA–256 (рис. 3.11).

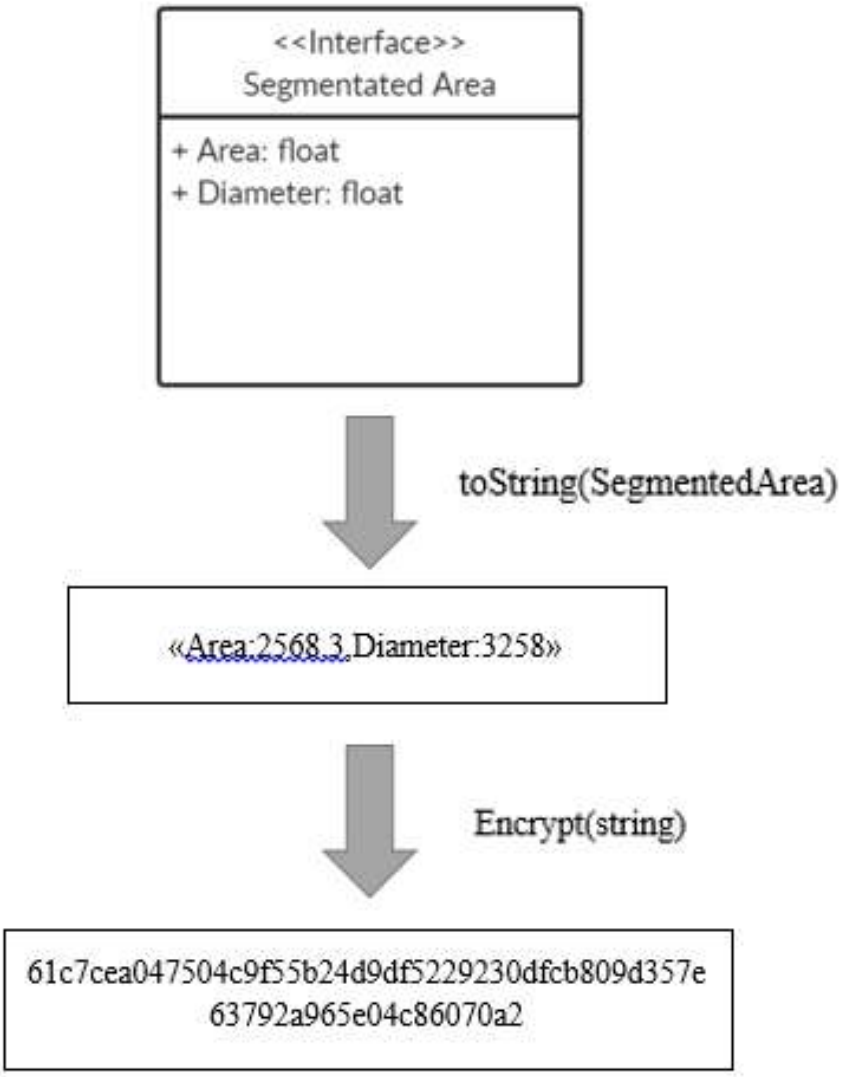


Рисунок 3.11 – Схематичне зображення роботи алгоритму шифрування SHA–256

Для шифрування медичних даних використовується еліптична криптографія. Представити цей процес в системі можна наступним чином (3.12):

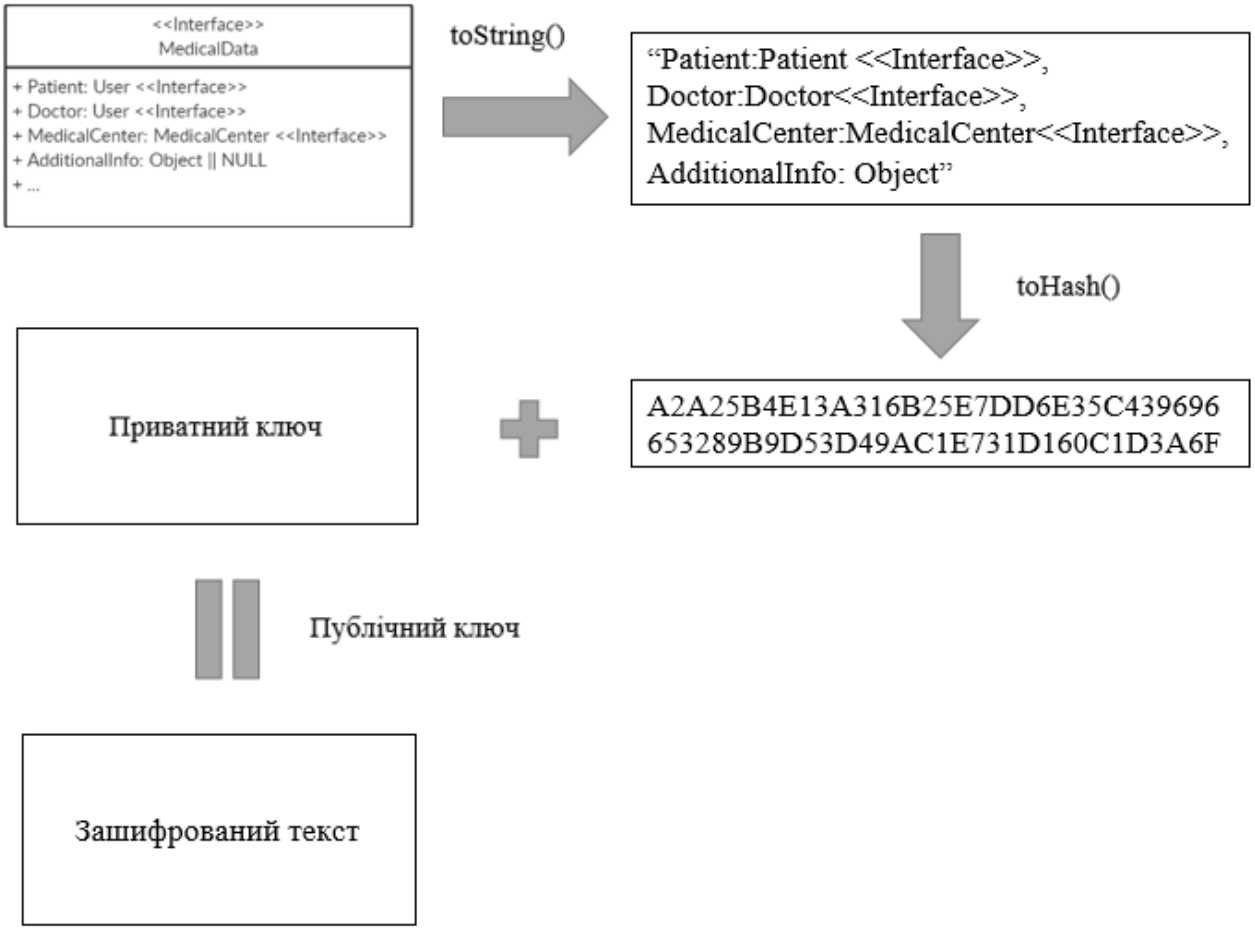


Рисунок 3.12 – Схематичне зображення роботи еліптичної криптографії

3.4 Розробка методу нанесення цифрового водяного знаку на медичне зображення

Для нанесення ЦВЗ використовується гібридний метод нанесення оснований на основі отриманих зашифрованих даних, цей процес можна представити наступним чином (рис 3.13).

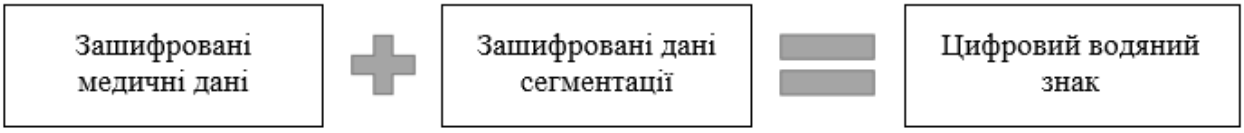


Рисунок 3.13 – Процес нанесення ЦВЗ

3.5 Розробка методів перевірки автентичності

Метод перевірки автентичності виступає в ролі валідатору для безпосередньо вхідного зображення, та якщо зображення автентичне, відображає «вшиту інформацію» в ЦВЗ у системі. Алгоритм роботи методу перевірки автентичності можна представити як (рис. 3.14):

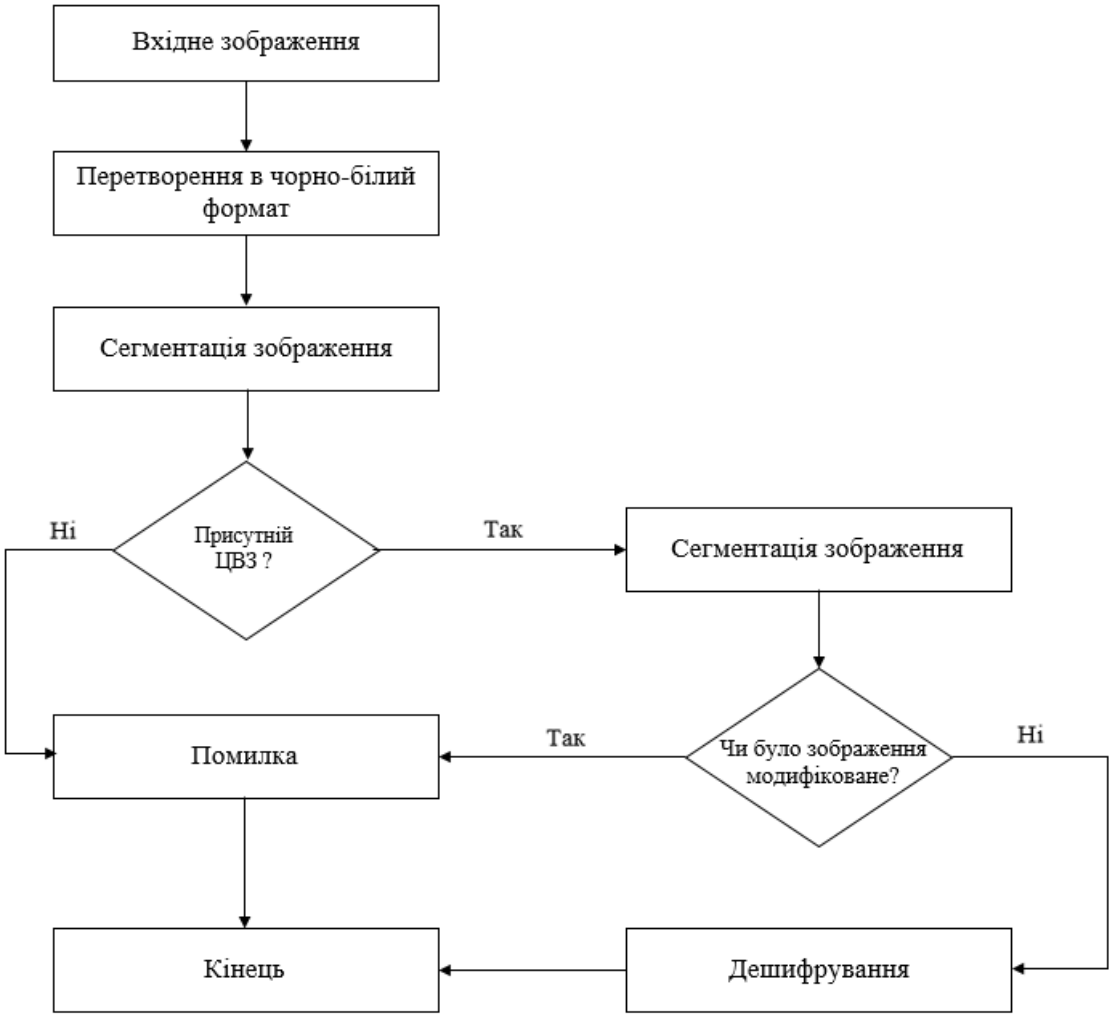


Рисунок 3.14 – Алгоритм роботи методу перевірки автентичності

При роботі методу дешифрування початкові кроки такі ж, як при нанесенні ЦВЗ. Далі відбувається перевірка наявності ЦВЗ, якщо перевірка виявила його присутність відбувається сегментація вхідного зображення, результати сегментації перевіряються з даними в ЦВЗ. Якщо дані співпадають, отже зображення автентичне, адже область інтересів не зазнала змін. Після цього зашифровані дані можна зчитати.

Висновки до розділу 3

Представлено основні інтерфейси та алгоритми для маркування та перевірки автентичності медичних даних. Детально описані поля відповідних інтерфейсів та відповідні методи. Наведені схеми роботи алгоритмів та оптимальні їх структури, послідовність виконання функцій.

Розділ 4

Дослідження ефективності інформаційної технології маркування та перевірки автентичності медичних зображень

4.1 Розробка інформаційної технології маркування та перевірки автентичності медичних зображень

Відповідно до побудованих алгоритмів у розділі три та опираючись на дані розділу один та розділу два було реалізовано відповідна інформаційна технологія.

Відповідна ІТ (веб застосунок) була розроблена за допомогою мови «Typescript» та фреймворку «Angular», що дозволило задовільнити такі вимоги як:

- швидкодія;
- кросплатформність;
- можливість інтегрування в існуючі ІТ;
- доступність.

Тобто веб застосунок може бути запущений на будь–якому пристрої який дозволяє відкривати веб посилання та підтримує нові веб стандарти.

На початку роботи з веб застосунком потрібно виконати вхід у систему відповідними логіном та паролем (рис. 4.1).

Наразі у системі існує 2 ролі «лікар» («Doctor») та «пацієнт» («Patient»). В залежності від відповідних ролей, змінюється функціонал та контент застосунку (пункти меню, тексти, посилання, кнопки та інші інтерактивні елементи.

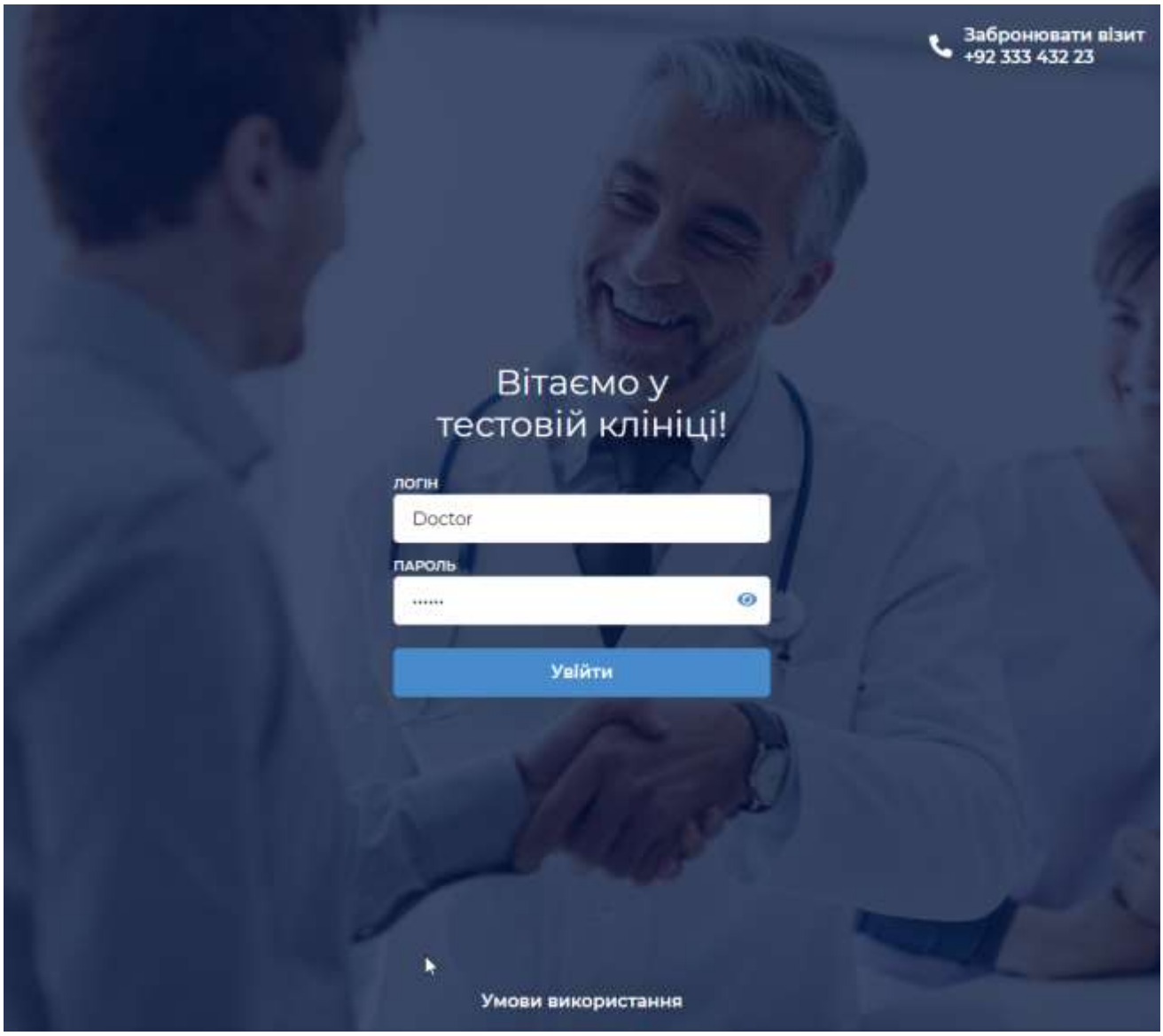


Рисунок 4.1 – Екран входу в систему

Інтерфейс користувача можна умовно розділити на такі три зони (рис. 4.2):

1) Бокове меню
2) Верхня навігація
3) Основний контент

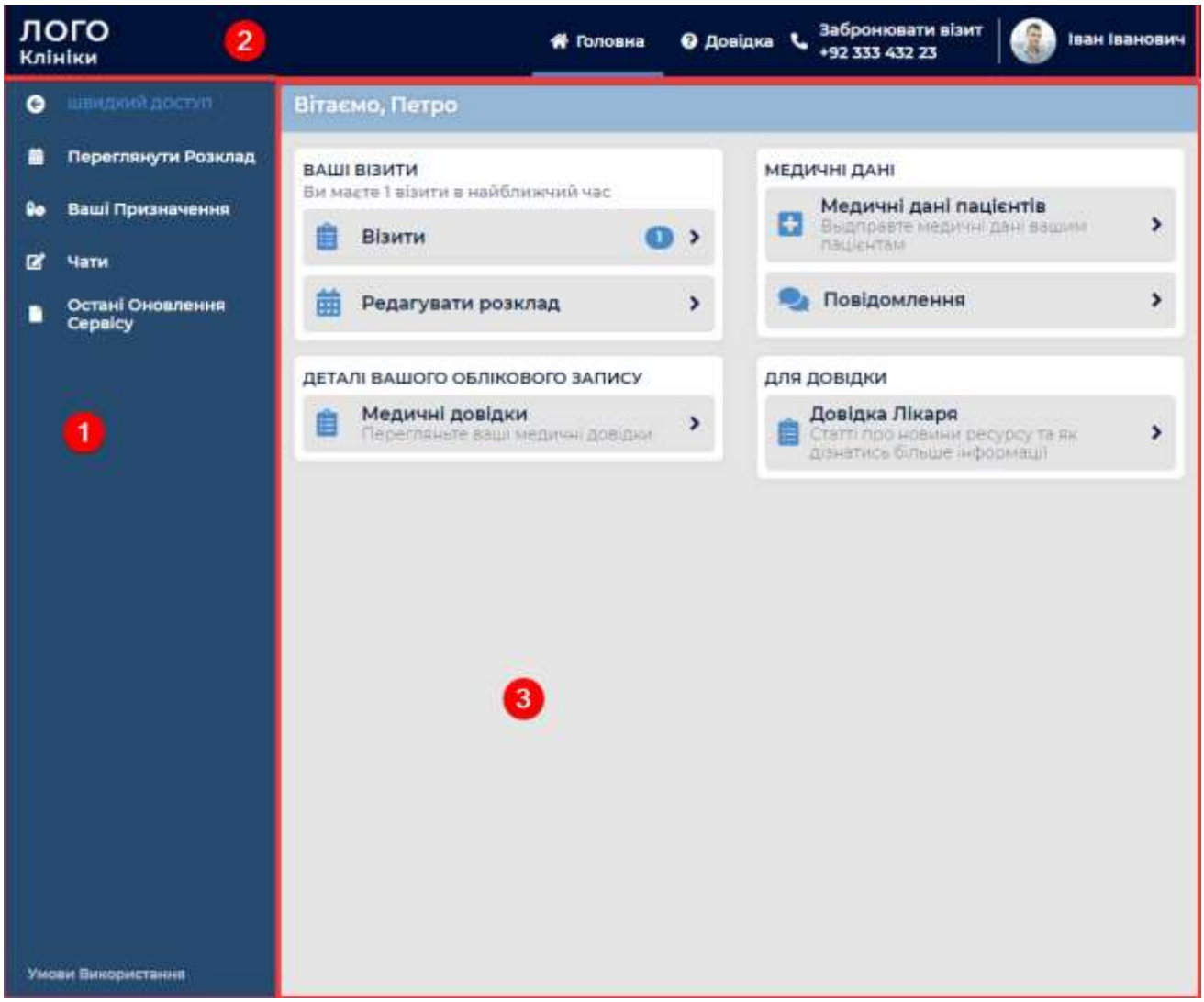


Рисунок 4.2 – Інтерфейс користувача лікаря

Для користувача «Пацієнт» цей екран буде виглядати аналогічно, з несуттєвими змінами відповідно до ролі користувача. У даному інтерфейсі функціонал маркування та перевірки автентичності медичних зображень представлено у блоці «Медичні даних».

У випадку, якщо користувач має роль «Лікар» («Doctor»), то у цьому випаду йому потрібно перейти за посиланням «Медичні дані пацієнтів» (рис. 4.3).

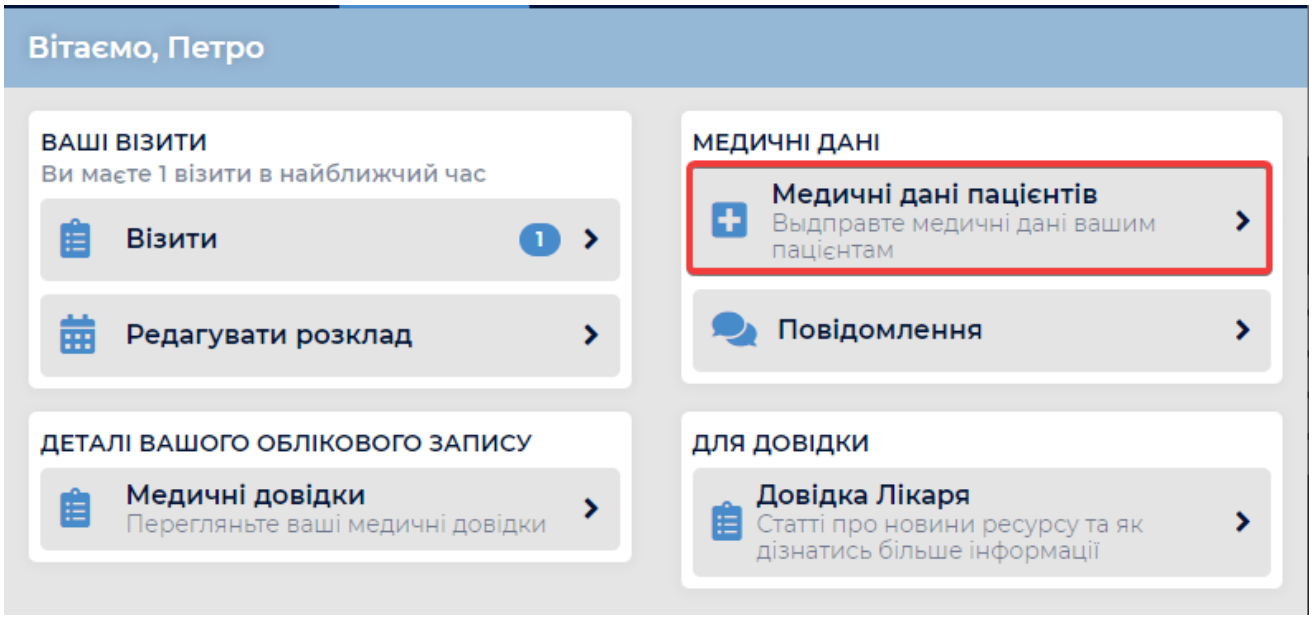


Рисунок 4.4 – Посилання на екран «Медичні дані пацієнта»

У випадку якщо роль користувача «Пацієнт» («Patient»), то в цьому випадку відповідне посилання буде виглядати наступним чином (рис. 4.5):

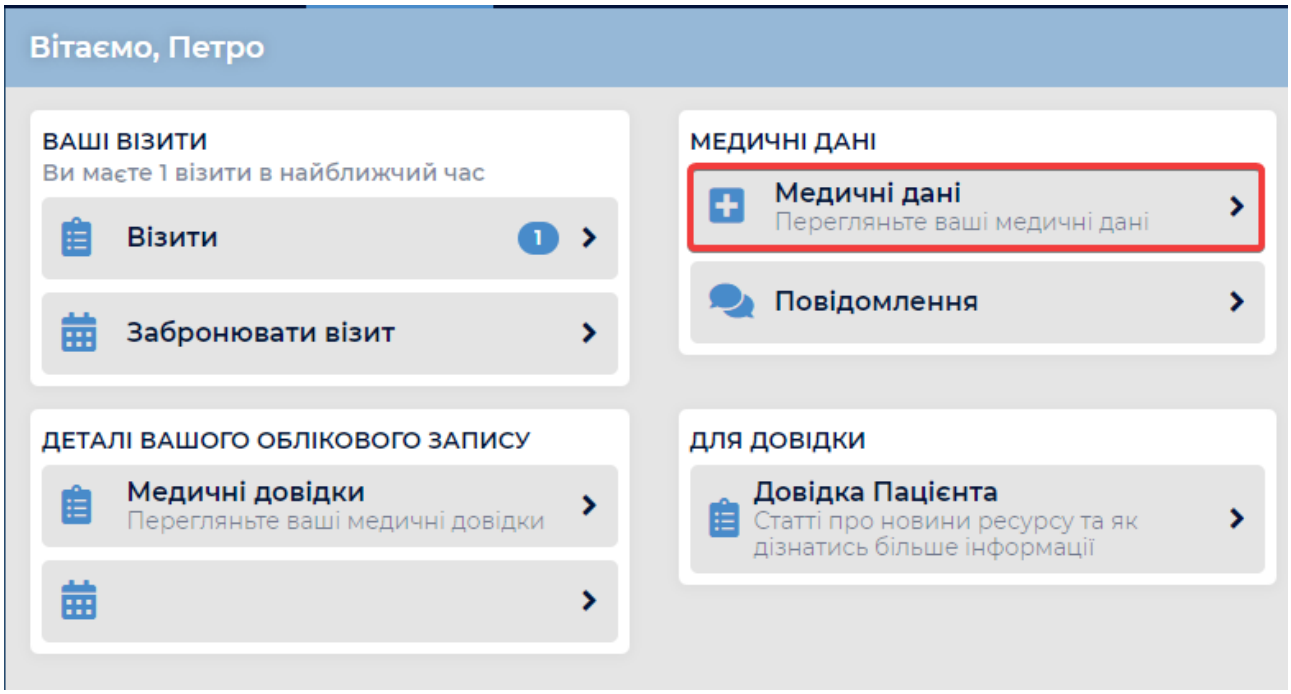


Рисунок 4.5 – Посилання на екран «Медичні дані» користувача

Якщо теперішній користувач системи має роль «лікар», то при переході на екран завантаження даних (зображення) перед безпосереднім їх завантаженням «лікарю» потрібно обрати пацієнта (рис. 4.6), потім натиснути на кнопку «Додати зображення» (рис. 4.7).

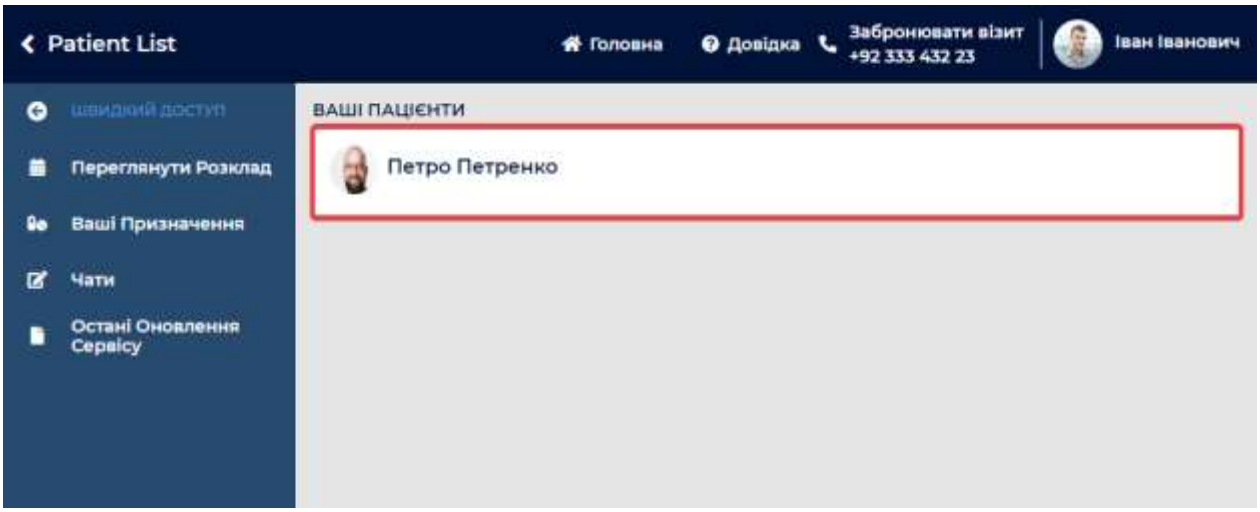


Рисунок 4.6 – Список пацієнтів лікаря

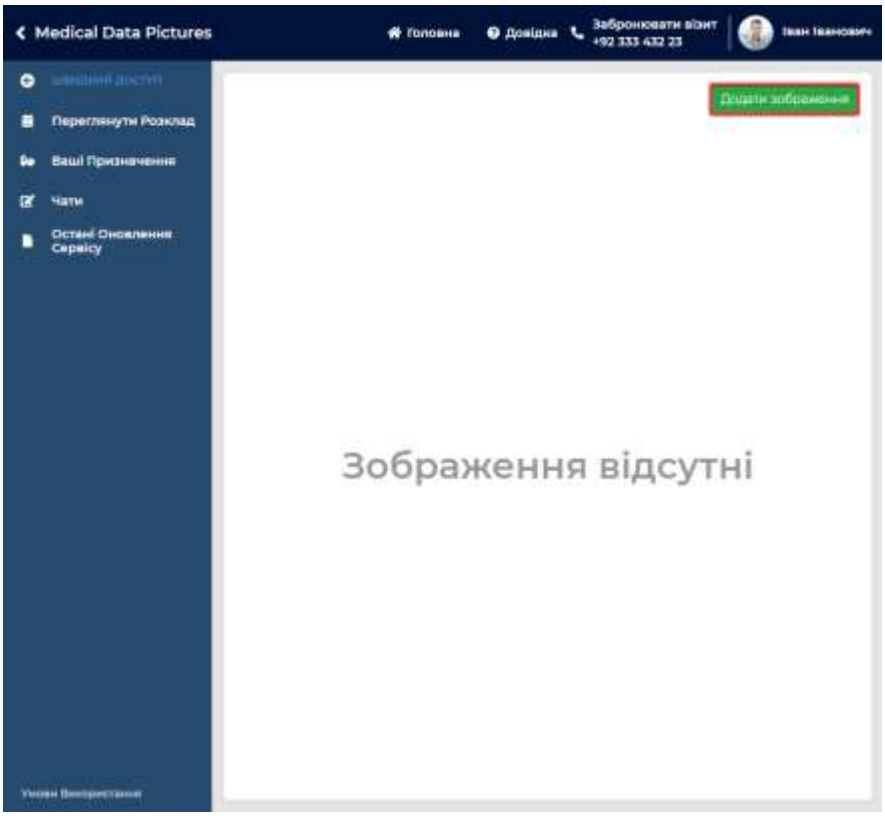


Рисунок 4.7 – Екран медичних зображень

Екран зображення для користувача з роллю «Пацієнт» виглядає ідентично до рисунку 4.7 за виключенням кнопки «Додати зображення».

Після натиску кнопки відбувається завантаженні вхідного зображення «Лікарем». Виконується перетворення зображення на чорно–біле (рис. 4.8).

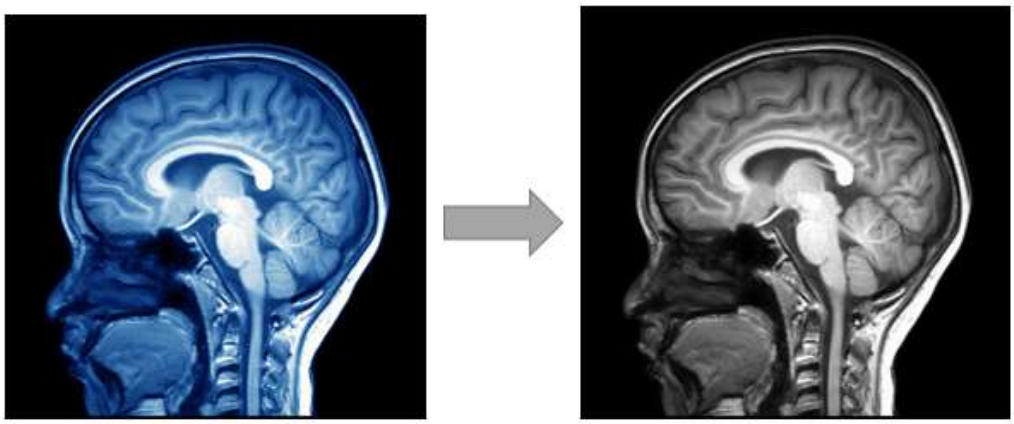


Рисунок 4.8 – Приклад перетворення кольорового зображення в чорно–біле (а) – вхідне зображення; (б) – результат перетворення

Далі методом росту регіонів визначається область інтересів (рис. 4.9):

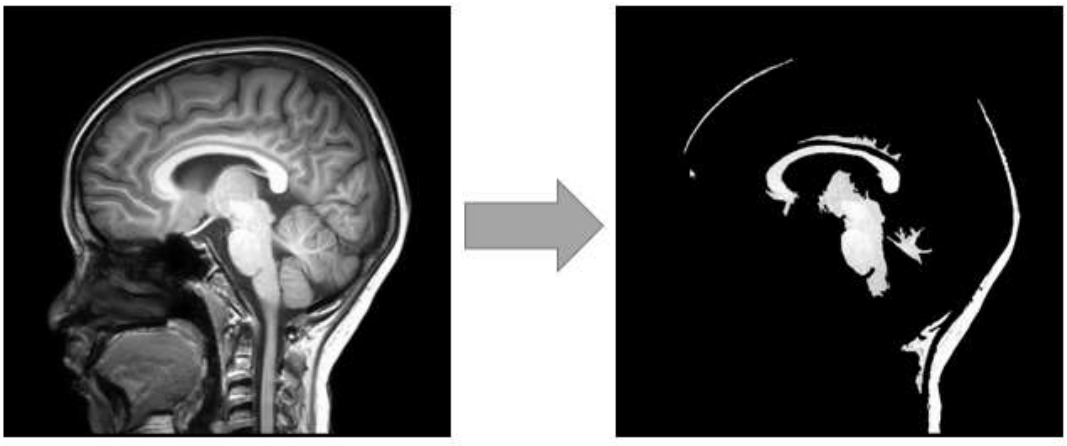


Рисунок 4.9 – Приклад роботи методу росту регіонів (а) – вхідне чорно–біле зображення; (б) – області інтересів отримані в результаті роботи алгоритму.

Після визначення області інтересів, відповідні дані хешуються алгоритмом SHA–256 (рис. 4.10)

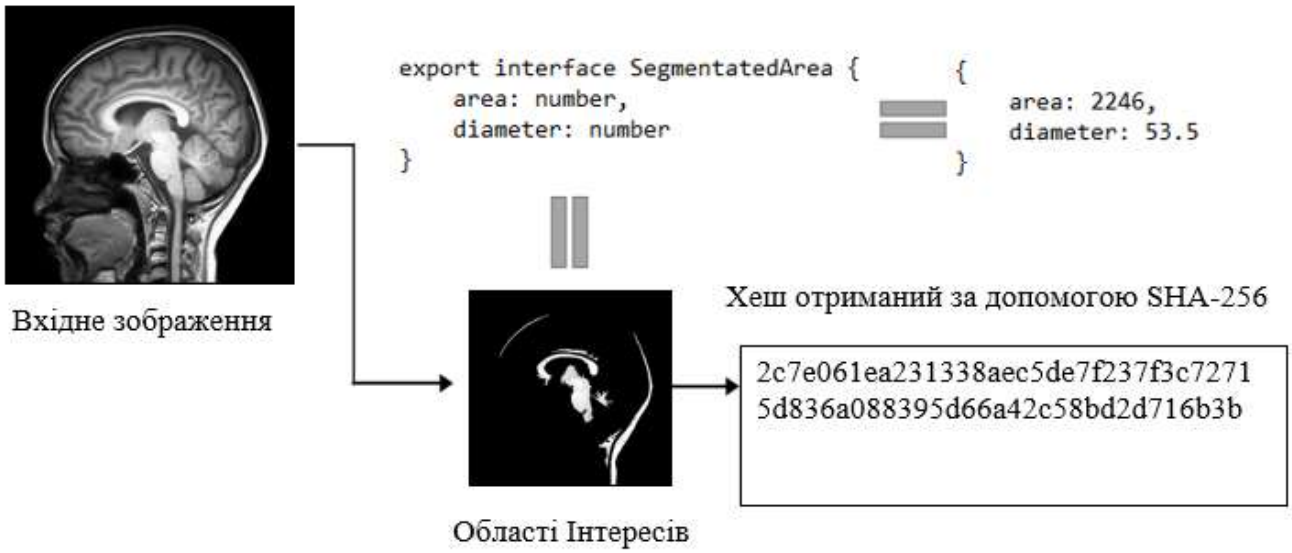


Рисунок 4.10 – Алгоритм роботи SHA–256

Отримавши хеш області інтересів, далі отримуються медичні дані (рис. 4.11), а саме:

– Дані «Лікаря»
– Дані «Пацієнта»
– Медичний центр
– Додаткова інформація (опціонально)

```
export interface MedicalData {
    patient: Patient,
    doctor: Doctor,
    medicalCenter: MedicalCenter,
    additionalInfo?: object
}
```

Рисунок 4.11 – Інтерфейс медичних даних

Отримані дані хешуються та конкатенуються з даними областей інтересів та наносяться на зображення.

Після нанесення відповідного ЦВЗ зображення додається у систему (рис 4.12).

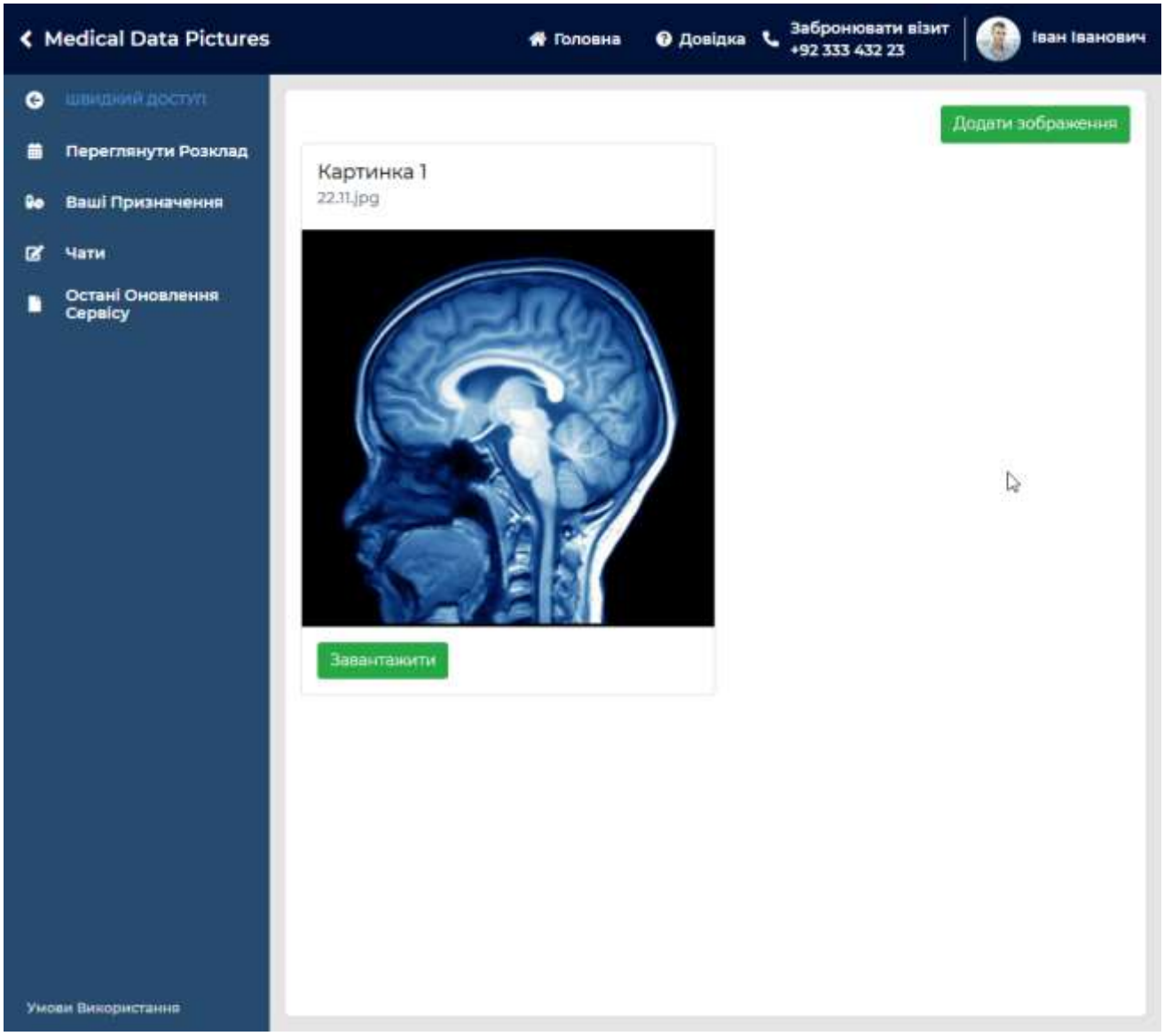


Рисунок 4.12 – Екран медичних зображень після додавання зображення

Після того, як зображення завантажено у систему на цьому етап нанесення ЦВЗ є закінченим. Зображення, яке було завантажено у систему має містити ЦВЗ з відповідними даним. Якщо зображення при перевірці на стороні користувача («Пацієнта»), не містить ЦВЗ, або цифровий водяний знак є пошкодженим, то відповідне зображення не є автентичним.

Процес перевірки зображення для «Пацієнта» виглядає наступним чином:

Відкрити екран медичних зображень, та на вхідному зображені натиснути кнопку «перевірка автентичності» (рис. 4.13).

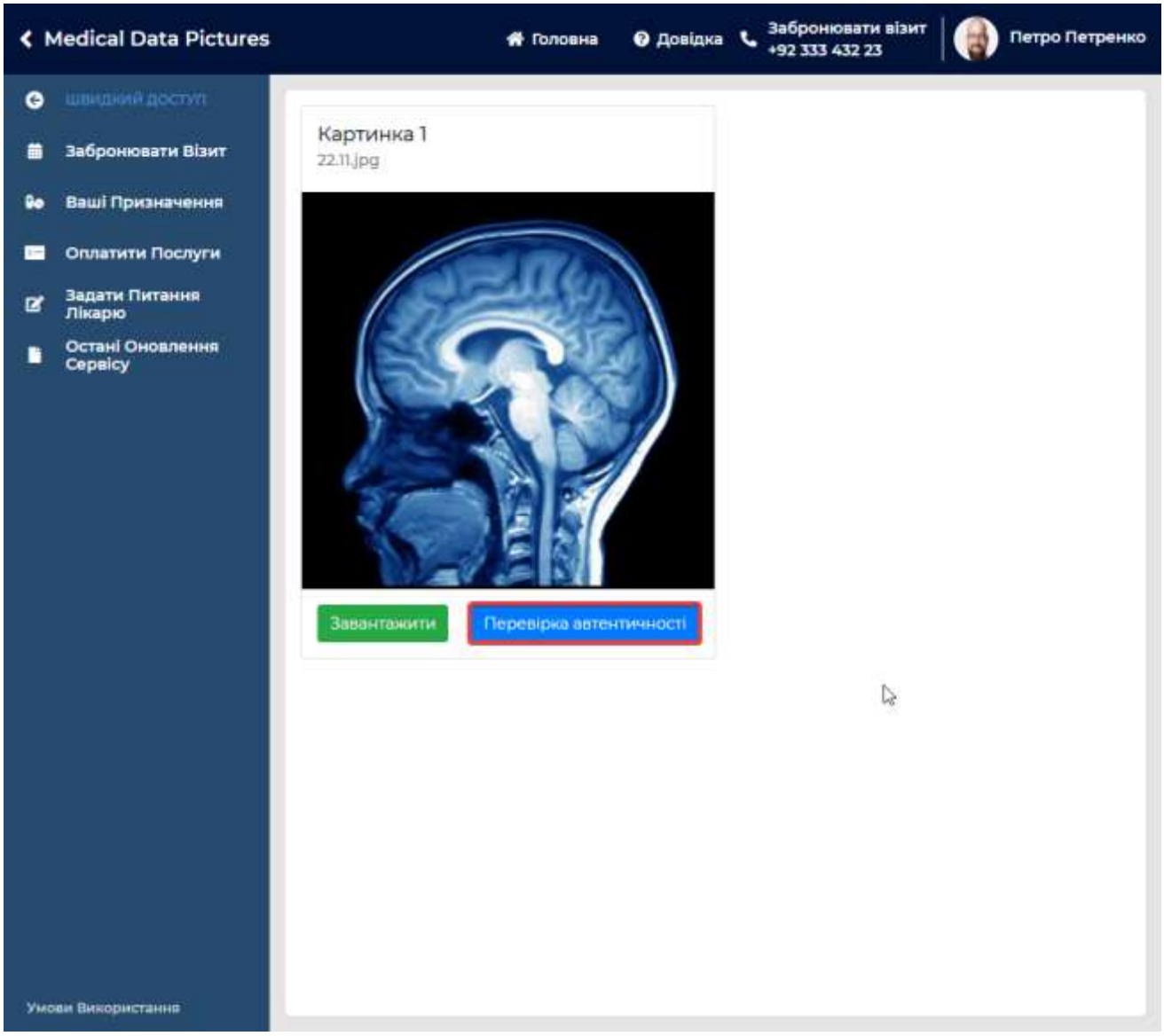


Рисунок 4.13 – Екран зображень користувача «Пацієнт» з кнопкою «перевірка автентичності»

Після того як кнопка перевірка автентичності була натиснута, то починається процес вилучення ЦВЗ на стороні клієнта описаного в розділі 2.9.

Після того процес завершується, система видає позитивний (рис. 4.14) або негативний результат (рис 4.15), тобто чи є зображення автентичним чи ні.

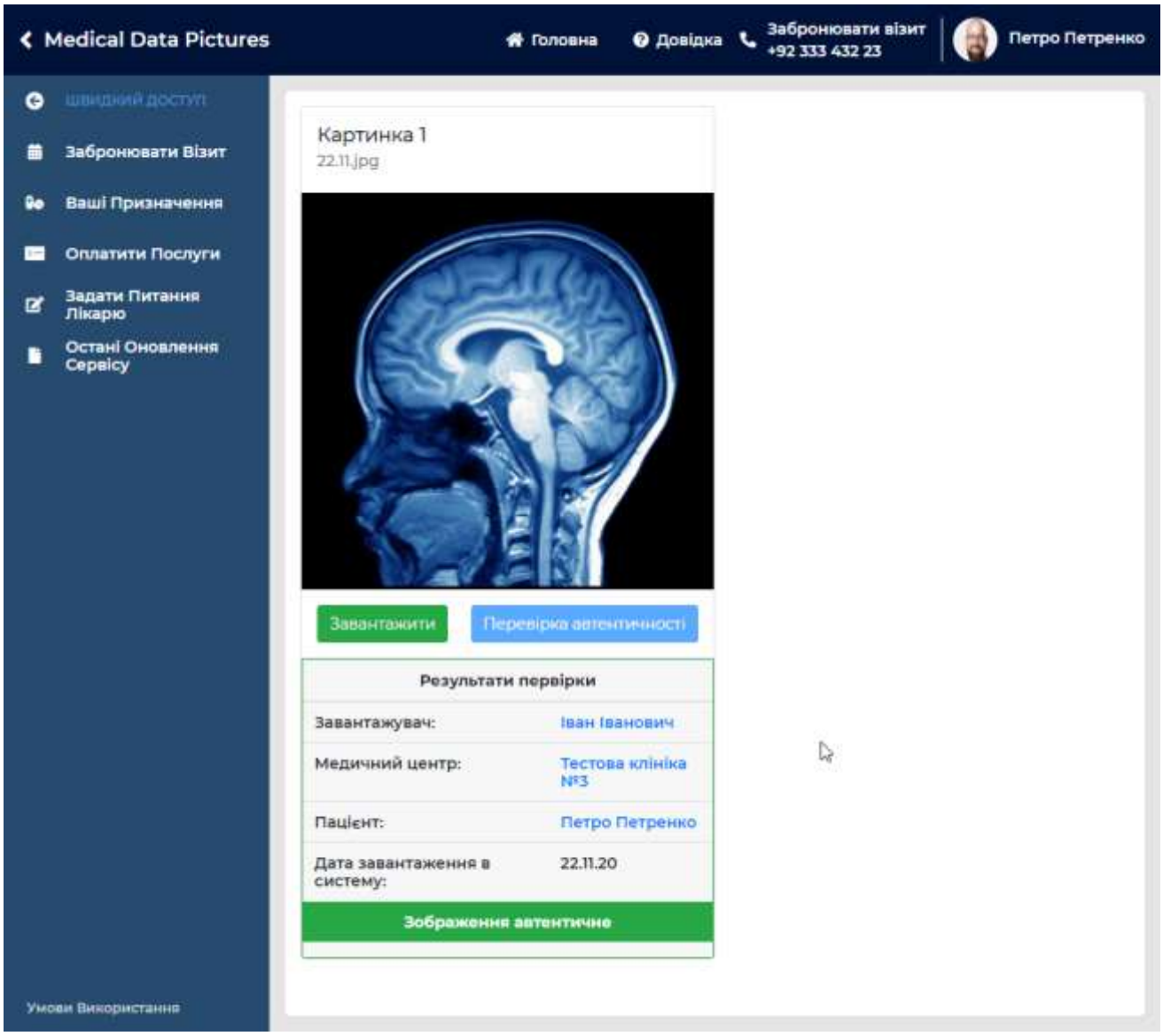


Рисунок 4.14 – Приклад позитивної перевірки вхідного зображення на автентичність

У разі отримання негативного результату рис (4.15), в наслідку наступних випадків:

- Пошкодження ЦВЗ;
- модифікація зображення;
- підміна зображення;
- тощо;

Користувач отримує повідомлення про те, що зображення не є автентичним і потрібно звернутись у підтримку системи, яка наддасть подальші інструкції.

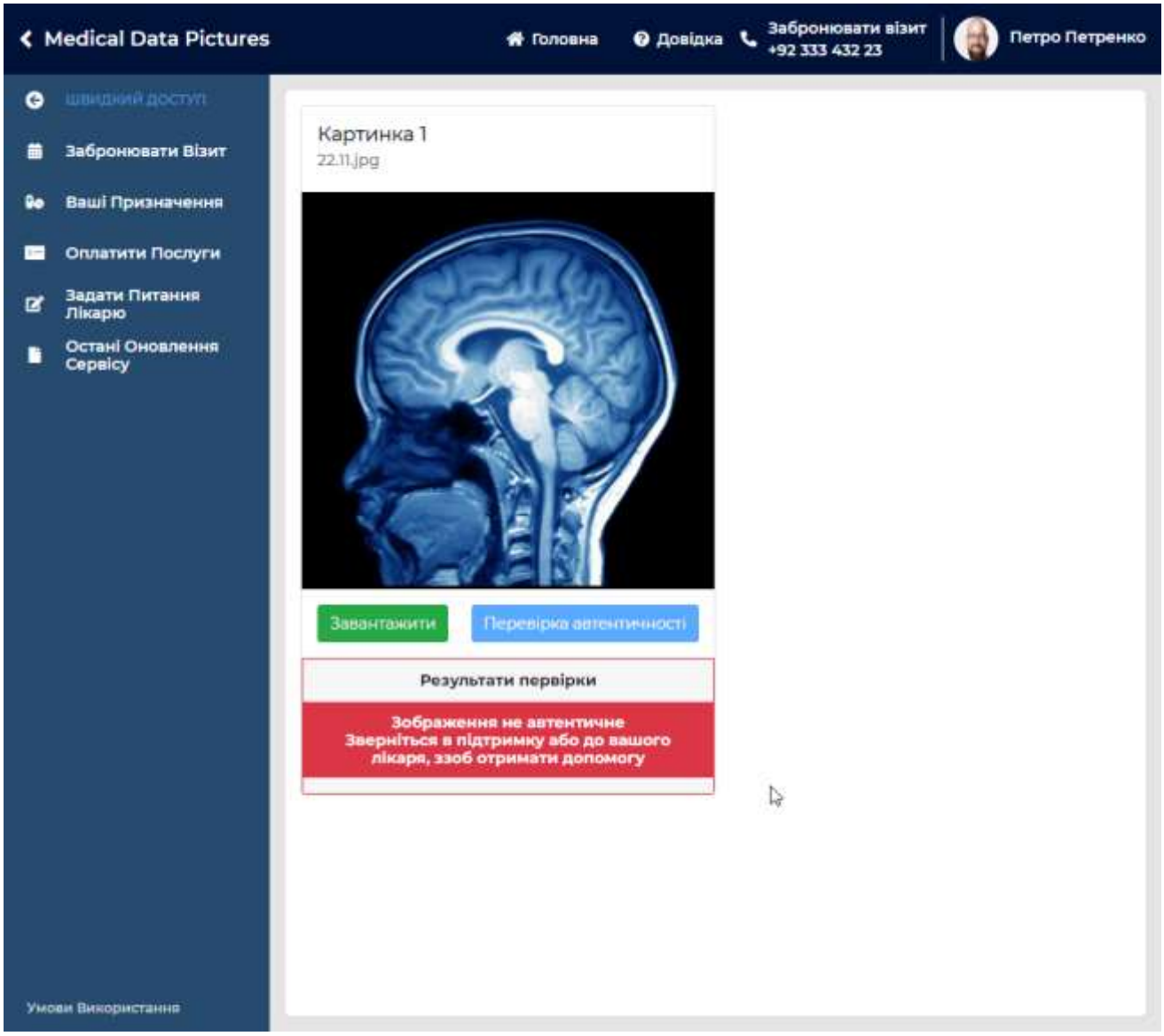


Рисунок 4.15 – Приклад невдалої перевірки (зображення не є автентичним)

На основі отриманих даних (при позитивному результаті), користувачу «Пацієнт» показується відповідний набір даних (лікаря, медичного центру, тощо) вилучені з ЦВЗ та відповідно помітка, що дане зображення є автентичним.

4.2 Дослідження функціональності інформаційної технології. Перевірка запропонованого підходу на надійність, ефективність та його можливості

При перевірці відповідного функціоналу та ефективності використано п’ять типів зображень пухлини головного мозку (рис.4.16) та відповідні медичні дані користувача. Приклад медичних даних, використаний при аналізі, наведено на рисунку 4.17.

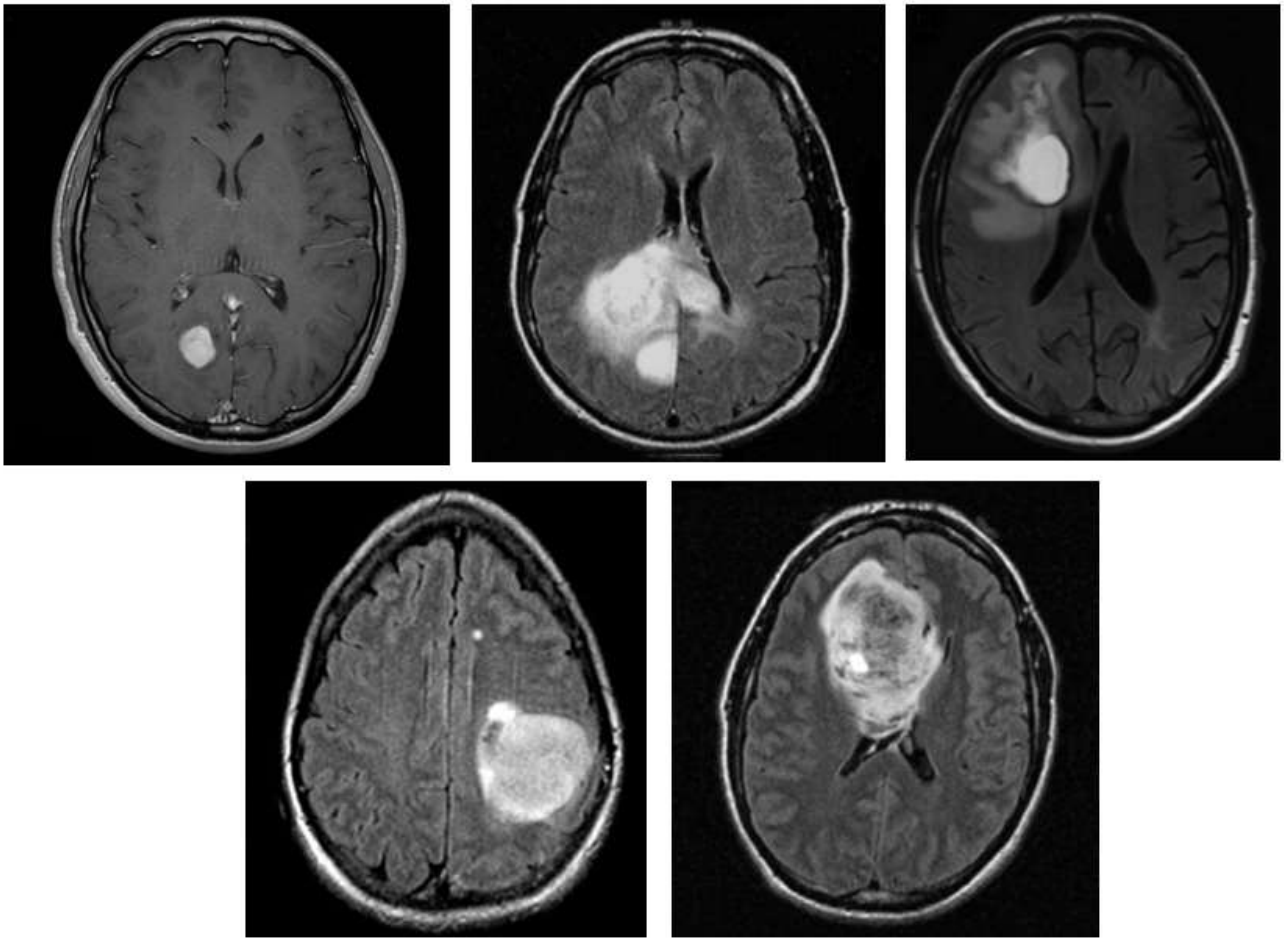


Рисунок 4.16 – Приклад зображень використаних для аналізу

```
Patient: Петро Петренко, PatientID: 43252124,
Doctor: Іван Іванович, DoctorID: 231231,
MedicalCenter: Тестова клініка,
AdditionalInfo: Тест
```

Рисунок 4.17 – Вхідні медичні дані для тестування

Алгоритм нанесення представлено наступним чином:

Вхідні дані: Оригінальне зображення I, Медичні дані, секретний ключ K_S.

Вихідні дані: Зображення з водяним знаком I_W.

Алгоритм:

1. Завантажити зображення
2. Використати алгоритм RG (region growing) для сегментації області інтересів.
3. Використати SHA–256 для обчислення хешу області інтересів.
4. Використати алгоритм ECC (еліптична криптографія) для шифрування медичних даних
5. Використати процес об’єднання для об’єднання зображення та інформації про EHR
6. Отримати результат

Оцінка запропонованої методики проводиться з використанням наступних показників:

– Пікове співвідношення сигналу до шуму (peak signal–to–noise ratio або PSNR)

– Нормалізована кореляція (Normalized Correlation або NC)

PSNR використовується для вимірювання якості зображення з нанесеним водяним знаком. PSNR – це співвідношення між вихідним зображенням та зображенням із водяним знаком. PSNR ідентифікується з використанням середньоквадратичної помилки (MSE). MSE дає сукупну похибку в квадраті між

пошкоджуючим шумом та максимальною потужністю сигналу. Вищі значення PSNR і нижчі значення MSE означають хорошу якість нанесення водяного знаку.

NC вимірює схожість між вихідним зображенням та зображенням з нанесеним водяним знаком, виділеним із атакованого зображення.

У запропонованому методі нанесення цифрового водяному знаку гібридизуються методи стиснення та шифрування даних без втрат, щоб вбудувати медичні дані (EHR) та хеш зображення в медичні зображення.

Гібридний підхід підвищує безпеку даних, що передаються. Спочатку сегментується частина області інтересів (ROI) із зображення за допомогою алгоритму росту регіонів (RG). Потім шифрується ROI за допомогою SHA–256 і шифруємо EHR, використовуючи алгоритм ECC.

Після цього вбудовується нанесення ЦВЗ на медичне зображення. На рисунку 4.18 показані вхідні зображення та зображення з водяними знаками, а на рисунку 4.19 – експериментальні результати сегментації.

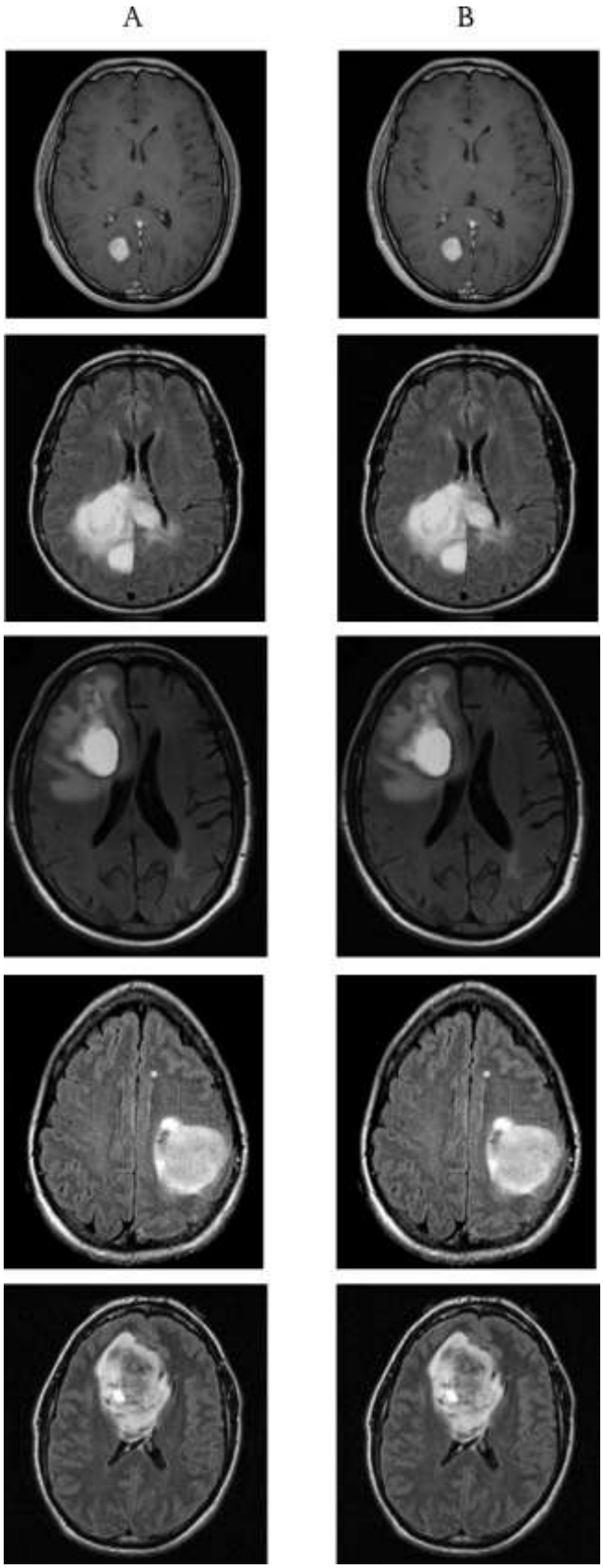


Рисунок 4.18 – Приклад зображень використаних для аналізу (А) – вхідні зображення (Б) – зображення з нанесеним ЦВЗ

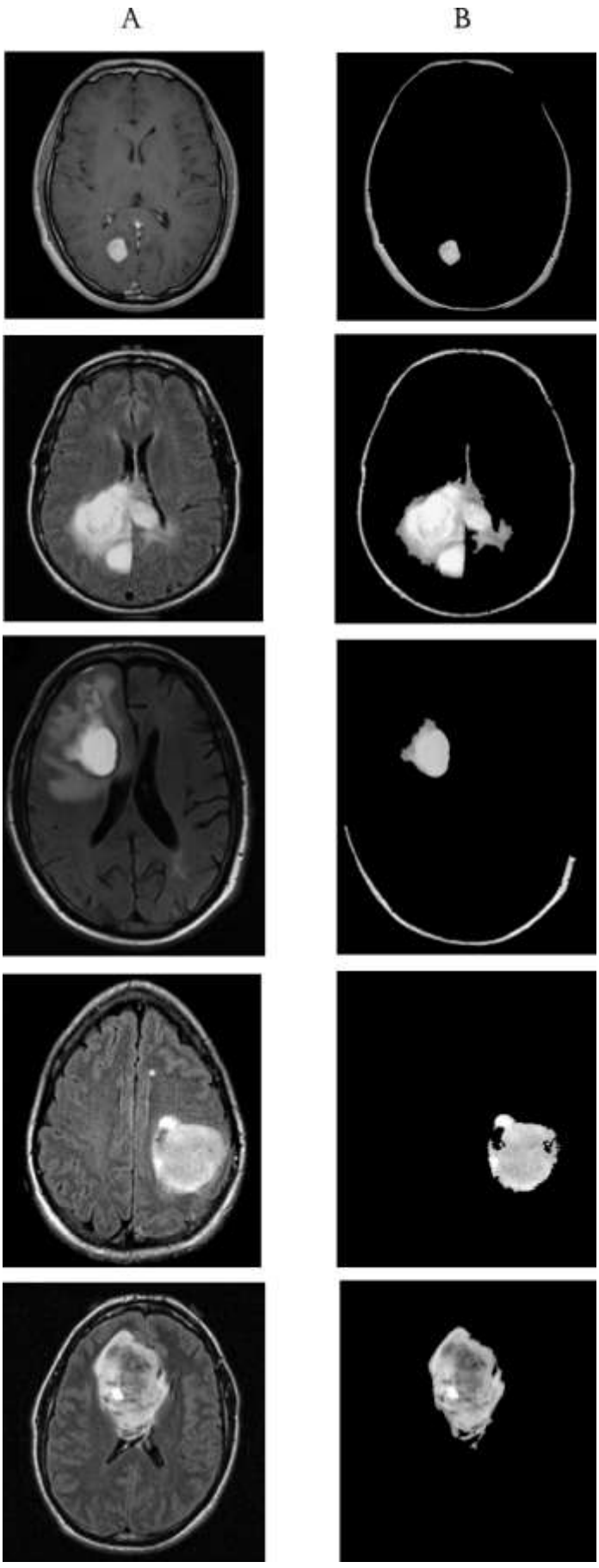


Рисунок 4.19 – Приклад зображень з результатами сегментації

У представленому методі ми приховуємо ROI та EHR всередині вхідного зображення. Після шифрування отримуються ROI та EHR із зображення га якому нанесено водяний знак. Для підвищення ефективності системи ми застосовуємо два типи рівня безпеки. Перший включає процес шифрування, а другий – процес стиснення. Цей гібридний алгоритм має меншу обчислювальну складність. На рисунку 4.20 показано ефективність запропонованого методу сегментації на основі PSNR шляхом порогування (threshold).

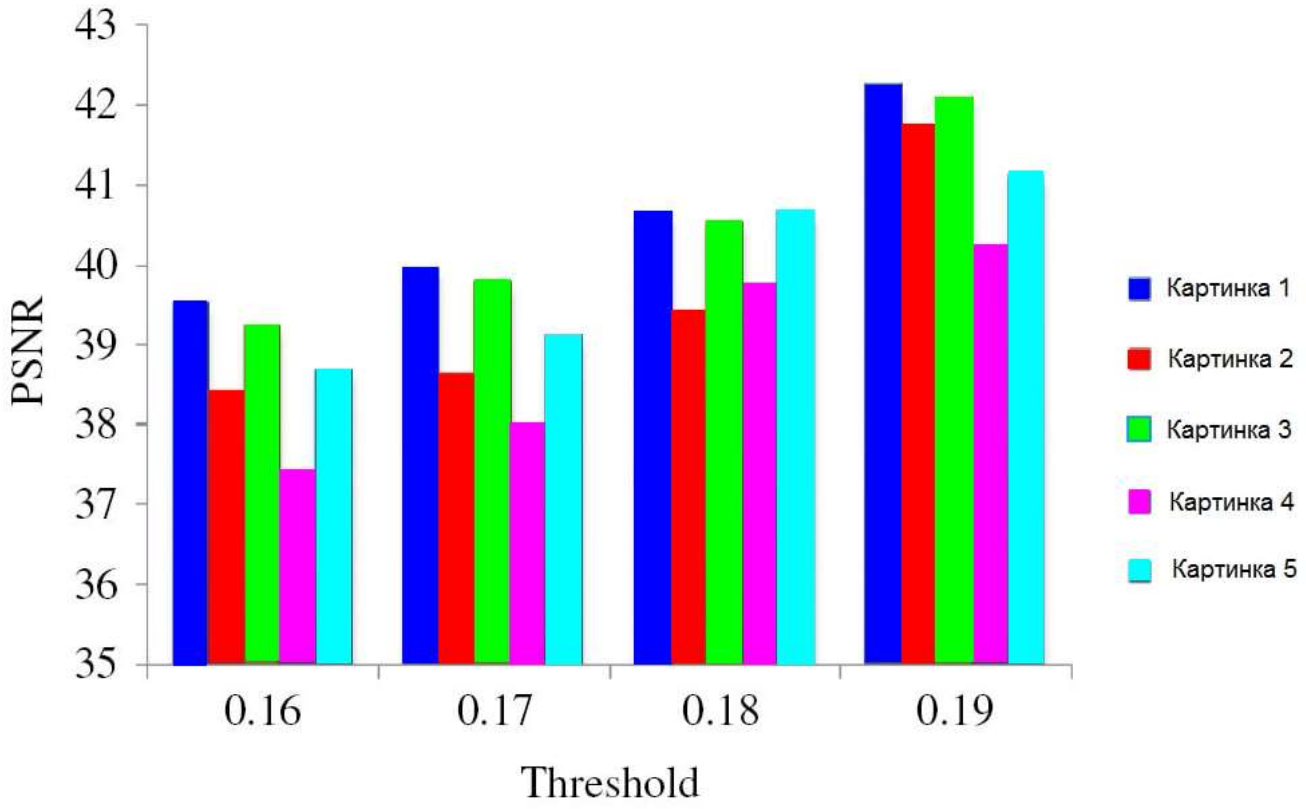


Рисунок 4.20 – Результати тестування ефективності запропонованого методу на основі PSNR шляхом порогування

У запропонованій методології для сегментації використовується алгоритм росту регіонів (RG). В алгоритмі RG продуктивність змінюється на основі порогового значення (threshold). Згідно з аналізом, значення PSNR поступово збільшується, коли порогове значення продовжує зростати. Значення PSNR є низьким, коли порогове значення становить 0,16; аналогічно, значення PSNR є

високим, коли порогове значення становить 0,19. Тут запропонований підхід досягає максимального значення PSNR у 42,2361 db.

На рисунку 4.21 порівнюється сегментоване зображення ROI із отриманим ROI. Отримані наступні результати: максимальну точність 98,5% для зображення 1, 97,07% для зображення 2, 96,4% для зображення 3, 97,78% для зображення 4 та 96,24% для зображення 5

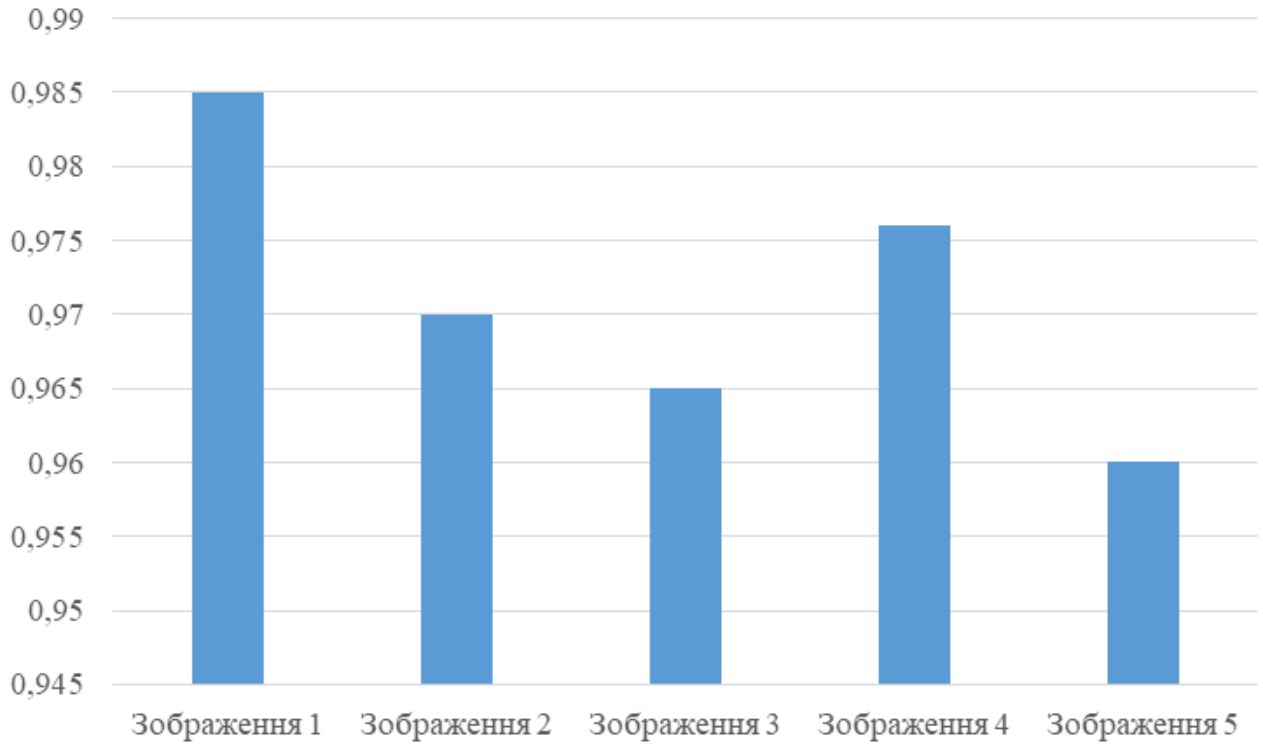


Рисунок 4.21 – Порівняння області інтересів вхідного зображення та маркованого зображення

. На рисунку 4.22 показано результати вимірювань нормалізованої кореляції (NC). Запропонований підхід досягає максимального значення NC.

Також на рисунку 4.23 показано ефективність запропонованої техніки з використанням різних атак. Атаки застосовуються на етапі шифрування.

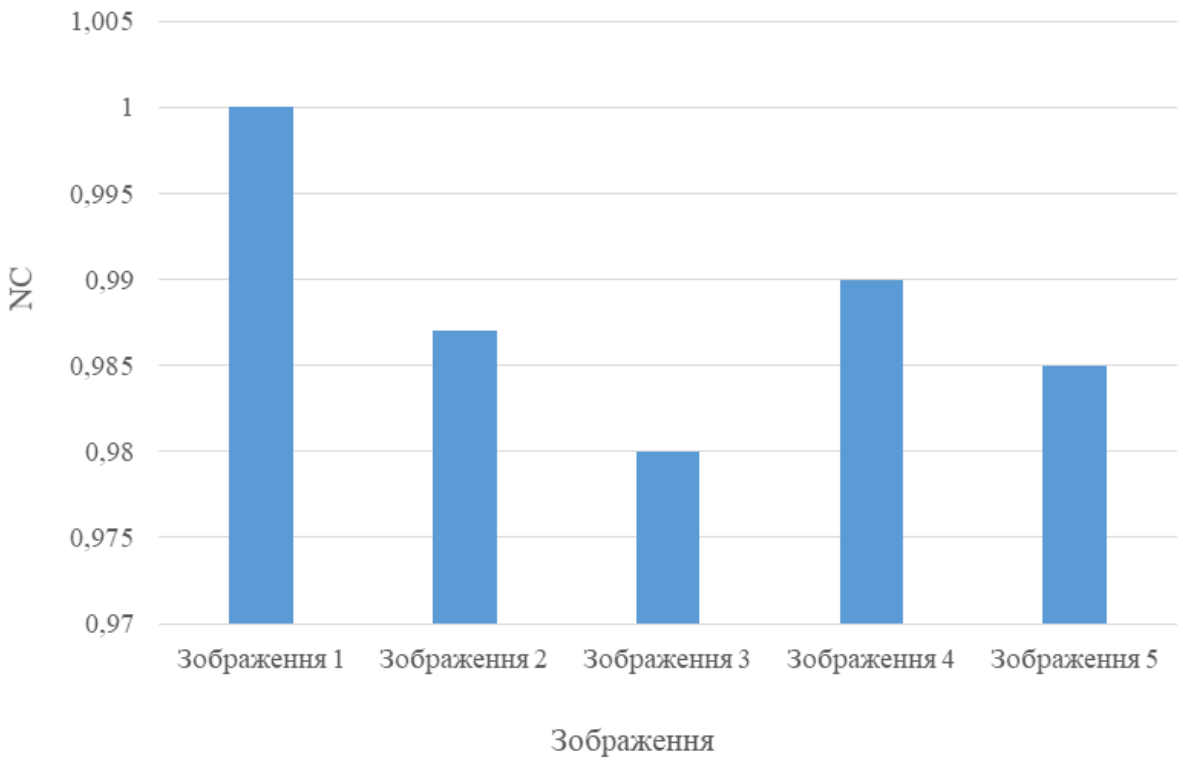


Рисунок 4.22 – Результати вимірювань нормалізованої кореляції (NC)

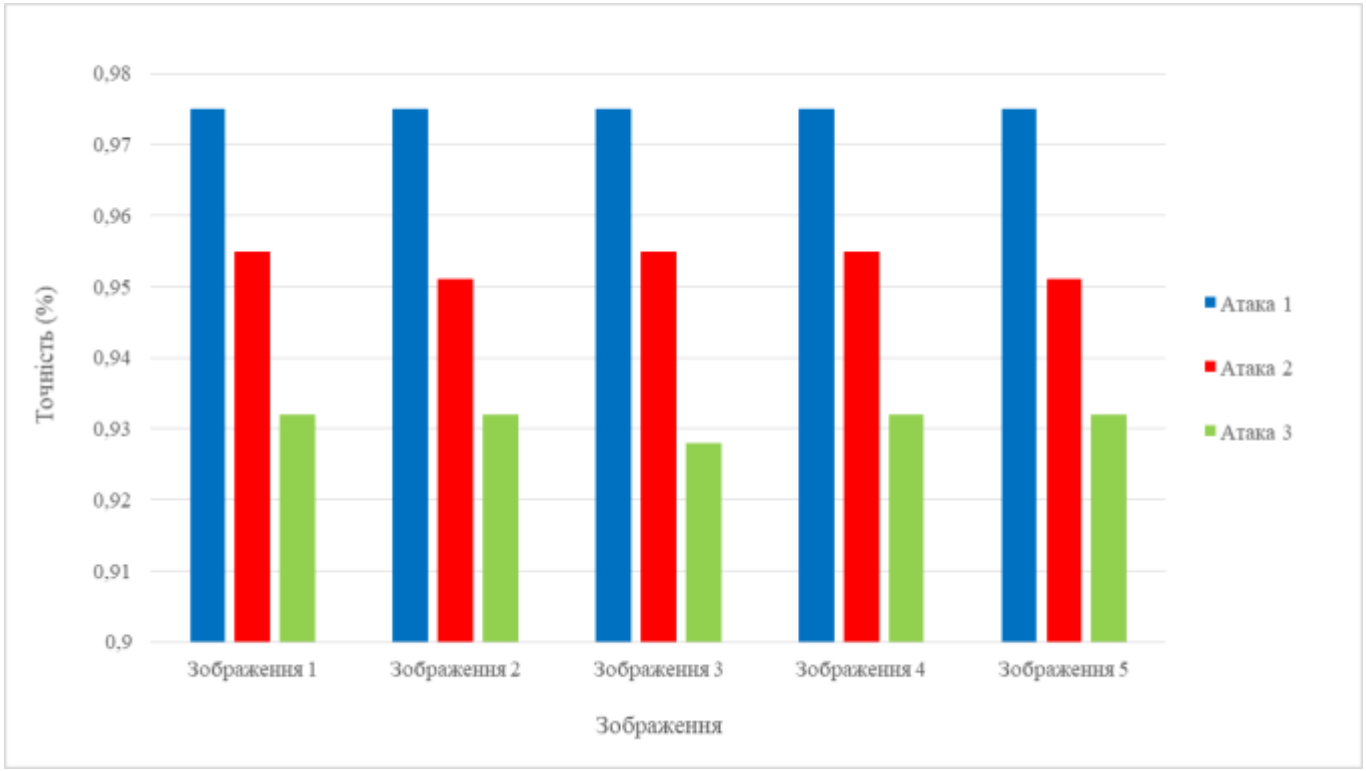


Рисунок 4.23 – Ефективність запропонованої техніки з використанням різних атак

Атака 1 являє собою зміну п’яти пікселів і застосовується алгоритм шифрування, а потім вимірюється результат. Отримані майже однакові значення. Таким чином, ця атака не впливає належним чином на результати. Подібним чином змінюються значення пікселів і тестуються отримані результати. Після застосування атаки метод показує кращі результати. У таблиці 4.24 наведені медичні дані (EHR) після застосування атак.

Результати отриманих медичних даних при проведені атак	
Оригінальні дані	Patient: Петро Петренко, PatientID: 43252124, Doctor: Іван Іванович, DoctorID: 231231,MedicalCenter: Тестова клініка, AdditionalInfo: Тест
Атака: зміна 5 пікселів	Patient: Петро Петренко, PatientID: 43252124, Doctor: Іва.н Іванович, DoctorID: 231231,MedicalCenter: 3Тестова клініка, AdditionalInfo: Тест
Атака: зміна 10 пікселів	Patient: Петро Петренко, PatientID: 43252124, Doctor: Іван Іванов2*ич, DoctorID: 231231,Me#dicalCenter: qТестова клініка, AdditionalInfo: Тест
Атака: зміна 15 пікселів	Patient:1 Петро Петренко, `PatientID: 43252124, D0ctor: 0ван Іванов./ич, DoctorID: 2312.31,MedicalCenter: Тестwова клініка, Ad1itionalInfo: Тест

Таблиця 4.24 – Отримані медичні дані (EHR) після застосування різного типу атак.

На рисунку 4.25 наведено результати атаки водяних знаків для різних зображень.

Attacks	PSNR				
	Зображення 1	Зображення 2	Зображення 3	Зображення 4	Зображення 5
Salt and pepper noise 10%	41.43	40.65	41.54	38.34	40.53
Salt and pepper noise 10%	40.5	39.74	40.36	37.61	38.52
Salt and pepper noise 10%	39.1	38.02	38.29	36.53	37.43
Gaussian noise 20%	40.46	39.99	41.34	37.73	39.45
Gaussian noise 40%	39.51	38.45	40.34	36.83	38.54
Gaussian noise 60%	38.18	37.47	38.45	35.13	37.63
Crop 5%	40.62	39.43	40.61	38.23	38.35
Crop 10%	38.54	38.12	38.56	36.83	37.45
Crop 20%	37.73	37.14	37.83	35.82	36.92

Рисунок 4.25 – Результати атаки водяних знаків для різних зображень.

У цій роботі ми використовували три типи атак: шум солі та перцю (Salt and pepper), гаусів шум (Gaussian noise) та обрізання (crop). З отриманих результатів зрозуміло, що атаки не впливають на зображення з нанесеним ЦВЗ. На рисунку 4.26 наведено ефективність запропонованого підходу з використанням значення PSNR та ємності вбудовування (біти). З результатів можна зробити висновок, що запропонований алгоритм досягає кращих результатів.

Method	Images	File format	PSNR	Embedding capacity (bits)
Proposed method	Image 1	JPEG	42.23	72,384
	Image 2	JPEG	41.71	72,384
	Image 3	JPEG	42.102	72,384
	Image 4	JPEG	40.22	72,384
	Image 5	JPEG	41.105	72,384

Рисунок 4.26 – Результати атак на зображення з ЦВЗ

Ефективність запропонованої ІТ нанесення водяних знаків аналізується за допомогою PSNR та NC. Ефективність запропонованої методики демонструється шляхом порівняння результатів узгодження запропонованої інформаційної технології методу з результатами існуючих методів [35].

На рисунку 4.27 представлено аналіз ефективності на основі точності сегментації. Порівнюється запропонований алгоритм RG з алгоритмом кластеризації k–середніх значень. Це також один з алгоритмів сегментації, що використовується для визначення області інтересів (ROI).

Запропонований підхід (рис. 4.27) досягає максимальної точності 95,3% для зображення 1, 97,1% для зображення 2, 92,54% для зображення 3, 93,62% для зображення 4 і 94,67% для зображення 5.

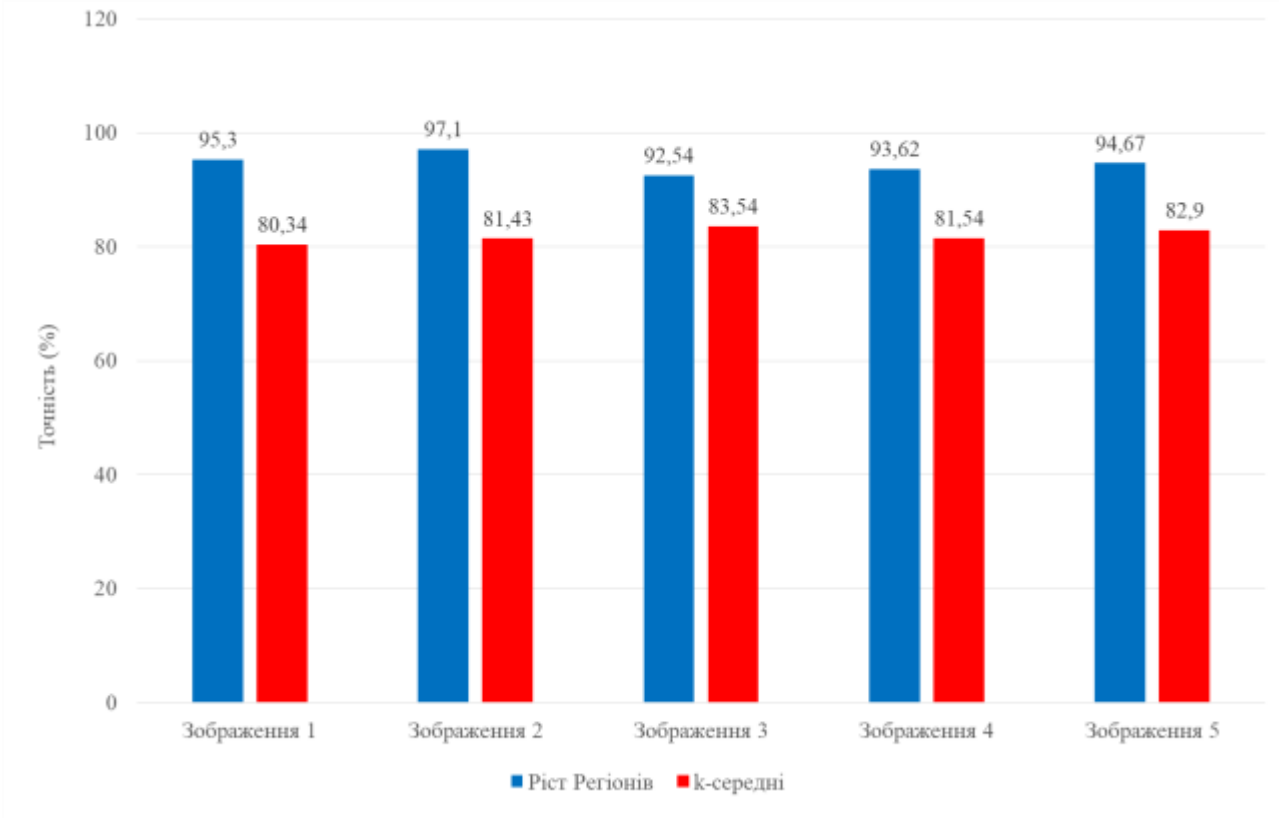


Рисунок 4.27 – Результати аналізу ефективності на основі точності сегментації

Аналогічно, використовуючи k–середній кластеризуючий алгоритм, ми отримуємо максимальні показники PSNR 80,34%, 81,43%, 83,54%, 81,54% та 82,9%, відповідно.

З результату ми робимо висновок, що запропонований алгоритм сегментації (росту регіонів) кращий, ніж алгоритм k–середніх. Більше того, на рисунку 4.28 наведено порівняльний аналіз запропонованого підходу щодо існуючого показника PSNR. Отже запропонований підхід досягає максимального значення PSNR 42,23db, що становить 38,53db для існуючого алгоритму. З результатів можна зробити висновок, що запропонований підхід кращий за існуючі.

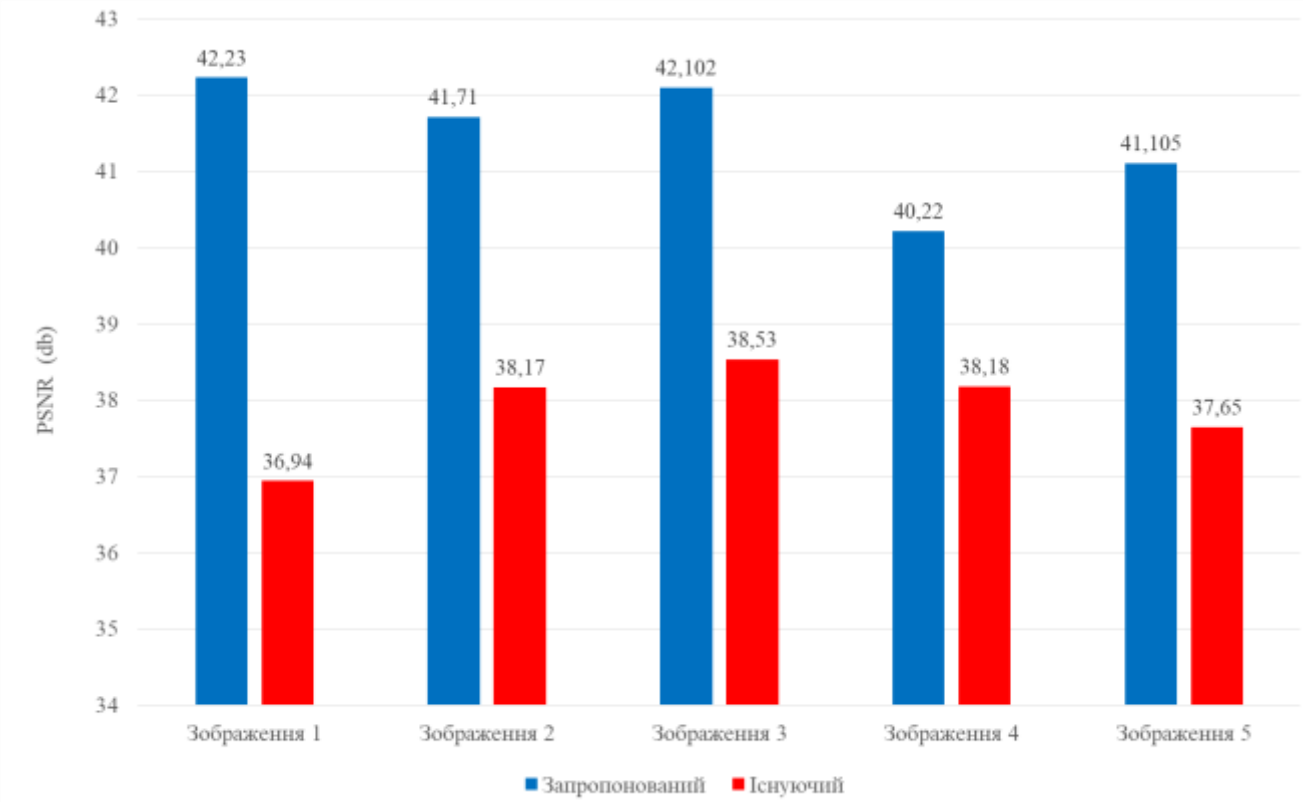


Рисунок 4.28 – Порівняння показника пікового співвідношення сигналу до шуму (PSNR) для запропонованого алгоритму та для існуючих

Також наведено порівняльний аналіз (рис. 4.29) між запропонованим та існуючим показником NC. Запропонований підхід забезпечує максимальне значення NC порівняно з іншими методами.

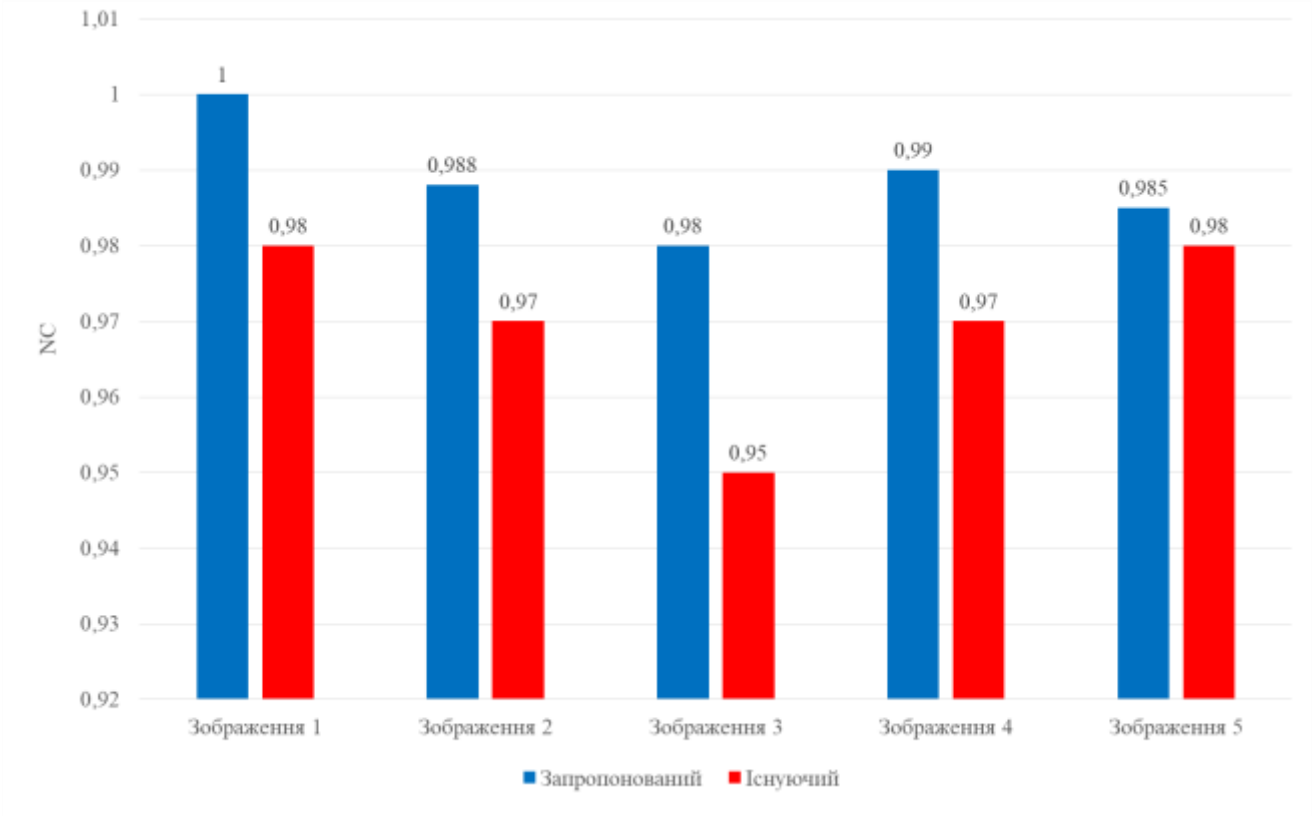


Рисунок 4.29 – Порівняння показника нормалізованої кореляції (NC) для запропонованого алгоритму та для існуючих

Висновки до розділу 4

Сформулювало інноваційний підхід до водяних знаків у цифровій обробці зображень. Проаналізовано алгоритми RG, SHA–256, ECC та AC. Складність обчислень запропонованого методу менша, оскільки він використовує прості математичні розрахунки для формування даних автентифікації та відновлення та для відновлення зашифрованих даних.

Запропонована схема підтримує якість зображення з водяним знаком із середнім значенням PSNR 42,23 db, потужністю вбудовування 72 384 біт, вилученою точністю 98% і NC.

Експериментальні результати також свідчать про те, що запропонована IT забезпечує кращу якість зображення із водяними знаками та збільшує продуктивність вбудовування. Запропонована методика може бути практично

включена до медичних інформаційних систем для забезпечення цілісності медичного зображення, автентифікації системи та конфіденційності.

Загальні висновки

Забезпечення автентичності, конфіденційності та безпеки є важливим завдання для медичних інформаційних систем, саме ці фактори не тільки впливають на довіру та безпеку користувачів, але й забезпечуються подальше правильне лікування та діагностування хвороби, адже автентичні зображення несуть оригінальну візуальну інформацію від лікаря до пацієнта і навпаки. Таким чином користувач системи може бути впевнений, що зображення які проходять через систему та є автентичними не піддались змінам з боку третьої сторони або якихось технічних помилок при передачі даних.

Згідно до завдань дипломної роботи розглянуто основні аспекти та процеси маркування та перевірки автентичності даних.

Проведено аналіз наукових робіт та публікацій на основі яких було визначено основні вимоги та проблематику маркування та перевірки автентичності медичних даних, нанесення ЦВЗ, а саме: технологія маркування не повинна змінювати візуальну інформацію вхідного зображення та нанесений ЦВЗ має бути невидимим для людського ока та максимально співпадати з вхідним зображенням, також задача перевірки автентичності, тобто передача та зчитування ЦВЗ як носія деякої інформації та не була детально розглянута та вивчена, що вказує на потребу побудови відповідних методів та ІТ.

На основі поставлених завдань та проведеного аналізу наукових робіт представлено інформаційну технологію маркування та перевірки автентичності медичних зображень та основну структуру інформаційної системи, основні алгоритми та методи нанесення цифрового водяного знаку. Також за допомогою таких методів як: метод росту регіонів (ROI), еліптичної криптографії та шифрування SHA–256, було запроваджено багатошаровий рівень захисту на кожному з етапів роботи з медичним зображенням та відповідними даними.

Також представлено основні алгоритми та інтерфейси для маркування та перевірки автентичності медичних даних. Описані відповідні поля інтерфейсів та

відповідні методи. Побудовано схеми оптимального використання та об’єднання відповідних алгоритмів та найкращу їх послідовність.

Представлено інноваційний підхід нанесения водяних знаків на медичні зображення. Проаналізовано та перевірено різні алгоритми які запроваджують додаткові рівні безпеки або є допоміжними для інших методів, а саме: RG, SHA–256, ECC та AC. Встановлено, що складність обчислень запропонованої інформаційної технології менша, оскільки вона використовує прості математичні розрахунки для формування даних автентифікації та відновлення та для відновлення зашифрованих даних.

Запропонована методологія підтримує якість зображення з цифровим водяним знаком із середнім значенням PSNR 42,23 db, потужністю вбудовування 72 384 біт, вилученою точністю 98% і NC.

Отримані експериментальні результати апробації вказують на те, що запропонована інформаційна технологія забезпечує кращу якість нанесення ЦВЗ на зображення порівнюючи з іншим та краще показує себе у кращій продуктивності вбудовування. Представлена інформаційна технологія може бути практично включена до медичних інформаційних систем для забезпечення цілісності медичного зображення, автентифікації системи та конфіденційності.

Отже, в результаті виконання поставлених задач та реалізації відповідної інформаційної технології було досягнуто поставленої мети дипломної роботи магістра

Перелік посилань

1. Медична_візуалізація [Електронний ресурс]. – Режим доступу: https://uk.wikipedia.org/wiki/Медична_візуалізація

2. Реберный хондрит (синдром Титце) [Електронний ресурс] Режим доступу: https://www.dikul.net/files/images/otdeleniya/rg_kisty_ruki_1.jpg

3. Ультразвуковое исследование брюшной аорты [Електронний ресурс] Режим доступу: https://mdcexpert.com/articles/us–abdominal–aorta/

4. Healthcare CRM [Електронний ресурс] Режим доступу: https://www.softwareadvice.com/crm/healthcare–crm–comparison/

5. DICOM [Електронний ресурс]. – Режим доступу: https://www.dicomstandard.org/

6. Significance of digital imaging and communication in medicine in digital imaging [Електронний ресурс]. – Режим доступу: https://www.digitmedicine.com/article.asp?issn=2226–8561;year=2015;volume=1;issue=2;spage=63;epage=66;aulast=Gupta#:~:text=A%20major%20disadvantage%20of%20the,are%20filled%20with%20incorrect%20data.

7. Strategic document [Електронний ресурс] Режим доступу: http://dicom.nema.org/dicom/geninfo/Strategy.pdf

8. Mustra, Mario; Delac, Kresimir; Grgic, Mislav. Overview of the DICOM Standard (PDF) / Mustra, Mario; Delac, Kresimir; Grgic, Mislav // International Symposium. Zadar, Croatia 2008 С.39–44.

9. DICOM Library users worldwide [Електронний ресурс] Режим доступу: https://www.dicomlibrary.com/

10. How to Insert a Watermark in Excel: Step by Step Tutorial [Електронний ресурс] Режим доступу: https://www.excelmasterconsultant.com/single–post/2018/08/30/How–to–Insert–a–Watermark–in–Excel–Step–by–Step–Tutorial

11. Invisible Watermarking Based on Creation and Robust Insertion–Extraction of Image Adaptive Watermarks [Електронний ресурс]. – Режим доступу:

https://www.researchgate.net/publication/220214375_Invisible_Watermarking_Based_on_Creation_and_Robust_Insertion–Extraction_of_Image_Adaptive_Watermarks

12. Криптографія [Електронний ресурс] Режим доступу: https://uk.wikipedia.org/wiki/ Криптографія

13. Steganography, Cryptography and Watermarking: A Review [Електронний ресурс]. – Режим доступу: http://www.ijirset.com/upload/2017/february/76_21_Steganography.pdf

14. Стеганографія [Електронний ресурс] Режим доступу: https://uk.wikipedia.org/wiki/Стеганографія

15. Анализ проблематики использования изображений в цифровом формате в медицинской практике [Електронний ресурс] Режим доступу: http://www.irbis–nbuv.gov.ua/cgi–bin/irbis_nbuv/cgiirbis_64.exe?C21COM=2&I21DBN=UJRN&P21DBN=UJRN&IMAGE_FILE_DOWNLOAD=1&Image_file_name=PDF/soi_2016_4_32.pdf

16. Сегментація зображення [Електронний ресурс] Режим доступу: https://uk.wikipedia.org/wiki/ Сегментація_зображенняя

17. Image segmentation Techniques and its application [Електронний ресурс] Режим доступу: https://www.researchgate.net/publication/340087951_Image_segmentation_Techniques_and_its_application

18. Segmentation of anatomical organs in medical data [Електронний ресурс] Режим доступу: https://vgg.fiit.stuba.sk/category/research–areas/medical_imaging/

19. Ідентифікація і захист мультимедійних даних [Електронний ресурс] Режим доступу:

http://www.hups.mil.gov.ua/periodic–app/article/10081/soi_2012_8_23.pdf

20. О методе цифровых водяных знаков на основе особенностей изображения моментов Цернике [Електронний ресурс] Режим доступу: http://dspace.nbuv.gov.ua/bitstream/handle/123456789/6977/13–Nikitina.pdf?sequence=1

21. Використання цифрових водяних знаків для захисту авторських прав [Електронний ресурс] Режим доступу:

http://conf.inf.od.ua/doklady–konferentsii/spisok–materialov–konferentsii/64–kravchuk–r–yu–studentki–2–go–kursu–institutu–prokuraturi–ta–slidstva–nu–oyua–naukovij–kerivnik–k–f–m–n–dotsent–kozin–o–b–vikoristannya–tsifrovikh–vodyanikh–znakiv–dlya–zakhistu–avtorskikh–prav

22. Дослідження програмних продуктів для накладення цифрового водяного знаку та інших методів захисту мультимедійних даних та об'єктів інтелектуальної власності http://www.irbis–nbuv.gov.ua/cgi–bin/irbis_nbuv/cgiirbis_64.exe?C21COM=2&I21DBN=UJRN&P21DBN=UJRN&IMAGE_FILE_DOWNLOAD=1&Image_file_name=PDF/Mtit_2017_80_28.pdf

23. Шляхи підвищення захисту авторського права за допомогою використання цифрових водяних знаків https://miljournals.knu.ua/index.php/zbirnuk/article/download/219/207/

24. Watermarking techniques used in medical images [Електронний ресурс] Режим доступу: https://europepmc.org/article/med/24871349

25. A Survey on Medical Image Watermarking Techniques [Електронний ресурс] Режим доступу: http://ijcsn.org/IJCSN–2014/3–5/A–Survey–on–Medical–Image–Watermarking–Techniques.pdf

26. Watermarking Techniques used in Medical Images: a Survey [Електронний ресурс] Режим доступу:

https://www.researchgate.net/publication/262695613_Watermarking_Techniques_used_in_Medical_Images_a_Survey

27. Medical images verification [Електронний ресурс] Режим доступу: http://www.hups.mil.gov.ua/periodic–app/article/6742/soi_2009_7_5.pdf

28. Исследование методов реверсивных цифровых знаков (ЦВЗ) для верификации медицинских изображений [Електронний ресурс] Режим доступу: http://www.hups.mil.gov.ua/periodic–app/article/17404/soi_2017_2_23.pdf

29. Mohanty S. P., Ramakrishnan K. R. A dual watermarking technique for images / Mohanty SP, Ramakrishnan KR // ACM Press, 1999 С.49–51.

30. Image Watermarking [Електронний ресурс] Режим доступу: https://www.sciencedirect.com/topics/engineering/image–watermarking

31. Digital Image Watermarking: An Overview [Електронний ресурс] Режим доступу: https://www.researchgate.net/publication/272747716_Digital_Image_Watermarking_An_Overview

32. An efficient medical image watermarking scheme based on FDCuT–DCT [Електронний ресурс] Режим доступу: https://www.sciencedirect.com/science/article/pii/S2215098617304287

33. Medical Image Watermarking Technique in the Application of E–diagnosis Using M–Ary Modulation [Електронний ресурс] Режим доступу: https://www.sciencedirect.com/science/article/pii/S187705091630597X

34. Example of Medical CMS Design [Електронний ресурс] Режим доступу: https://dz7xwpjzpkel8.cloudfront.net/production/6454–0–original.jpg

35. S. A. Parah, J. A. Sheikh, F. Ahad, N. A. Loan and G. M. Bhat, Information hiding in medical images: a robust medical image watermarking system for E–healthcare / S. A. Parah, J. A. Sheikh, F. Ahad, N. A. Loan and G. M // *J. Multimed. Tools* – 2017 №76.

36. Мостовий В.В., Горященко С. Л. Сегментація медичних зображень / Мостовий В.В, // «Вісник Хмельницького Національно Університету» – Хмельницький: ХНУ, 2020.

ДОДАТКИ

Додаток А

Програмні коди

```
// SHA-256
var Crypto = Crypto || function(a, m) {
    var r = {},
        f = r.lib = {},
        g = function() {},
        l = f.Base = {
            extend: function(a) {
                g.prototype = this;
                var b = new g;
                a && b.mixIn(a);
                b.hasOwnProperty("init") || (b.init = function() {
                    b.$super.init.apply(this, arguments)
                });
                b.init.prototype = b;
                b.$super = this;
                return b
            },
            create: function() {
                var a = this.extend();
                a.init.apply(a, arguments);
                return a
            },
            init: function() {},
            mixIn: function(a) {
                for (var b in a) a.hasOwnProperty(b) && (this[b] = a[b]);
                a.hasOwnProperty("toString") && (this.toString =
a.toString)
            },
            clone: function() {
                return this.init.prototype.extend(this)
            }
        },
        p = f.WordArray = l.extend({
            init: function(a, b) {
                a = this.words = a || [];
                this.sigBytes = b != m ? b : 4 * a.length
            },
            toString: function(a) {
                return (a || q).stringify(this)
            },
            concat: function(a) {
                var b = this.words,
                    d = a.words,
                    c = this.sigBytes;
                a = a.sigBytes;
```

```
                this.clamp();
                if (c % 4)
                    for (var j = 0; j < a; j++) b[c + j >>> 2] |= (d[j >>>
2] >>> 24 - 8 * (j % 4) & 255) << 24 - 8 * ((c + j) % 4);
                else if (65535 < d.length)
                    for (j = 0; j < a; j += 4) b[c + j >>> 2] = d[j >>>
2];
                else b.push.apply(b, d);
                this.sigBytes += a;
                return this
            },
            clamp: function() {
                var n = this.words,
                    b = this.sigBytes;
                n[b >>> 2] &= 4294967295 <<
                    32 - 8 * (b % 4);
                n.length = a.ceil(b / 4)
            },
            clone: function() {
                var a = l.clone.call(this);
                a.words = this.words.slice(0);
                return a
            },
            random: function(n) {
                for (var b = [], d = 0; d < n; d += 4) b.push(4294967296 *
a.random() | 0);
                return new p.init(b, n)
            }
        }),
        y = r.enc = {},
        q = y.Hex = {
            stringify: function(a) {
                var b = a.words;
                a = a.sigBytes;
                for (var d = [], c = 0; c < a; c++) {
                    var j = b[c >>> 2] >>> 24 - 8 * (c % 4) & 255;
                    d.push((j >>> 4).toString(16));
                    d.push((j & 15).toString(16))
                }
                return d.join("")
            },
            parse: function(a) {
                for (var b = a.length, d = [], c = 0; c < b; c += 2) d[c
>>> 3] |= parseInt(a.substr(c,
                    2), 16) << 24 - 4 * (c % 8);
                return new p.init(d, b / 2)
            }
        },
        G = y.Latin1 = {
            stringify: function(a) {
```

```
                var b = a.words;
                a = a.sigBytes;
                for (var d = [], c = 0; c < a; c++)
d.push(String.fromCharCode(b[c >>> 2] >>> 24 - 8 * (c % 4) & 255));
                return d.join("")
            },
            parse: function(a) {
                for (var b = a.length, d = [], c = 0; c < b; c++) d[c >>>
2] |= (a.charCodeAt(c) & 255) << 24 - 8 * (c % 4);
                return new p.init(d, b)
            }
        },
        fa = y.Utf8 = {
            stringify: function(a) {
                try {
                    return decodeURIComponent(escape(G.stringify(a)))
                } catch (b) {
                    throw Error("Malformed UTF-8 data");
                }
            },
            parse: function(a) {
                return G.parse(unescape(encodeURIComponent(a)))
            }
        },
        h = f.BufferedBlockAlgorithm = l.extend({
            reset: function() {
                this._data = new p.init;
                this._nDataBytes = 0
            },
            _append: function(a) {
                "string" == typeof a && (a = fa.parse(a));
                this._data.concat(a);
                this._nDataBytes += a.sigBytes
            },
            _process: function(n) {
                var b = this._data,
                    d = b.words,
                    c = b.sigBytes,
                    j = this.blockSize,
                    l = c / (4 * j),
                    l = n ? a.ceil(l) : a.max((l | 0) -
this._minBufferSize, 0);
                n = l * j;
                c = a.min(4 * n, c);
                if (n) {
                    for (var h = 0; h < n; h += j) this._doProcessBlock(d,
h);
                    h = d.splice(0, n);
                    b.sigBytes -= c
                }
```

```
                return new p.init(h, c)
            },
            clone: function() {
                var a = l.clone.call(this);
                a._data = this._data.clone();
                return a
            },
            _minBufferSize: 0
        });
    f.Hasher = h.extend({
        cfg: l.extend(),
        init: function(a) {
            this.cfg = this.cfg.extend(a);
            this.reset()
        },
        reset: function() {
            h.reset.call(this);
            this._doReset()
        },
        update: function(a) {
            this._append(a);
            this._process();
            return this
        },
        finalize: function(a) {
            a && this._append(a);
            return this._doFinalize()
        },
        blockSize: 16,
        _createHelper: function(a) {
            return function(b, d) {
                return (new a.init(d)).finalize(b)
            }
        },
        _createHmacHelper: function(a) {
            return function(b, d) {
                return (new ga.HMAC.init(a,
                    d)).finalize(b)
            }
        }
    });
    var ga = r.algo = {};
    return r
}(Math);
(function(a) {
    var m = CryptoJS,
        r = m.lib,
        f = r.Base,
        g = r.WordArray,
        m = m.x64 = {};
```

```
    m.Word = f.extend({
        init: function(a, p) {
            this.high = a;
            this.low = p
        }
    });
    m.WordArray = f.extend({
        init: function(l, p) {
            l = this.words = l || [];
            this.sigBytes = p != a ? p : 8 * l.length
        },
        toX32: function() {
            for (var a = this.words, p = a.length, f = [], q = 0; q < p;
q++) {
                var G = a[q];
                f.push(G.high);
                f.push(G.low)
            }
            return g.create(f, this.sigBytes)
        },
        clone: function() {
            for (var a = f.clone.call(this), p = a.words =
this.words.slice(0), g = p.length, q = 0; q < g; q++) p[q] = p[q].clone();
            return a
        }
    })
})();
(function() {
    function a() {
        return g.create.apply(g, arguments)
    }
    for (var m = CryptoJS, r = m.lib.Hasher, f = m.x64, g = f.Word, l =
f.WordArray, f = m.algo, p = [a(1116352408, 3609767458), a(1899447441,
602891725), a(3049323471, 3964484399), a(3921009573, 2173295548),
a(961987163, 4081628472), a(1508970993, 3053834265), a(2453635748,
2937671579), a(2870763221, 3664609560), a(3624381080, 2734883394),
a(310598401, 1164996542), a(607225278, 1323610764), a(1426881987,
3590304994), a(1925078388, 4068182383), a(2162078206, 991336113),
a(2614888103, 633803317),
            a(3248222580, 3479774868), a(3835390401, 2666613458),
a(4022224774, 944711139), a(264347078, 2341262773), a(604807628,
2007800933), a(770255983, 1495990901), a(1249150122, 1856431235),
a(1555081692, 3175218132), a(1996064986, 2198950837), a(2554220882,
3999719339), a(2821834349, 766784016), a(2952996808, 2566594879),
a(3210313671, 3203337956), a(3336571891, 1034457026), a(3584528711,
2466948901), a(113926993, 3758326383), a(338241895, 168717936),
a(666307205, 1188179964), a(773529912, 1546045734), a(1294757372,
1522805485), a(1396182291,
                2643833823), a(1695183700, 2343527390), a(1986661051,
1014477480), a(2177026350, 1206759142), a(2456956037, 344077627),
```

```
a(2730485921, 1290863460), a(2820302411, 3158454273), a(3259730800,
3505952657), a(3345764771, 106217008), a(3516065817, 3606008344),
a(3600352804, 1432725776), a(4094571909, 1467031594), a(275423344,
851169720), a(430227734, 3100823752), a(506948616, 1363258195),
a(659060556, 3750685593), a(883997877, 3785050280), a(958139571,
3318307427), a(1322822218, 3812723403), a(1537002063, 2003034995),
a(1747873779, 3602036899),
            a(1955562222, 1575990012), a(2024104815, 1125592928),
a(2227730452, 2716904306), a(2361852424, 442776044), a(2428436474,
593698344), a(2756734187, 3733110249), a(3204031479, 2999351573),
a(3329325298, 3815920427), a(3391569614, 3928383900), a(3515267271,
566280711), a(3940187606, 3454069534), a(4118630271, 4000239992),
a(116418474, 1914138554), a(174292421, 2731055270), a(289380356,
3203993006), a(460393269, 320620315), a(685471733, 587496836),
a(852142971, 1086792851), a(1017036298, 365543100), a(1126000580,
2618297676), a(1288033470,
                3409855158), a(1501505948, 4234509866), a(1607167915,
987167468), a(1816402316, 1246189591)
        ], y = [], q = 0; 80 > q; q++) y[q] = a();
    f = f.SHA512 = r.extend({
        _doReset: function() {
            this._hash = new l.init([new g.init(1779033703, 4089235720),
new g.init(3144134277, 2227873595), new g.init(1013904242, 4271175723),
new g.init(2773480762, 1595750129), new g.init(1359893119, 2917565137),
new g.init(2600822924, 725511199), new g.init(528734635, 4215389547), new
g.init(1541459225, 327033209)])
        },
        _doProcessBlock: function(a, f) {
            for (var h = this._hash.words,
                    g = h[0], n = h[1], b = h[2], d = h[3], c = h[4], j =
h[5], l = h[6], h = h[7], q = g.high, m = g.low, r = n.high, N = n.low, Z
= b.high, O = b.low, $ = d.high, P = d.low, aa = c.high, Q = c.low, ba =
j.high, R = j.low, ca = l.high, S = l.low, da = h.high, T = h.low, v = q,
s = m, H = r, E = N, I = Z, F = O, W = $, J = P, w = aa, t = Q, U = ba, K
= R, V = ca, L = S, X = da, M = T, x = 0; 80 > x; x++) {
                var B = y[x];
                if (16 > x) var u = B.high = a[f + 2 * x] | 0,
                    e = B.low = a[f + 2 * x + 1] | 0;
                else {
                    var u = y[x - 15],
                        e = u.high,
                        z = u.low,
                        u = (e >>> 1 | z << 31) ^ (e >>> 8 | z << 24) ^ e
>>> 7,
                        z = (z >>> 1 | e << 31) ^ (z >>> 8 | e << 24) ^ (z
>>> 7 | e << 25),
                        D = y[x - 2],
                        e = D.high,
                        k = D.low,
                        D = (e >>> 19 | k << 13) ^
```

```
                    (e << 3 | k >>> 29) ^ e >>> 6,
                    k = (k >>> 19 | e << 13) ^ (k << 3 | e >>> 29) ^
(k >>> 6 | e << 26),
                    e = y[x - 7],
                    Y = e.high,
                    C = y[x - 16],
                    A = C.high,
                    C = C.low,
                    e = z + e.low,
                    u = u + Y + (e >>> 0 < z >>> 0 ? 1 : 0),
                    e = e + k,
                    u = u + D + (e >>> 0 < k >>> 0 ? 1 : 0),
                    e = e + C,
                    u = u + A + (e >>> 0 < C >>> 0 ? 1 : 0);
                B.high = u;
                B.low = e
            }
            var Y = w & U ^ ~w & V,
                C = t & K ^ ~t & L,
                B = v & H ^ v & I ^ H & I,
                ha = s & E ^ s & F ^ E & F,
                z = (v >>> 28 | s << 4) ^ (v << 30 | s >>> 2) ^ (v <<
25 | s >>> 7),
                D = (s >>> 28 | v << 4) ^ (s << 30 | v >>> 2) ^ (s <<
25 | v >>> 7),
                k = p[x],
                ia = k.high,
                ea = k.low,
                k = M + ((t >>> 14 | w << 18) ^ (t >>> 18 | w << 14) ^
(t << 23 | w >>> 9)),
                A = X + ((w >>> 14 | t << 18) ^ (w >>> 18 | t << 14) ^
(w << 23 | t >>> 9)) + (k >>> 0 < M >>>
                    0 ? 1 : 0),
                k = k + C,
                A = A + Y + (k >>> 0 < C >>> 0 ? 1 : 0),
                k = k + ea,
                A = A + ia + (k >>> 0 < ea >>> 0 ? 1 : 0),
                k = k + e,
                A = A + u + (k >>> 0 < e >>> 0 ? 1 : 0),
                e = D + ha,
                B = z + B + (e >>> 0 < D >>> 0 ? 1 : 0),
                X = V,
                M = L,
                V = U,
                L = K,
                U = w,
                K = t,
                t = J + k | 0,
                w = W + A + (t >>> 0 < J >>> 0 ? 1 : 0) | 0,
                W = I,
```

```
                J = F,
                I = H,
                F = E,
                H = v,
                E = s,
                s = k + e | 0,
                v = A + B + (s >>> 0 < k >>> 0 ? 1 : 0) | 0
            }
            m = g.low = m + s;
            g.high = q + v + (m >>> 0 < s >>> 0 ? 1 : 0);
            N = n.low = N + E;
            n.high = r + H + (N >>> 0 < E >>> 0 ? 1 : 0);
            O = b.low = O + F;
            b.high = Z + I + (O >>> 0 < F >>> 0 ? 1 : 0);
            P = d.low = P + J;
            d.high = $ + W + (P >>> 0 < J >>> 0 ? 1 : 0);
            Q = c.low = Q + t;
            c.high = aa + w + (Q >>> 0 < t >>> 0 ? 1 : 0);
            R = j.low = R + K;
            j.high = ba + U + (R >>> 0 < K >>> 0 ? 1 : 0);
            S = l.low =
                S + L;
            l.high = ca + V + (S >>> 0 < L >>> 0 ? 1 : 0);
            T = h.low = T + M;
            h.high = da + X + (T >>> 0 < M >>> 0 ? 1 : 0)
        },
        _doFinalize: function() {
            var a = this._data,
                f = a.words,
                h = 8 * this._nDataBytes,
                g = 8 * a.sigBytes;
            f[g >>> 5] |= 128 << 24 - g % 32;
            f[(g + 128 >>> 10 << 5) + 30] = Math.floor(h / 4294967296);
            f[(g + 128 >>> 10 << 5) + 31] = h;
            a.sigBytes = 4 * f.length;
            this._process();
            return this._hash.toX32()
        },
        clone: function() {
            var a = r.clone.call(this);
            a._hash = this._hash.clone();
            return a
        },
        blockSize: 32
    });
    m.SHA256 = r._createHelper(f);
    m.HmacSHA256 = r._createHmacHelper(f)
})();
```

```
//helpers function
function rgb2ycbcr(r,g,b){
    /* RGB to Y Cb Cr space */
     return   [0.299*r+0.587*g+0.114*b,   128-0.168736*r-0.331264*g+0.5*b,
128+0.5*r-0.418688*g-0.081312*b];
}

function ycbcr2rgb(y,cb,cr){
    /* Y Cb Cr to RGB space */
     return   [y+1.402*(cr-128),   y-0.344136*(cb-128)-0.714136*(cr-128),
y+1.772*(cb-128)];
}

function get_hashed_order(password, arr_len){
    // O(arr_len) algorithm
    var orders = Array.from(Array(arr_len).keys());
    var result = [];
    var loc;
    var seed = CryptoJS.SHA512(password).words.reduce(function (total, num)
{return total + Math.abs(num);}, 0);
    var rnd = new MersenneTwister(seed);
    for(var i=arr_len; i>0; i--){
        loc = rnd.genrand_int32() % i;
        result.push(orders[loc]);
        orders[loc] = orders[i-1];
    }
    return result;
}


function dct(dataArray) {
    // Apply DCT to a 8*8 data array (64). Expected input is [8*8]
    // input 8*8 | x,y loc x*8+y
```

```
    // output 8*8| u,v loc u*8+v
    var result = Array(64).fill(0);
    var cu, cv, sum;
    for(var u=0; u<8;u++) for(var v=0; v<8; v++){
        cu = (u==0)?1/Math.sqrt(2):1;
        cv = (v==0)?1/Math.sqrt(2):1;
        sum = 0;
        for(var x=0;x<8;x++) for(var y=0;y<8;y++){
            sum                                                    +=
dataArray[x*8+y]*Math.cos((2*x+1)*u*Math.PI/16)*Math.cos((2*y+1)*v*Math.PI
/16);
        }
        result[u*8+v]=(1/4)*cu*cv*sum;
    }

    return result;
}

function idct(dataArray) {
    // Apply inverse DCT to a 8*8 data array (64). Expected output is [8*8]
-> Y Cb Cr
    //input 8*8*3 | u,v loc u*8+v
    //output 8*8*3| x,y loc x*8+y
    result = Array(64).fill(0);
    var cu, cv, sum;
    for(var x=0; x<8;x++) for(var y=0; y<8; y++){
        sum = 0;
        for(var u=0;u<8;u++) for(var v=0;v<8;v++){
            cu = (u==0)?1/Math.sqrt(2):1;
            cv = (v==0)?1/Math.sqrt(2):1;
            sum                                                    +=
cu*cv*dataArray[u*8+v]*Math.cos((2*x+1)*u*Math.PI/16)*Math.cos((2*y+1)*v*M
ath.PI/16);
```

```
        }
        result[x*8+y] = (1/4)*sum;
    }

    return result;
}

function quantization_matrix(multiply){
    /*
    return a quantization matrix with given multiply. pre-defined Q from

https://en.wikipedia.org/wiki/Quantization_(image_processing)#Quantization
_matrices
    */

    var Q = [
        16, 11, 10, 16, 24, 40, 51, 61,
        12, 12, 14, 19, 26, 58, 60, 55,
        14, 13, 16, 24, 40, 57, 69, 56,
        14, 17, 22, 29, 51, 87, 80, 62,
        18, 22, 37, 56, 68, 109, 103, 77,
        24, 35, 55, 64, 81, 104, 113, 92,
        49, 64, 78, 87, 103, 121, 120, 101,
        72, 92, 95, 98, 112, 100, 103, 99,
    ]
    for(var i=0; i<64; i++){
        Q[i] *= multiply;
    }
    return Q
}

function quantize_diff(multiply, loc, mat, encode_bits){
    /* quantize the size 64 (8*8) matrix.
```

```
    Input:
        multiply (int): the multiply for quantization matrix Q. Larger value
is more robust but changes more image details.
        loc (array): where to apply quantization.
        mat (array of size 64): the matrix.
        encode_bits (0/1 bit array with same size as loc)
    Output:
        diff (array of size 64): the diff to be added to original array for
stego
    */
    if(loc.length != encode_bits.length) throw "LOC and ENCODE_BITS have
different sizes! This is a bug in code!";
    var Q = quantization_matrix(multiply);
    var result = Array(64).fill(0);
    var div_Q, low, high;
    for(var i=0; i<loc.length; i++){
        div_Q = mat[loc[i]] / Q[loc[i]];
        low = Math.floor(div_Q);
        if(Math.abs(low % 2) != encode_bits[i]) low-=1;
        high = Math.ceil(div_Q);
        if(Math.abs(high % 2) != encode_bits[i]) high+=1;
        if(div_Q - low > high - div_Q) low = high;
        result[loc[i]] = low * Q[loc[i]] - mat[loc[i]];
    }
    return result;
}

function get_bit_from_quantized(multiply, loc, quantized_mat){
    /* get bits from quantized size 64 (8*8) matrix.
    Input:
        multiply (int): the multiply for quantization matrix Q. Larger value
is more robust but changes more image details.
        loc (array): where quantization is applied.
```

```
        quantized_mat (array of size 64): the matrix.
    Output:
        bits (array of size loc.length): the extracted bits
    */
    var Q = quantization_matrix(multiply);
    var result = [];
    for(var i=0; i<loc.length; i++){
        result.push(Math.abs(Math.round(quantized_mat[loc[i]] / Q[loc[i]])
% 2));
    }
    return result;
}

function img_16x16_to_8x8(mat){
    /* Resize image from 16 * 16 to 8 * 8
    Input:
        mat (size 256)
    Output:
        out_mat (size 64)
    */
    var result = Array(64);
    for(var i=0; i<8; i++) for(var j=0; j<8; j++){
        result[i*8+j] = (mat[i*2*8 + j*2] + mat[(i*2+1) * 8 + j*2] +
mat[i*2*8 + j*2 + 1] + mat[(i*2+1) * 8 + j*2 + 1]) / 4;
    }
    return result;
}

function img_8x8_to_16x16(mat){
    /* Resize image from 8 * 8 to 16 * 16
    Input:
        mat (size 64)
    Output:
```

```
        out_mat (size 256)
    */
    var result = Array(256);
    for(var i=0; i<16; i++) for(var j=0; j<16; j++){
        result[i*16+j] = mat[Math.floor(i/2) * 8 + Math.floor(j/2)];
    }
    return result;
}

function rgbclip(a){
    a=Math.round(a);
    a=(a>255)?255:a;
    return (a<0)?0:a;
}

function str_to_bits(str, num_copy)
{
    var utf8array=utf8Encode(str);
    var result=Array();
    var utf8strlen=utf8array.length;
    for(var i=0;i<utf8strlen;i++){
        for(var j=128; j>0; j=Math.floor(j/2))
        {
            if(Math.floor(utf8array[i]/j))
            {
                for(var cp=0; cp<num_copy; cp++) result.push(1);
                utf8array[i] -=j;
            } else for(var cp=0; cp<num_copy; cp++) result.push(0);
        }
    }
    for(var j=0;j<24;j++) for(var i=0;i<num_copy;i++) {
        result.push(1);
    }
```

```
    return result;
}

function bits_to_str(bitarray, num_copy)
{
    function merge_bits(bits){
        var bits_len = bits.length;
        var bits_sum = 0;
        for(var i=0;i<bits_len;i++) bits_sum += bits[i];
        return Math.round(bits_sum / bits_len);
    }

    var msg_array=Array();
    var data, tmp;

    var msg_array_len=Math.floor(Math.floor(bitarray.length/num_copy)/8);
    for(var i=0; i<msg_array_len; i++){
        data = 0;
        tmp = 128;
        for(var j=0; j<8; j++){
            data += merge_bits(bitarray.slice((i*8 + j) *num_copy, (i*8 +
j + 1) *num_copy))*tmp;
            tmp = Math.floor(tmp/2);
        }
        if(data == 255) break; //END NOTATION
        msg_array.push(data);
    }

    return utf8Decode(msg_array);
}

function extract_block(mat, block_size, x_min, y_min, img_num_col){
    var result = Array(block_size * block_size);
```

```
    for(var i=0; i<block_size; i++) for(var j=0; j<block_size; j++){
        result[i*block_size + j] = mat[(x_min+i)*img_num_col + y_min+j];
    }
    return result;
}

function replace_block(mat, block_size, x_min, y_min, img_num_col,
new_data){
    for(var i=0; i<block_size; i++) for(var j=0; j<block_size; j++){
        mat[(x_min+i)*img_num_col + y_min+j] = new_data[i*block_size + j];
    }
}
function utf8Decode(bytes) {
  var chars = [], offset = 0, length = bytes.length, c, c2, c3;

  while (offset < length) {
    c = bytes[offset];
    c2 = bytes[offset + 1];
    c3 = bytes[offset + 2];

    if (128 > c) {
      chars.push(String.fromCharCode(c));
      offset += 1;
    } else if (191 < c && c < 224) {
      chars.push(String.fromCharCode(((c & 31) << 6) | (c2 & 63)));
      offset += 2;
    } else {
      chars.push(String.fromCharCode(((c & 15) << 12) | ((c2 & 63) << 6) |
(c3 & 63)));
      offset += 3;
    }
  }
```

```
  return chars.join('');
}

function utf8Encode(str) {
  var bytes = [], offset = 0, length, char;

  str = encodeURI(str);
  length = str.length;

  while (offset < length) {
    char = str[offset];
    offset += 1;

    if ('%' !== char) {
      bytes.push(char.charCodeAt(0));
    } else {
      char = str[offset] + str[offset + 1];
      bytes.push(parseInt(char, 16));
      offset += 2;
    }
  }

  return bytes;
}

//read image functions
function prepare_read_data(data_bits, enc_key){
    var data_bits_len = data_bits.length;
    var result=Array(data_bits_len);
    var order = get_hashed_order(enc_key, data_bits_len);

    for(var i=0; i<data_bits_len; i++) result[i] = data_bits[order[i]];
```

```
    return result;
}

function get_bits_lsb(imgData){
    var result=Array();
    for (var i=0;i<imgData.data.length;i+=4)
    {
        result.push((imgData.data[i]%2==1)?1:0);
        result.push((imgData.data[i+1]%2==1)?1:0);
        result.push((imgData.data[i+2]%2==1)?1:0);
    }
    return result;
}

function get_dct_y(channel_data, channel_width, channel_length, multiply,
loc){
    /* get bits from Y channel
    Input:
        channel_data (1D array of size (channel_width * channel_length)):
manipulated data
        channel_width (int): channel width
        channel_length (int): channel length
        multiply (int): int for Q matrix to be multiplied
        loc (1D array of int): which location on block to stego on.
    Output:
        bits_stream.
    */

    var row_block = Math.floor(channel_length / 8);
    var col_block = Math.floor(channel_width / 8);
    var num_block_bits = loc.length;
    var result=Array();
    var reference_dct_block;
```

```
    for(var i=0; i<row_block; i++) for(var j=0; j<col_block; j++){
        var block_y = extract_block(channel_data, 8, i*8, j*8,
channel_width);
        var dct_y = dct(block_y);
        if(i==0 && j==0){
            reference_dct_block = dct_y;
            continue;
        }
        result.push(get_bit_from_quantized(multiply, loc,
            dct_y.map(function (num, idx) {return num -
reference_dct_block[idx];})));
    }

    return [].concat.apply([], result);
}

function get_dct_CbCr(channel_data, channel_width, channel_length,
multiply, loc){
    /* get bits from CbCr channel
    Input:
        channel_data (1D array of size (channel_width * channel_length)):
manipulated data
        channel_width (int): channel width
        channel_length (int): channel length
        multiply (int): int for Q matrix to be multiplied
        loc (1D array of int): which location on block to stego on.
    Output:
        bits_stream.
    */

    var row_block = Math.floor(channel_length / 16);
    var col_block = Math.floor(channel_width / 16);
```

```
    var num_block_bits = loc.length;
    var result=Array();
    var reference_dct_block;

    for(var i=0; i<row_block; i++) for(var j=0; j<col_block; j++){
        var block_y = extract_block(channel_data, 16, i*16, j*16,
channel_width);
        block_y = img_16x16_to_8x8(block_y);
        var dct_y = dct(block_y);
        if(i==0 && j==0){
            reference_dct_block = dct_y;
            continue;
        }
        result.push(get_bit_from_quantized(multiply, loc,
            dct_y.map(function (num, idx) {return num -
reference_dct_block[idx];})));
    }
    return [].concat.apply([], result);
}

function get_bits_dct(imgData, channel_width, channel_length, multiply,
loc, use_y, use_downsampling){
    /* Get Stego from imgData using DCT
    Input:
        imgData: manipulated data
        channel_width (int): channel width
        channel_length (int): channel length
        multiply (int): int for Q matrix to be multiplied
        loc (1D array of int): which location on block to stego on.
        use_y (bool): whether to manipulate y channel
        use_downsampling(bool): whether to downsample on CrCb
    Output:
        bit_stream
```

```
    */

    var y=Array(), cb=Array(), cr=Array(), result=Array();
    for (var i=0;i<imgData.data.length;i+=4)
    {
        var ycbcr = rgb2ycbcr(imgData.data[i],imgData.data[i+1],imgData.data[i+2]);
        y.push(ycbcr[0]);
        cb.push(ycbcr[1]);
        cr.push(ycbcr[2]);
    }
    if(use_y) result.push(get_dct_y(y, channel_width, channel_length, multiply, loc));
    var cbcr_func = (use_downsampling)?get_dct_CbCr : get_dct_y;
    result.push(cbcr_func(cb, channel_width, channel_length, multiply, loc));
    result.push(cbcr_func(cr, channel_width, channel_length, multiply, loc));

    return [].concat.apply([], result);
}

// main function
function readMsgFromCanvas_base(canvasid, enc_key, use_dct, num_copy, multiply, loc, use_y, use_downsampling){
    /* Read message from canvas
    Input:
        canvasid: Canvas ID to read/write data
        enc_key (string): encryption key for msg
        use_dct (bool): use true for DCT, false for LSB
        num_copy (int): how many copies of each bit to write into image. Larger value is more robust but reduces capacity.
```

```
        -- below only valid for use_dct=true --

        multiply (int): int for Q matrix to be multiplied
        loc (1D array of int): which location on block to stego on.
        use_y (bool): whether to manipulate y channel
        use_downsampling(bool): whether to downsample on CrCb
    Output:
        [status, message]: status is a boolean: true means success and false
means failure.
            On success, message is the decrypted message and on failure,
message is the error message.
    */

    use_dct=(use_dct === undefined)?false:use_dct;
    enc_key=(enc_key === undefined)?'':enc_key;
    num_copy=(num_copy === undefined)?5:num_copy;
    multiply=(multiply=== undefined)?30:multiply;
    loc=(loc === undefined)? [1,2,8,9,10,16,17]:loc;
    use_y=(use_y === undefined) ? true: use_y;
    use_downsampling=(use_downsampling    ===    undefined)    ?    true:
use_downsampling;

    var c, ctx, imgData;

    try{
        c=document.getElementById(canvasid);
        ctx=c.getContext("2d");
        imgData=ctx.getImageData(0,0,c.width,c.height);
    }
    catch(err){
        return [false, err];
    }
```

```
    try{
        var bits_stream = (use_dct)?get_bits_dct(imgData, c.width,
c.height, multiply, loc, use_y, use_downsampling):get_bits_lsb(imgData);
        bits_stream = prepare_read_data(bits_stream, enc_key);
        var msg = bits_to_str(bits_stream, num_copy);
        if(msg==null) return [false,
            "Message does not decrypt. Maybe due to (1) wrong password /
enc method. (2) corrupted file"];
        return [true, msg];
    }
    catch(err){
        return [false, "Message does not decrypt. Maybe due to (1) wrong
password / enc method. (2) corrupted file"];
    }
}

//write image
function prepare_write_data(data_bits, enc_key, encode_len){
    var data_bits_len = data_bits.length;
    if(data_bits.length > encode_len) throw "Can not hold this many data!";
    var result=Array(encode_len);
    for(var i=0; i<encode_len; i++){
        result[i] = Math.floor(Math.random()*2); //obfuscation
    }

    var order = get_hashed_order(enc_key, encode_len);
    for(var i=0; i<data_bits_len; i++) result[order[i]] = data_bits[i];

    return result;
}

function write_dct_y(channel_data, channel_width, channel_length, setdata,
multiply, loc){
```

```
    /* write a DCT manipulated Y channel from original Y channel
    Input:
        channel_data (1D array of size (channel_width * channel_length)): original Y data
        channel_width (int): channel width
        channel_length (int): channel length
        setdata (1D array of bits 0/1 array): data to stego
        multiply (int): int for Q matrix to be multiplied
        loc (1D array of int): which location on block to stego on.
    */

    var row_block = Math.floor(channel_length / 8);
    var col_block = Math.floor(channel_width / 8);
    var num_block_bits = loc.length;
    if( num_block_bits * (row_block * col_block - 1)!= setdata.length) throw "Image size does not match data size (Y channel)";
    var reference_dct_block;

    for(var i=0; i<row_block; i++) for(var j=0; j<col_block; j++){
        var block_y = extract_block(channel_data, 8, i*8, j*8, channel_width);
        var dct_y = dct(block_y);
        if (i==0 && j==0){
            reference_dct_block = dct_y;
            continue;
        }
        var dct_diff = dct_y.map(function (num, idx) {return num - reference_dct_block[idx];});
        var qdiff = quantize_diff(multiply, loc, dct_diff, setdata.slice(num_block_bits * (i*col_block + j - 1), num_block_bits * (i*col_block + j)));
        dct_y = dct_y.map(function (num, idx) {return num + qdiff[idx];});
        block_y = idct(dct_y);
```

```
        //replace original block with stego Y
        replace_block(channel_data, 8, i*8, j*8, channel_width, block_y);
    }
}

function write_dct_CbCr(channel_data, channel_width, channel_length, setdata, multiply, loc){
    /* get a DCT manipulated Cb or Cr channel from original channel
    Input:
        channel_data (1D array of size (channel_width * channel_length)): original CbCr data
        channel_width (int): channel width
        channel_length (int): channel length
        setdata (1D array of bits 0/1 array): data to stego
        multiply (int): int for Q matrix to be multiplied
        loc (1D array of int): which location on block to stego on.
    */

    var row_block = Math.floor(channel_length / 16);
    var col_block = Math.floor(channel_width / 16);
    var num_block_bits = loc.length;
    if( num_block_bits * (row_block * col_block - 1) != setdata.length) throw "Image size does not match data size (CbCr channel)";
    var reference_dct_block;

    for(var i=0; i<row_block; i++) for(var j=0; j<col_block; j++){
        var block_y = extract_block(channel_data, 16, i*16, j*16, channel_width);
        var block_y_8x8 = img_16x16_to_8x8(block_y);
        var dct_y = dct(block_y_8x8);
        if (i==0 && j==0){
            reference_dct_block = dct_y;
            continue;
```

```
        }
        var dct_diff = dct_y.map(function (num, idx) {return num -
reference_dct_block[idx];});
        var qdiff = quantize_diff(multiply, loc, dct_diff,
setdata.slice(num_block_bits * (i*col_block + j - 1), num_block_bits *
(i*col_block + j)));
        dct_y = dct_y.map(function (num, idx) {return num + qdiff[idx];});
        var block_y_stego = idct(dct_y);
        var stego_diff = block_y_stego.map(function (num, idx) {return num
- block_y_8x8[idx];});
        stego_diff = img_8x8_to_16x16(stego_diff);
        block_y = block_y.map(function (num, idx) {return num +
stego_diff[idx];});

        //replace original block with stego Y
        replace_block(channel_data, 16, i*16, j*16, channel_width,
block_y);
    }
}


function write_lsb(imgData,setdata) {
    function unsetbit(k){
        return (k%2==1)?k-1:k;
    }

    function setbit(k){
        return (k%2==1)?k:k+1;
    }
    var j=0;
    for (var i=0;i<imgData.data.length;i+=4)
    {
```

```
        imgData.data[i] = (setdata[j])?setbit(imgData.data[i]):unsetbit(imgData.data[i]);
        imgData.data[i+1] = (setdata[j+1])?setbit(imgData.data[i+1]):unsetbit(imgData.data[i+1]);
        imgData.data[i+2] = (setdata[j+2])?setbit(imgData.data[i+2]):unsetbit(imgData.data[i+2]);
        imgData.data[i+3]=255;
        j+=3;
    }
}

function dct_data_capacity(channel_width, channel_length, loc, use_y, use_downsampling){
    var y_data_len = (use_y)?(Math.floor(channel_length / 8) * Math.floor(channel_width / 8) - 1)* loc.length : 0;
    var cblock = (use_downsampling)? 16 : 8;
    var cbcr_data_len = (Math.floor(channel_length / cblock) * Math.floor(channel_width / cblock) - 1) * loc.length;
    return [y_data_len, cbcr_data_len];
}

function write_dct(imgData, channel_width, channel_length, setdata, multiply, loc, use_y, use_downsampling){
    /* Write Stego to imgData using DCT
    Input:
        imgData: to manipulate
        channel_width (int): channel width
        channel_length (int): channel length
        setdata (1D array of bits 0/1 array): data to stego
        multiply (int): int for Q matrix to be multiplied
        loc (1D array of int): which location on block to stego on.
        use_y (bool): whether to manipulate y channel
        use_downsampling (bool): whether to downsample on CrCb
```

```
    */
    var data_capacity = dct_data_capacity(channel_width, channel_length,
loc, use_y, use_downsampling);
    var y_data_len = data_capacity[0];
    var cbcr_data_len = data_capacity[1];

    var y=Array(), cb=Array(), cr=Array();
    for (var i=0;i<imgData.data.length;i+=4)
    {
        var ycbcr =
rgb2ycbcr(imgData.data[i],imgData.data[i+1],imgData.data[i+2]);
        y.push(ycbcr[0]);
        cb.push(ycbcr[1]);
        cr.push(ycbcr[2]);
    }
    if(use_y) write_dct_y(y, channel_width, channel_length,
setdata.slice(0, y_data_len), multiply, loc);
    var cbcr_func = (use_downsampling)?write_dct_CbCr : write_dct_y;

    cbcr_func(cb, channel_width, channel_length, setdata.slice(y_data_len,
y_data_len + cbcr_data_len), multiply, loc);
    cbcr_func(cr, channel_width, channel_length,
        setdata.slice(y_data_len + cbcr_data_len, y_data_len +
cbcr_data_len + cbcr_data_len), multiply, loc);
    var j=0;
    for (var i=0;i<imgData.data.length;i+=4)
    {
        var rgb = ycbcr2rgb(y[j], cb[j], cr[j]);
        imgData.data[i] = rgbclip(rgb[0]);
        imgData.data[i+1] = rgbclip(rgb[1]);
        imgData.data[i+2] = rgbclip(rgb[2]);
        j+=1;
    }
```

```
}

// main function
function writeMsgToCanvas_base(canvasid, msg, enc_key, use_dct, num_copy,
multiply, loc, use_y, use_downsampling){
    /* Write message to canvas
    Input:
        canvasid: Canvas ID to read/write data
        msg (string): message to stego
        enc_key (string): encryption key for msg
        use_dct (bool): use true for DCT, false for LSB
        num_copy (int): how many copies of each bit to write into image.
Larger value is more robust but reduces capacity.

        -- below only valid for use_dct=true --


        multiply (int): int for Q matrix to be multiplied
        loc (1D array of int): which location on block to stego on.
        use_y (bool): whether to manipulate y channel
        use_downsampling(bool): whether to downsample on CrCb
    Output:
        isSuccess: === true: success, otherwise, a string with error
message.
    */

    use_dct=(use_dct === undefined)?false:use_dct;
    enc_key=(enc_key === undefined)?'':enc_key;
    num_copy=(num_copy === undefined)?5:num_copy;
    multiply=(multiply=== undefined)?30:multiply;
    loc=(loc === undefined)? [1,2,8,9,10,16,17]:loc;
    use_y=(use_y === undefined) ? true: use_y;
```

```
    use_downsampling=(use_downsampling === undefined) ? true: use_downsampling;

    try{
        var c=document.getElementById(canvasid);
        var ctx=c.getContext("2d");
        var imgData=ctx.getImageData(0,0,c.width,c.height);

        var encode_len = Math.floor(imgData.data.length / 4) * 3;
        if(use_dct){
            var cap = dct_data_capacity(c.width, c.height, loc, use_y, use_downsampling);
            encode_len = cap[0] + 2 * cap[1];
        }
        // prepare data
        var bit_stream = str_to_bits(msg, num_copy);
        bit_stream = prepare_write_data(bit_stream, enc_key, encode_len);
        if(use_dct){
            write_dct(imgData, c.width, c.height, bit_stream, multiply, loc, use_y, use_downsampling);
        } else write_lsb(imgData, bit_stream);
        ctx.putImageData(imgData,0,0);
        return true;
    }
    catch(err){
        return err;
    }
}
```

Додаток Б

Презентація дипломної роботи магістра

Актуальність

Потреба застосування та впровадження в медичні інформаційні системи технології маркування та перевірки автентичності медичних даних обумовлення стрімким ростом кількості відповідних суб'єктів, таких як: приватні та державні лікарні, медичні центри, лабораторії, госпіталів та їх діяльності в МІС, а саме розповсюдження медичних даних, які повинні бути захищеними для кінцевого їх отримувача, тобто автентичним .

Інформаційні медичні системи, здебільшого використовують застаріле програмне забезпечення, яке з кожним роком все складніше підтримувати або взагалі не підтримується розробниками на даний момент. Відповідно, застарілість ПЗ створює проблему низького рівня захисту відповідних систем в порівнянні з теперішнім рівнем розвитку технологій. Найбільш не захищені є дані які передаються через сервіси цих систем що становить загрозу не тільки приватності, а що більше важливо – можливої фальсифікації чи модифікації цих даних третіми особами, що викликає втрату цілісності самих даних.

Це і призвело до необхідності дослідження та розробки відповідної ІС на основі існуючих методів та їх модифікації з подальшим впровадженням в медичні інформаційні системи.

Мета і завдання

Мета роботи полягає у реалізації інформаційної технології маркування та перевірки автентичності медичних зображень но основі ЦВЗ з покращеним рівнем захисту.

Для досягнення поставленої мети визначенні наступні задачі дослідження:

- Провести аналіз існуючих методів технологій та рішень маркування та перевірки автентичності медичних зображень на основі ЦВЗ;
- Удосконалення існуючих методів маркування та перевірки автентичності медичних зображень на основі ЦВЗ у рамка покращення рівня захисту;
- Розробити інформаційну технологію маркування та перевірки автентичності медичних зображень за допомогою отриманих моделей та методів
- Виконати експериментальну перевірку маркування та перевірки автентичності медичних зображень

Об’єкт та предмет дослідження

Об’єкт дослідження: Процес маркування та перевірки автентичності медичних зображень.

Предмет дослідження: Моделі методи підходи та засоби інформаційної технології маркування та перевірки автентичності медичних зображень

Наукова новизна

В результаті проведеної роботи було удосконалено існуючі методи маркування та перевірки автентичності медичних зображень на основі ЦВЗ у рамках покращення рівня захисту;

Практична цінність

В результаті виконання дипломної роботи магістра реалізоване відповідне програмне забезпечення, яке підтвердило вірність запропонованих положень.

Застосування інформаційної технології дає можливість маркувати та перевіряти автентичність медичних зображень з підвищеним ступенем захисту.

Апробація результатів дипломної роботи магістра

Основні наукові та практичні результати опубліковані в фаховому науковому виданні:

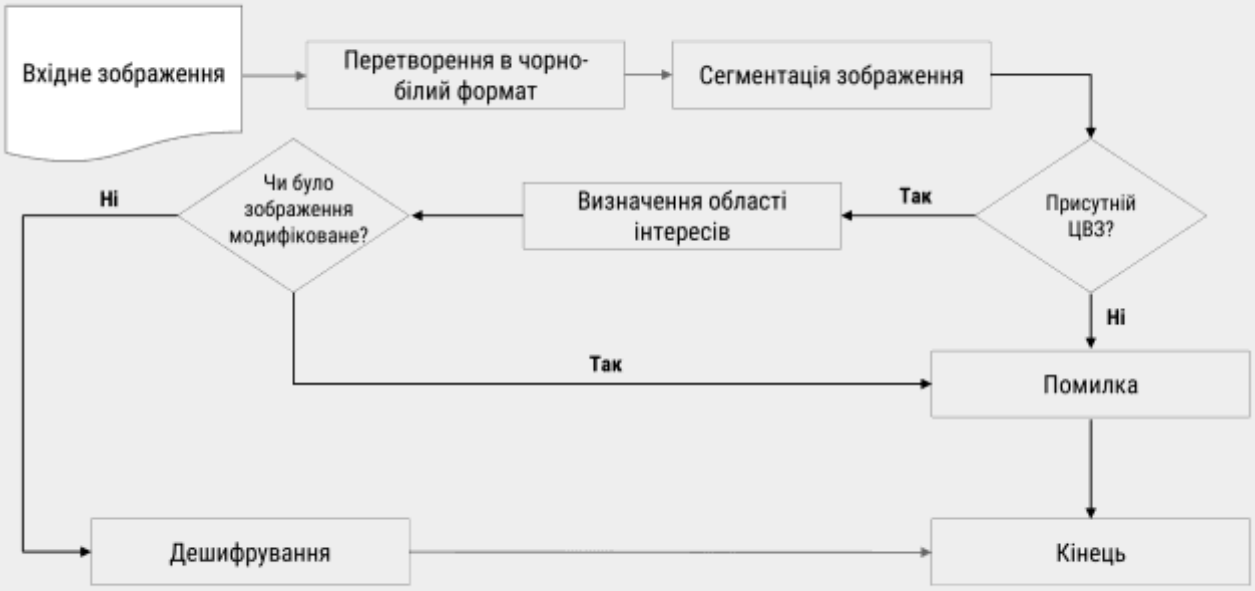
– стаття на тему «**Сегментація медичних зображень**» у журналі «Вісник Хмельницького Національно Університету»

Загальний опис технології маркування та перевірки автентичності медичних зображень


Отримання та сегментація вхідного зображення
Вхідне зображення
Тип 1
Тип 2
Тип 3
Тип 4
Властивості
Розмір зображення
Глибина кольору
Колірний простір
Роздільна здатність
Області Інтересів зображення
Чорно-біле зображення
Метод Росту Регіонів (RG)

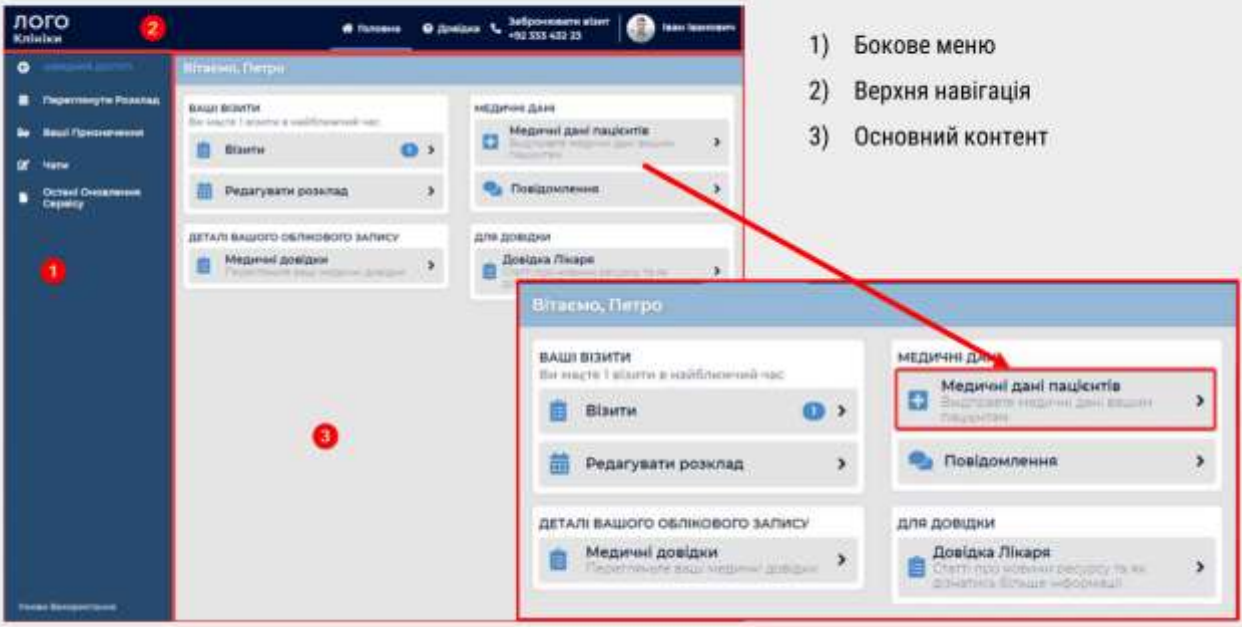

Шифрування Області Інтересів, Медичних Даних та нанесення ЦВЗ
Медичні Дані
Хешовані Медичні Дані
Шифровані Медичні Дані
Текст для вставки в зображення
Нанесення ЦВЗ
Області Інтересів
SHA-256
Хешовані області інтересів

Перевірка автентичності та зчитування ЦВЗ

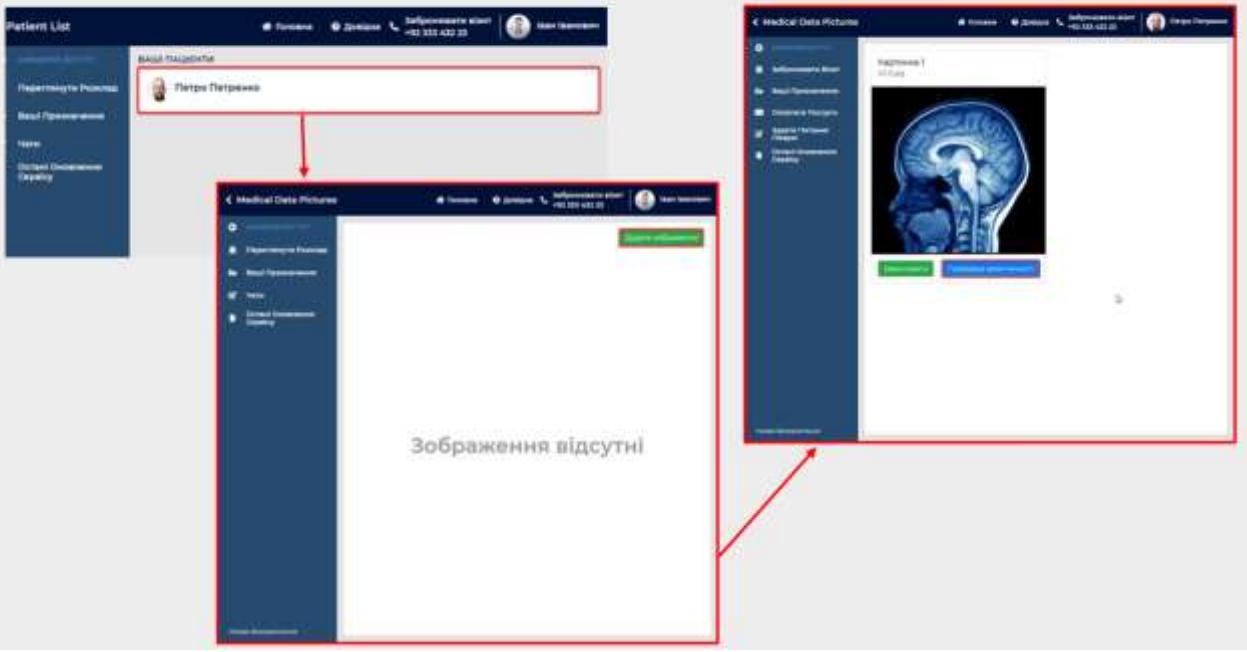


Екран входу в систему

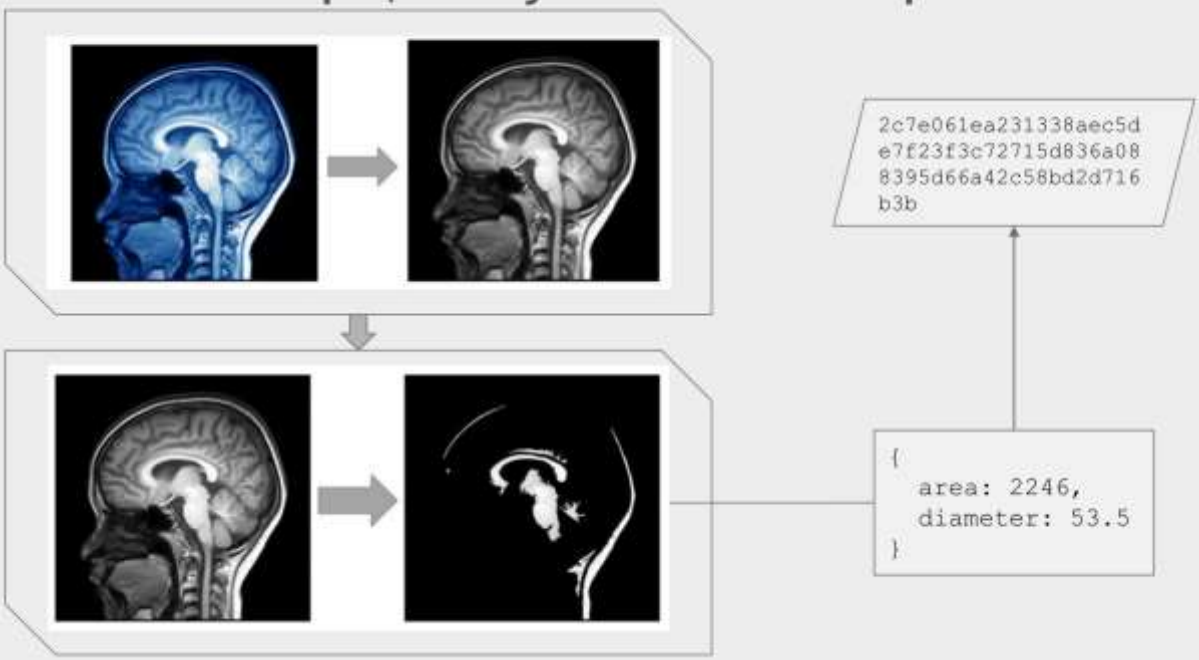
Інтерфейс користувача (лікар або пацієнт)



Процес завантаження медичного зображення



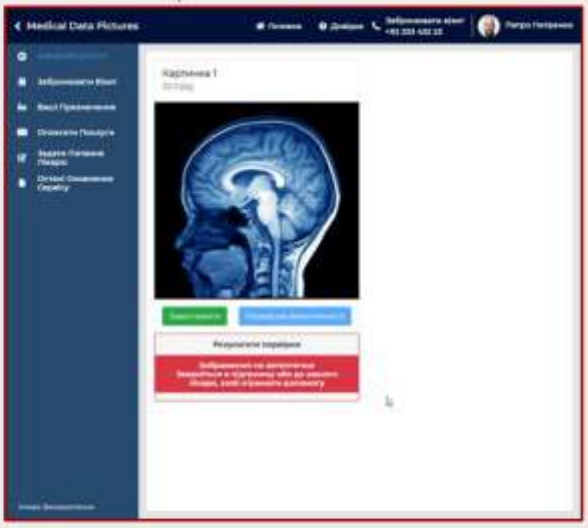
Процес хешування області інтересів



Перевірка автентичності

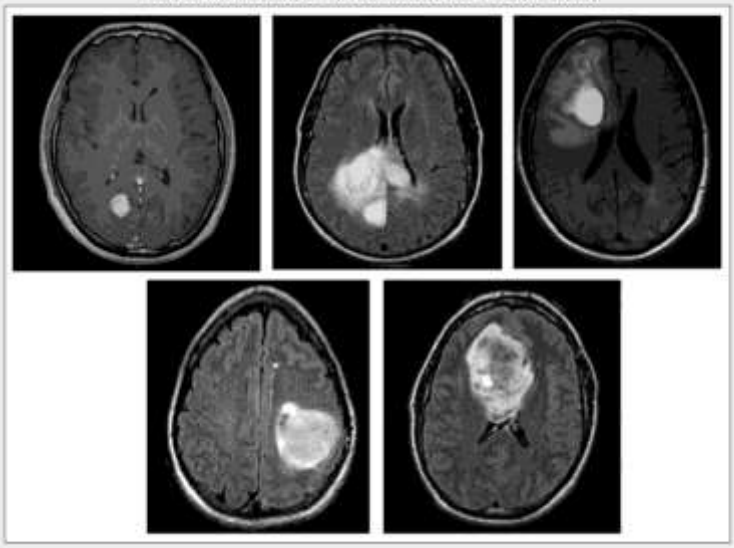
Зображення автентичне

Зображення неавтентичне



Перевірка запропонованого методу (Вхідні дані)

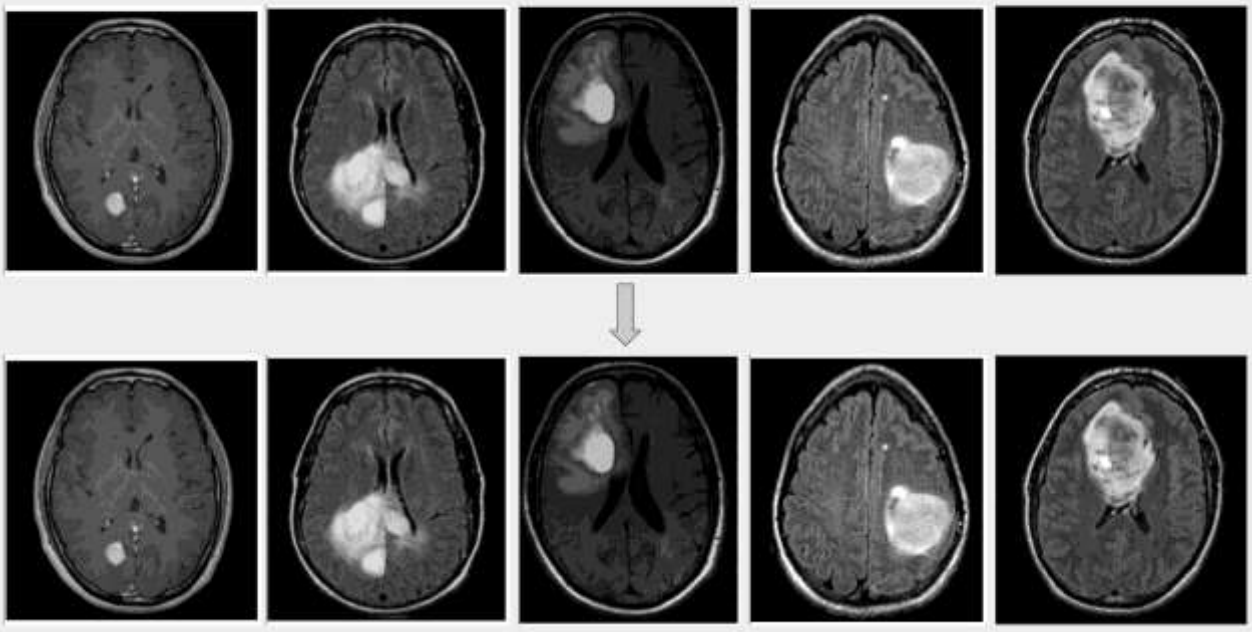
Вхідні медичні зображення для тестування



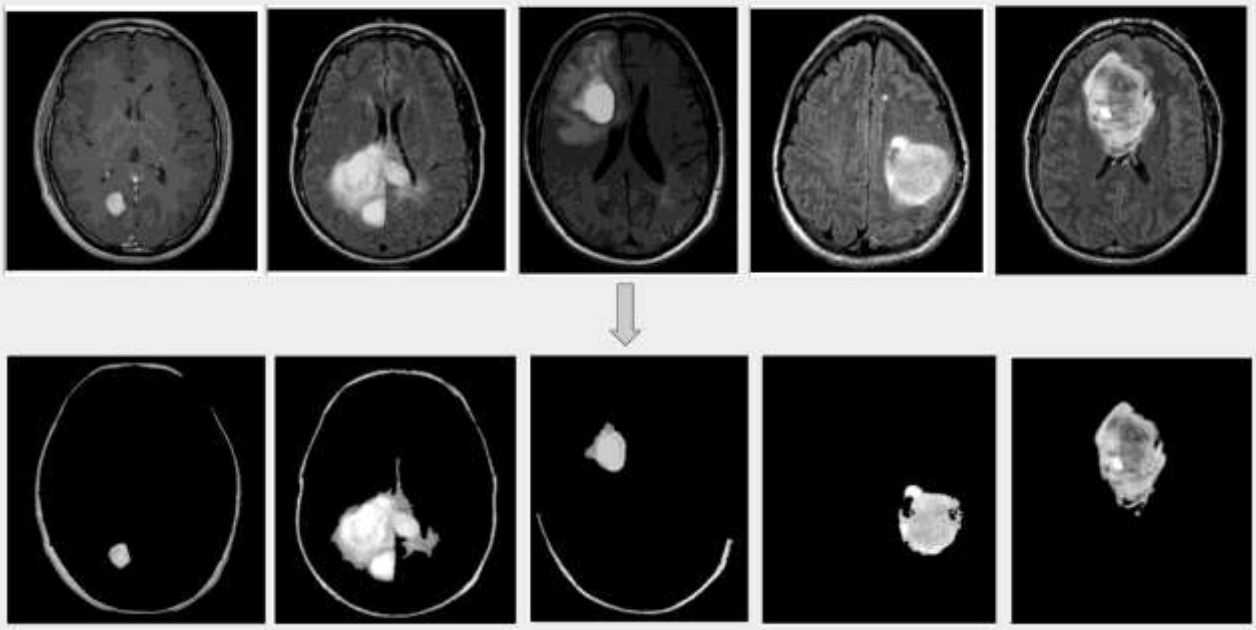
Вхідні медичні дані для тестування

```
{
  Patient: "Петро Петренко",
  PatientID: 43252124,
  Doctor: "Іван Іванович",
  DoctorID: 231231,
  MedicalCenter: "Тестова клініка",
  AdditionalInfo: "Тест"
}
```

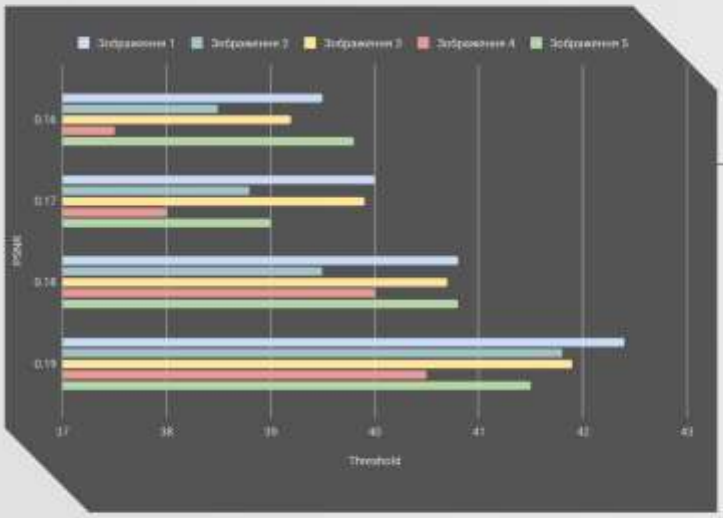
Перевірка запропонованого методу (Візуальна оцінка нанесеного ЦВЗ)



Перевірка запропонованого методу (Результат сегментації)



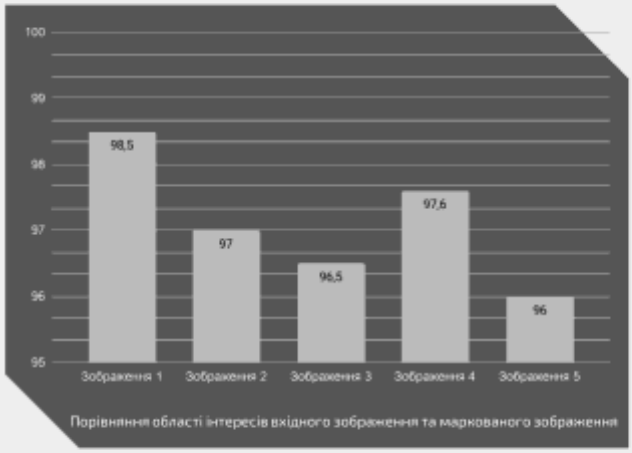
Перевірка запропонованого методу (Сегментація)



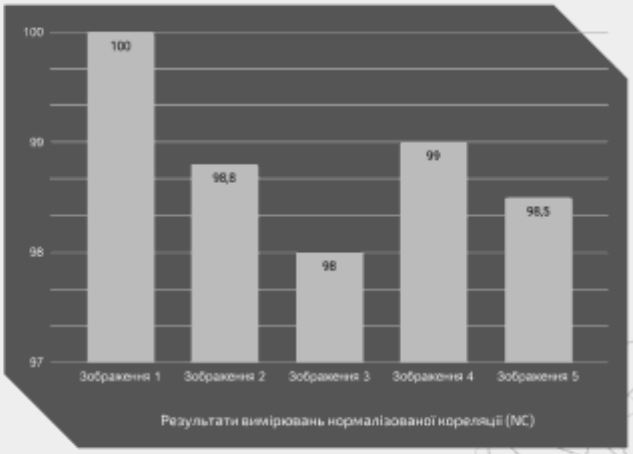
Результати тестування ефективності запропонованого методу на основі PSNR шляхом порогування

У запропонованій методології для сегментації використовується алгоритм **росту регіонів (RG)**. В алгоритмі RG продуктивність змінюється на основі **порогового значення (threshold)**. Згідно з аналізом, значення PSNR поступово збільшується, коли порогове значення продовжує зростати. Значення PSNR є низьким, коли порогове значення становить 0,16; аналогічно, значення PSNR є високим, коли порогове значення становить 0,19. Тут запропонований підхід досягає максимального значення PSNR у 42,2361 db.

Перевірка запропонованого методу (Статистичні Дані)

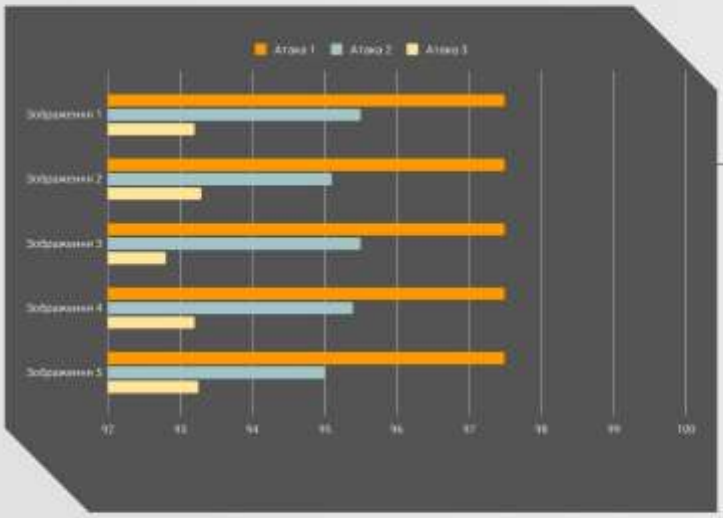


Порівняння області інтересів вхідного зображення та маркованого зображення



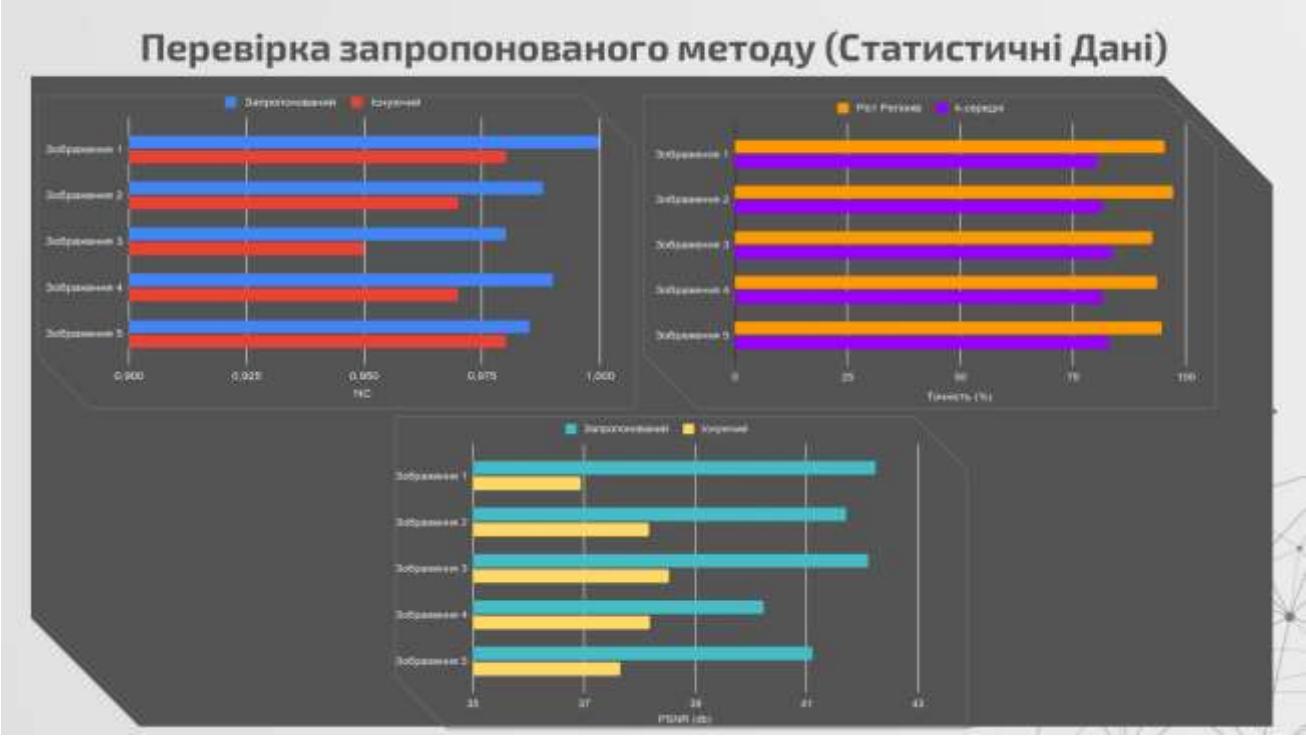
Результати вимірювань нормалізованої кореляції (NC)

Перевірка запропонованого методу (Статистичні Дані)



Ефективність запропонованої техніки з використанням атак різного виду

Атака 1 являє собою зміну п'яти пікселів і застосовується алгоритм шифрування, а потім вимірюється результат. Отримані майже однакові значення. Таким чином, ця атака не впливає належним чином на результати. Подібним чином змінюються значення пікселів і тестуються отримані результати. Після застосування атаки метод показує кращі результати.



Висновки

Згідно до завдань дипломної роботи розглянуто основні аспекти та процеси маркування та перевірки автентичності даних.

На основі поставлених завдань та проведеного аналізу наукових робіт представлено інформаційну технологію маркування та перевірки автентичності медичних зображень та основну структуру інформаційної системи, основні алгоритми та методи нанесення цифрового водяного знаку. Також за допомогою таких методів як: метод росту регіонів (ROI), еліптичної криптографії та шифрування SHA–256, було запроваджено багатошаровий рівень захисту на кожному з етапів роботи з медичним зображенням та відповідними даними.

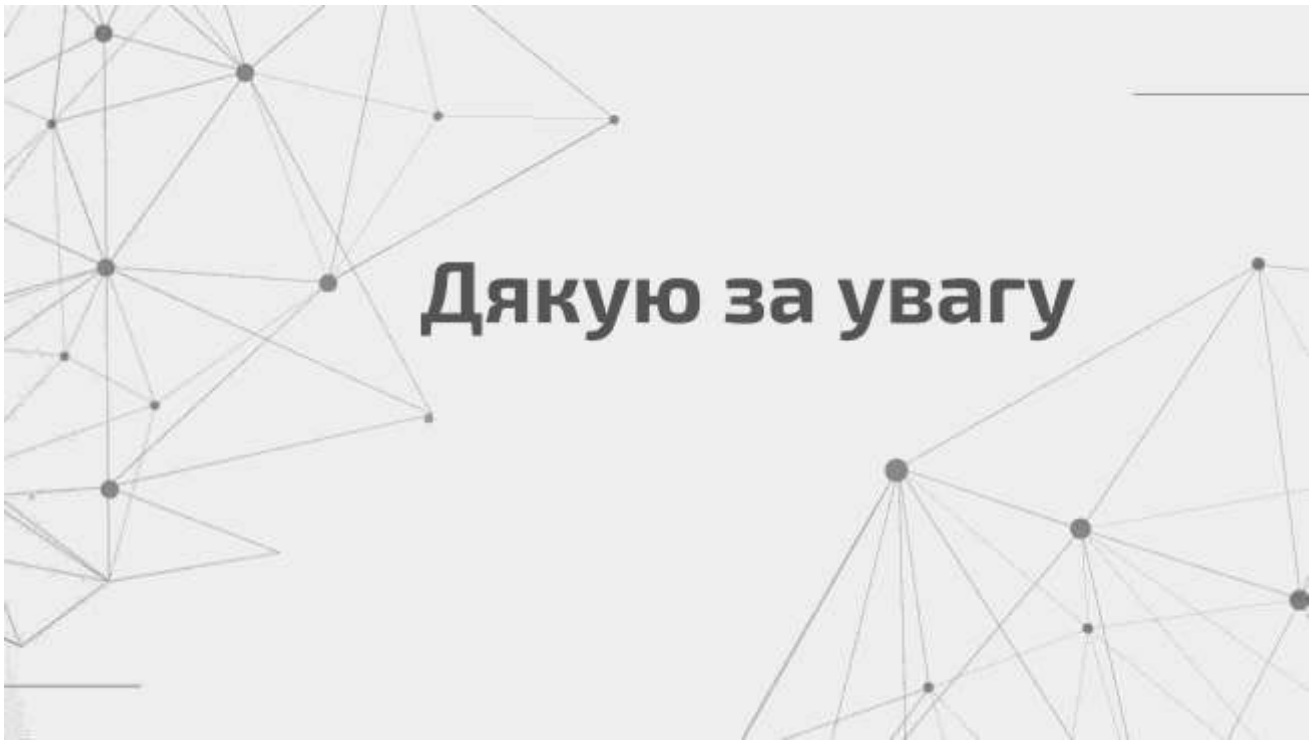
Також представлено основні алгоритми та інтерфейси для маркування та перевірки автентичності медичних даних. Описані відповідні поля інтерфейсів та відповідні методи. Побудовано схеми оптимального використання та об'єднання відповідних алгоритмів та найкращу їх послідовність.

Представлено інноваційний підхід нанесения водяних знаків на медичні зображення. Проаналізовано та перевірено різні алгоритми які запроваджують додаткові рівні безпеки або є допоміжними для інших методів, а саме: RG, SHA–256, ECC та AC. Встановлено, що складність обчислень запропонованої інформаційної технології менша, оскільки вона використовує прості математичні розрахунки для формування даних автентифікації та відновлення та для відновлення зашифрованих даних.

Запропонована методологія підтримує якість зображення з цифровим водяним знаком із середнім значенням PSNR 42,23 db, потужністю вбудовування 72 384 біт, вилученою точністю 98% і NC.

Отримані експериментальні результати апробації вказують на те, що запропонована інформаційна технологія забезпечує кращу якість нанесення ЦВЗ на зображення порівнюючи з іншим та краще показує себе у кращій продуктивності вбудовування. Представлена інформаційна технологія може бути практично включена до медичних інформаційних систем для забезпечення цілісності медичного зображення, автентифікації системи та конфіденційності.

Отже, в результаті виконання поставлених задач та реалізації відповідної інформаційної технології було досягнуто поставленої мети дипломної роботи магістра


Дякую за увагу

Додаток В

Ксерокопії наукових публікацій, виконаних при роботі над дипломною роботою магістра

Перелік наукових публікацій:

1. Мостовий В. В, Горященко С. Л.. Сегментація медичних зображень / Мостовий В. В, Горященко С. Л // Науковий журнал «Вісник Хмельницького національного університету» серія: Технічні науки. Хмельницький, 2020, №5.

УДК 004.93+616-018

В. В. МОСТОВИЙ, С.Л. ГОРЯЩЕНКО
Хмельницький національний університет

СЕГМЕНТАЦІЯ МЕДИЧНИХ ЗОБРАЖЕНЬ

В роботі розглянуто та проаналізовано можливості застосування методів сегментації на основі ознак зв'язаності для інших типів зображень, проведений аналітичний огляд і наведена класифікація відомих методів сегментації, на підставі чого сформульовані вимоги щодо розробки структурних моделей для задач сегментації мікроскопічних медичних зображень, обґрунтована актуальність використання ознаки зв'язаності щодо задач сегментації і побудовані її математичні моделі.

Ключові слова: сегментація, математичні моделі сегментації, імітаційне моделювання

V. V. MOSTOVYI, S. L. HORIASHCHENKO
Khmelnytsky National University, Ukraine

SEGMENTATION OF MEDICAL IMAGES

Abstract .– The article considers and analyzes the possibility of applying segmentation methods based on signs of connectivity for other types of images, conducted an analytical review and classification of known segmentation methods, based on which the requirements for developing structural models for segmentation of microscopic medical images, substantiated the relevance of the feature connections on segmentation problems and its mathematical models are built.

Keywords: segmentation, mathematical models of segmentation, simulation modeling

Вступ

Сегментація є складовою частиною процесу цифрової обробки зображень. Він є поділенням або ж розбиттям зображення на деякі частини, що відповідають заданим ознакам та характеризують ці області та зображення загалом. На етапі сегментації вирішуються питання, які доповнюють стандартні задачі обробки зображення, а саме кодування, реставрація, покращення якості.

Процес сегментації розглядається як невід'ємна частина задач розпізнавання, класифікації та ідентифікації зображень [4]. Саме тому сегментація знайшла своє широке застосування в таких сферах як мікробіологія, медицина, астрономія, військова техніка і решті сфер життєдіяльності людини. Також такі дослідження допомагають психологам та фізіологам у вивченні таких процесів, як сприйняття форм, навчання і розпізнавання об'єктів живими організмами та мозком людини, тощо.

Сегментація широко застосовується при автоматизації мікроскопічних обстежень різноманітних медичних об'єктів, до яких можна віднести опрацювання зображень клітин організмів та їхніх складових і гемоцитологічних препаратів. Цей процес є складовою частиною розпізнавання та класифікації у медичній діагностиці.

Останнім часом розпочато роботу щодо повної автоматизації процесу сегментації зображень біологічних об'єктів задля підвищення достовірності діагностування різного роду захворювань.

Інформація, яка отримується в результаті сегментації також використовується для виявлення впливів різного роду несприятливих факторів і допомагає прогнозувати протікання лейкозів, лімфосарком, анемій та інших захворювань людського організму [9].

Автоматизація процесів сегментації медичних зображень шляхом побудови відповідної системи на основі інформаційних технологій сприяє об'єктивізації отримуваних результатів з одночасним підвищенням їх достовірності у більш стислі терміни. Остання обставина є особливо актуальною в умовах масових захворювань (епідемій та пандемій), зокрема в зв'язку із поширенням пандемії COVID-19.

У провідних країнах світу ведуться дослідження зі створення пристроїв, які допомагають автоматизувати розпізнавання і вимірювання різного роду мікроскопічних зображень у медичних препаратах. Однак, задача повної автоматизації сегментації і отримання цитоморфологічних властивостей клітин та їхніх невід'ємних частин є складною з математичної і технічної точки зору.

Першочерговим аспектом задачі сегментації є математичне обґрунтування задачі та визначення способів і методів її розв'язання. Алгоритмічна та обчислювальна сторона задачі сегментації медичних зображень полягають у втіленні даного методу в сукупність алгоритмічних процедур та особливостей для їхнього відтворення.

Проаналізувавши стан теоретичних досліджень існуючих методів, що моделюють процес сегментації і розглянувши математичні моделі, які використовуються в процесі сегментації і їх практичне застосування в галузі досліджень і діагностики мікроскопічних зображень гемоцитологічних препаратів, було зроблено висновок, що не існує достатньо ефективного універсального методу сегментації.

На теперішній час розроблено багато методів сегментації та алгоритмів для їх реалізації, але, нажаль, ті які задовольняють заданій точності і достовірності є надзвичайно складними і потребують багато часу для їхнього втілення. В той же час, моделі які відрізняються простотою реалізації та своєю швидкодією, не дають потрібної точності та достовірності.

Таким чином, постає питання формулювання рекомендацій застосування тих чи інших методів сегментації зображень в конкретній практичній ситуації.

Експериментальна частина

Широкого поширення для задач сегментації медичних зображень набули моделі порогового обмеження, морфологічного градієнту та моделі нарощування областей. При порівнянні вище вказаних методів встановлено, що найбільш перспективним методом є метод нарощування областей, при цьому методі зображення представляється у вигляді сукупності структурно-зв'язаних областей, тобто враховується просторовий взаємозв'язок елементів зображення на відміну від порогової обробки. Саме це дає можливість в процесі сегментації обчислювати цитоморфометричні характеристики зображення, які можна використовувати в подальшій обробці при вирішенні задач розпізнавання, на відміну від морфологічного методу.

Основним недоліком методу нарощування областей є проблема вибору стартових точок, що заважає повній автоматизації процесу сегментації, до недоліків також можна віднести проблему вибору порогу. При цьому метод є емпіричним та не має чіткого теоретичного обґрунтування.

Задля вирішення зазначених вище проблем при таких структурних моделей зображення в роботі [2] запропоновано використовувати поняття зв'язності.

Як зазначено в роботі [2] поняття зв'язності доцільно застосовувати, як окрему ознаку зображення, що має широку інформативність, яка порівняна тільки з яскравістю, а математичне представлення цієї ознаки, розглядається як структурно-зв'язана модель бінарного зображення. з використанням понять усіченого зображення та бінарного зрізу.

Усіченим зображенням із номером k називається зображення, представлене у вигляді матриці $A^k(M,N)$, елементи якої $a^k(m,n)$ визначаються у результаті операцій віднімання дискретної величини d від елементів $a^{k-1}(m,n)$ матриці $A^{k-1}(M,N)$ зображення, причому $A^0(M,N)$ – матриця, що визначає вхідне зображення.

Зрізом з номером k називається сукупність бінарних елементів зображення $b^k(m,n)$, що представлена у вигляді матриці $B^k(M,N)$:

причому $b^k(m,n) = \begin{cases} 1, & \text{якщо } a^k(m,n) = 0 \\ 0, & \text{якщо } a^k(m,n) \neq 0 \end{cases}$

де $a^k(m,n)$ – елемент матриці усіченого зображення $A^k(M,N)$.

Зв'язаність $\delta^k(m,n)$ одиничного елемента $b^k(m,n)$ у межах даного зрізу за номером k розраховується сумою одиничних елементів, що з ним зв'язані, тобто маємо:

$$\delta^k(m,n)=b^k(m+1,n)+b^k(m-1,n)+b^k(m,n+1)+b^k(m,n-1)+b^k(m+1,n+1)+b^k(m-1,n-1)+b^k(m+1,n-1)+b^k(m-1,n+1),$$

$\forall b^k(m,n)=1$.

Зв'язаність $\delta^{k,k+1}(m,n)$ елемента $b^k(m,n)$ з елементами сусіднього зрізу за номером $k+1$ (або міжзрізова зв'язаність елементу) визначається за формулою:

$\delta^{k,k+1}(m,n)=b^{k+1}(m+1,n)+b^{k+1}(m-1,n)+b^{k+1}(m,n+1)+b^{k+1}(m,n-1)+b^{k+1}(m+1,n+1)+b^{k+1}(m-1,n-1)+b^{k+1}(m+1,n-1)+$
$+b^{k+1}(m-1,n+1), \quad \forall b^k(m,n)=1.$

Медичні зображення біологічних об'єктів в сучасних умовах частіше отримують у вигляді напівтонових зображень (рис. 1).

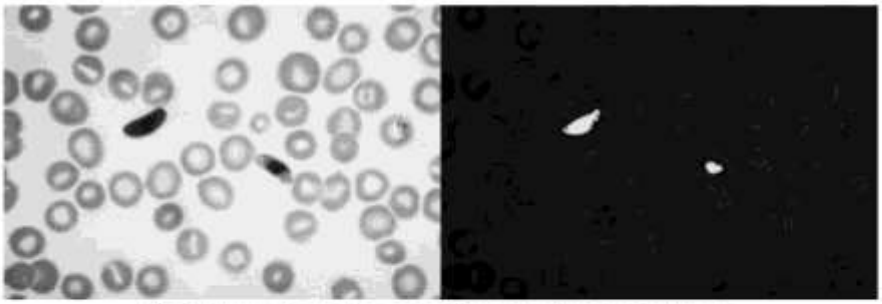


Рис.1. Приклад напівтонового мікроскопічного зображення

Розглянемо структурну модель напівтонового зображення у вигляді сукупності бінарних зображень (зрізів), що відповідають деякому рівню яскравості:

$$A(M,N)=\bigcup_{k=1}^{K} B^k(M,N) \tag{1}$$

Дана модель враховує як характеристики яскравості так і просторовий зв'язок елементів зображення та може бути реалізована не складними обчислювальними операціями.

В роботі [7] запропоноване математичне представлення ознак зв'язаності як для окремих зрізів так і для всього зображення в загалом.

Внутрішньозрізова зв'язаність Δ^k (або зв'язаність зрізу за номером k) являє собою суму зв'язаностей його елементів і визначається за наступною формулою :

$$\Delta^k=\sum_{m=1}^{M}\sum_{n=1}^{N}\delta^k(m,n) \tag{2}$$

Міжзрізова зв'язність $\Delta^{k,\ k=1}$ (тобто зв'язність між зрізами за номерами k і $k+1$) представляє суму зв'язностей елементів зрізу k з елементів зрізу k+1 визначається формулою:

$$\Delta^{k,k=1}=\sum_{m=1}^{M}\sum_{n=1}^{N}\delta^{k,k+1}(m,n) \tag{3}$$

Тоді можна представити:

$$\Delta(k)=\Delta^1,\Delta^2,\dots,\Delta^{k-1},\Delta^k,\Delta^{k+1},\dots,\Delta^K, \tag{4}$$

де $\Delta(k)$ – функція внутрізрізової зв'язності.

Функція міжзрізової зв'язаності має наступний вигляд:

$$\Delta'(k)=\Delta^{1,2},\Delta^{2,3},\dots,\Delta^{k-1,k},\Delta^{k,k+1},\Delta^{k+1,k+2},\dots,\Delta^{K-1,K}. \tag{5}$$

Це дозволить використовувати їх для наступної обробки зображень, а також і при сегментації.

Опрацьований спосіб конструювання для перебігу сегментації та отримання структурно-зв'язностного зразка зображення, може бути використаний як метод сегментації напівтонових зображень.

Даний метод можна представити такими етапами [6]:

1) квантування зображення за рівнем яскравості згідно формули (1);

2) формування функції внутрішньозрізової зв'язаності за формулами (2, 4);

3) формулювання глобального та локального максимумів функції міжзрізової зв'язаності;

4) формулювання номеру зрізу, що відповідає значенню за п.3;

5) формування функції міжзрізової зв'язаності за формулами (3, 5);

6) знаходження значень міжзрізової зв'язаності для зрізу, знайденого в п.4 та сусіднього з ним;

7) порівняння значень, отриманих за п.6 з пороговим значенням;

8) об'єднання виділених зрізів;

9) об'єднання областей за ознаками зв'язаності.

Даний метод є інваріантним до зміни орієнтації об'єкту у площині. На відміну від вже існуючих методів він дозволяє проводити визначення ініціюючих областей сегментації не випадково, а у відповідності із зв'язностними характеристиками зображення об'єкту.

Предикат однорідності для створення перебігу сегментації за зв'язаністю, є математичною моделлю даного процесу.

$$LP(S_i^*) = \begin{cases} true, if\ \Delta'(k) \geq P_1, k_{1min} \leq k \leq k_{0max}, \\ true, if\ \Delta' \geq P_2, \quad k_0 \leq k \leq k_{2min}, \\ false, if\ other\ cases. \end{cases}$$

де $\Delta'(k)$ – функція міжзрізової зв'язності зображення; P_1P_2 – порогові значення зв'язності; k_{1min},k_{2min} – "ліве" і "праве" значення номерів зрізів, для яких міжзрізова зв'ність приймає значення P_1 і P_2 відповідно (або значення локального або глобального мінімуму) відносно номеру зрізу k_{0max}; k_{0min} – номер зрізу для якого міжзрізова зв'язність приймає значення локального або глобального максимуму.

Розробляються алгоритмічні моделі перебігу сегментації за зв'язаністю для украй малих гемоцитологічних зображень, що дозволяють провести імітаційне створення перебігу сегментації за зв'язаністю на прикладних зображеннях та доводять реалістичність запропонованих моделей.

Розглянуті особливості алгоритмічної реалізації методу для покращення результатів при моделюванні процесу сегментації з застосуванням проявів зв'язаності при послідовній обробці для важких зображень із великими перерізами розподілів оптичних щільностей, дають змогу ввести поняття зв'язаності між областями[11].

Припустимо, маємо виділені області $V_1(B_i,B_{i+1},\ldots B_L)$ і $V_2(B_j,B_{j+1},\ldots B_K)$, де $B_i,B_{i+1},\ldots B_L$ – бінарні зрізи, що складають область V_1, а $B_j,B_{j+1},\ldots B_K$ – бінарні зрізи, що складають область V_2. Розглянемо задачу, яка полягає у визначенні зв'язаності між зрізами областей V_1 і V_2. Тобто зв'язаність між цими областями буде обчислюватись за формулою:

$$\Delta(V_1, V_2) = \sum_{i=1}^{L} \sum_{j=1}^{N} \Delta(B_i, B_j), \tag{6}$$

де $\Delta(B_i, B_j)$ – зв'язаність між бінарними зрізами B_i, B_j.

Приклад такого результату роботи, поєднання програмної реалізації способу моделювання сегментації та вищевказаного способу підняття підсумків сегментації надано на рис. 2, 3. Саме це дозволяє значно покращити точність сегментації.

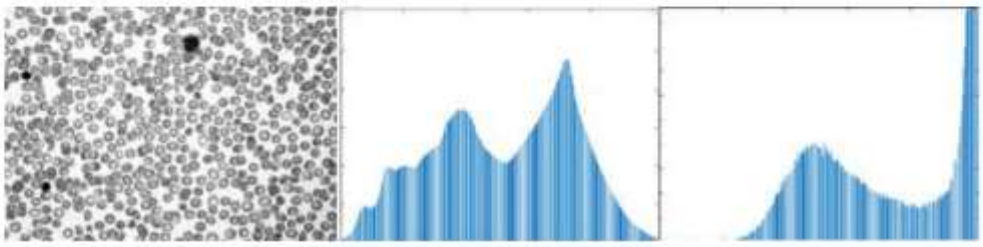


Рис. 2. Приклад вхідного зображення для сегментації: мікроскопічне зображення мазку периферичної крові та гістограми міжзрізової та внутрішньозрізової зв'язності

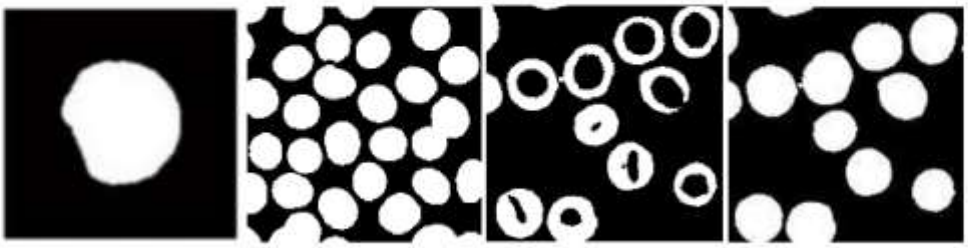


Рис. 3. Кінцевий результат сегментації: (а) – ядра лейкоциту; (б) – еритроцитів; (в) – мембрани лейкоцитів; (г) – цілого лейкоциту

Як критерій адекватності моделі процесу сегментації в роботі [10] було обрано нормовану середньоквадратичну помилку сегментації (НСКПС), вона дозволяє оцінити близькість результатів моделювання процесу сегментації деяким методом до результатів „ідеальної" сегментації. Він може бути представлений за такими формулами.

$$\text{НКПС}_i = \frac{\sum_{m=1}^{M}\sum_{n=1}^{N}\left\{b_{\lambda_i}(m,n) - \widetilde{b_{\lambda_i}}(m,n)\right\}^2}{\tilde{S}_{\lambda_i}},$$

де НКПС_i – нормована квадратична помилка сегментації і-го сегменту; $\tilde{S}_{\lambda_i}$ – площа сегменту з міткою λ_i на ідеальному зображенні, елементи - елементи матриці ідеального зображення сегменту з міткою λ, а $\tilde{b}_{\lambda_i}(m, n)$ – елементи матриці реального зображення сегменту з міткою λ_i. Саме тому необхідно усереднити НКПС за кількістю сегментів у вихідній сукупності:

$$\text{НСКПС} = \frac{\sum_{i=1}^{L}\text{НКПС}'}{L}$$

де L - кількість зображень у виборці.

Результати даного моделювання наведені в табл. 1 та на рис. 4.

Таблиця 1

Результати експериментальних досліджень методів сегментації

№п/п	*Назва методу*	*НСКПС*
1	Метод нарощування областей в автоматичному режимі	13,8%
2	Метод нарощування областей за участю оператора	5,2%
3	Метод порогової сегментації	7,1%
4	Метод зв'язностної сегментації	6,9%
5	Метод зв'язностної сегментації + метод покращення результатів	4,1%

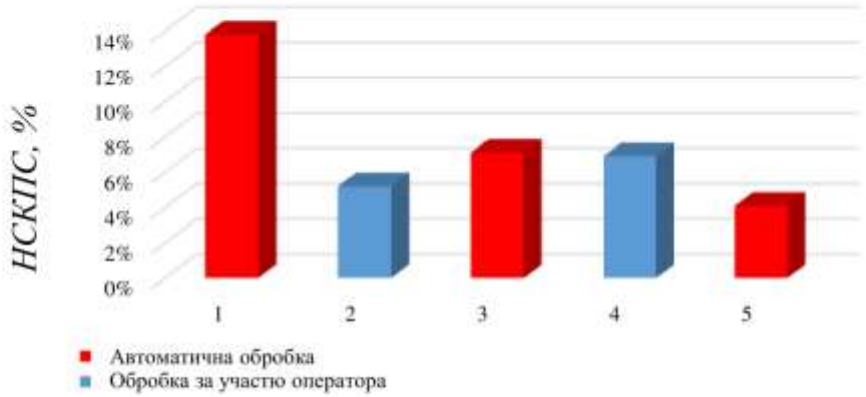


Рис. 4. Порівняльна діаграма результатів експерименту

Отже, можна зробити підсумок, що запропонована модель є адекватною. Також було встановлено, що для гемоцитологічних зображень з нечіткими розмитими границями зв'язностний метод дає кращі результати за точністю (в середньому – на 3%)[10].

Розроблені моделі зв'язностні можуть бути представлені як на різних зображеннях одного типу, наприклад, на гемоцитологічних, так і на зображеннях інших типів. Для плямових зображень лазерних пучків сегментація зображення проводиться шляхом побудови структурної зв'язностної моделі зображення та відповідних до неї гістограм міжзрізової та внутрішньозрізової зв'язаностей. В якості ініціюючого зрізу обирався зріз з максимальною яскравістю. А поріг цього зрізу обчислювався на рівні мінімальних значень зв'язностей вліво та вправо від ініціюючого зрізу.

До задачі сегментації медичних зображень, можуть бути віднесені питання текстурної сегментації зображень псоріатичного ураження шкіри. Хвороба псоріазу – це порушення, що характеризується плямами рожевого кольору, з чіткими краями, які покриті лусочками сріблясто-білого кольору. Алгоритм класифікації дерматологічних зображень на основі методу опорних векторів, який включає такі етапи [3]:

1. Завантаження RGB (Red, Green, Blue) зображення зображень псоріатичного ураження шкіри.
2. Формування вектора ознак, з використанням статистичних моментів: начального моменту 1-го порядку і центральний момент другого порядку [3].
3. Формування навчальної вибірки – кожному елементу вектора ознак ставиться у відповідність метка, яка визначає до якого кластеру відносяться образи об'єктів розпізнавання.
4. Навчання класифікатору, на основі навчальної вибірки з використанням лінійної функції ядра.
5. Завантаження зображення псоріазного захворювання.
6. Формування тестової множини для завантаженого зображення.
7. Застосування навченого класифікатору для класифікації тестової множини.
8. Обчислення точності класифікації.

Висновки

Розглянуто спосіб сегментації напівтонових зображень з одержанням зв'язностної конфігурованої моделі зображення.

Було розглянуто математичне представлення предикату сумісності для сегментації зображень на підставі властивостей зв'язаності, що створює умови для автоматизації процесу, що у поєднанні із методом знаходження точок зрізів з їх найбільшим розумінням внутрішньозрізової зв'язаності як початкових, та знаходження граничних значень як мінімальних значень функцій міжзрізової зв'язаності, що за потребою дає змогу оператору брати участь

у процесі на всіх його етапах.

Досліджено алгоритм та імітаційний макет процесу сегментації за зв'язаністю на мікроскопічних гемоцитологічних зображеннях популярних методів сегментації, застосування способу опорних векторів для класифікації образів об'єктів розпізнавання в задачі сегментації медичних зображень.

Література

1. Вапник, В.Н. Теория распознавания образов (статистические проблемы обучения). – М.: Главная редакция физико-математической литературы изд-ва «Наука», 1974. – 416 с.
2. Гонсалес Р. Цифровая обработка зображений. – М. : Техносфера, 2005. – 1072с.
3. Кучеренко Г. О., Горпенко Д. Р., Волкова Н.П. Сегментація медичних зображень за допомогою кольорових ознак // Восьма Міжнародна наукова конференція студентів та молодих вчених «Сучасні інформаційні технології», ОНПУ, 23-25 травня, 2018, С.101–102.
4. Мартинюк Т.Б., Скорюкова Я.Г., Хом'юк В.В. Особливості використання позрізової обробки для сегментації багатоградаційних зображень // Вісник Вінницького політехнічного інституту – 2004.- №4. – С.82-88.
5. Патент 55790 А України, МПК 7 G06 G7/14, Підсумовувальний пороговий пристрій / Мартинюк Т.Б., Скорюкова Я.Г., Барський С.Б., Баранов Р.К. - №2002065; Заявл.20.06.2002; Опубл.15.04.2003. Бюл.№4. 2003 МПІ – 3 с.
6. Патент № 2024939С1 РФ, МКИ G 06 K 9/00, Метод и устройство выделения изображения / Тимченко Л.И., Кутаев Ю.Ф., Марков С.М., Скорюкова Я.Г. - №5036557; Заявл. 08.07.91, опубл.15.12.92., бюл.№ 23. – 8с.
7. Рыжов А. П. Элементы теории нечетких множеств и ее приложений. – М.: Изд-во МГУ, 2003.
8. Тимченко Л.И., Скорюкова Я.Г., Марков С.М. Анализ и синтез алгоритмов распознавания объектов в масштабе реального времени / УкрНИИТИ. – Киев, 1991 - 46 с. – Рус. - Депонированные научные работы, 1991, №1195 - Ук91, №12(242), б/о 498 от 16.08.91
9. Тимченко Л.І., Скорюкова Я.Г. Метод покращення результатів сегментації гемоцитологічних зображень // Оптико-електронні інформаційно-енергетичні технології. – 2003.- № 1-2 (5-6). – С. 46-49
10. Тимченко Л.І., Скорюкова Я.Г., Марков С.М., Гальченко Я.О. Сегментація багатоградаційних зображень на основі ознак просторової зв'язаності // Вісник Вінницького політехнічного інституту. – 1998.- №4. – С.39- 44.
11. Харалик Р. М. Статистический и структурный подходы к описанию текстур / Р. М. Харалик // ТИИЭР. – 1979. – Т. 67, № 5. – С. 98 — 120.

Рецензія/Peer review:

Надрукована/Printed:

Рецензент: д.т.н., проф.. ……

Wed Dec 02 13:46:05 EET 2020, Петровський Сергій Степанович, Хмельницький національний університет, ХНУ

Anti-Plagiarism v-15.257

Максимальне співпадіння з одним документом 1.0%

Словники перевірки: en_US, ru_RU, ua_UA. **Помилок в документах: 11%**

	Документ		Сумарний збіг по Базі Даних	
ID: 82067 Назва: Інформаційна технологія маркування та перевірки автентичності медичних зображень Додано в БД: 2020-12-02 Автора: Мостовий Влад Вікторович Керівники: Пасічник О.А. Консультанти: Опоненти:	Символи	Лексеми	Символи	Лексеми
	64726	541	918 (1%)	12 (2%)

Джерело плагіату

ID	Опис	Наявність плагіату в документі	
		Символи	Лексеми

РІШЕННЯ КАФЕДРИ **КОМП'ЮТЕРНИХ НАУК ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ** ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованого системою виявлення текстових збігів/ідентичності/схожості:

Назва: **Інформаційна технологія маркування та перевірки автентичності медичних зображень**
Автор: **Мостовий В. В.**
Спеціальність: **122 Комп'ютерні науки**
Науковий керівник: **к.т.н., доцент Пасічник О.А.**

Після аналізу звіту подібності зроблено такий висновок:

№	Висновок	Позначка про відповідність
1	Запозичення, виявлені в роботі, є законними і не є плагіатом (далі – зазначаються підстави віднесення запозичень до правомірних). Робота приймається до захисту.	**відповідає**
2	Виявлені запозичення не є плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи (далі – зазначаються детальні та аргументовані підстави віднесення запозичень до правомірних). Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої та електронної версії роботи	-
3	Робота містить навмисні текстові спотворення, передбачувані спроби укриття запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	-
4	Інше:	-

Підтвердження: Виявленні запозичення не є плагіатом т.я. розміщенні в розділах, які не описують безпосередньо авторське дослідження, складають 3,79% та мають посилання на приведений список літературних джерел

02.12.2020
Дата

Підпис керівника

Підпис завідувача кафедри

Хмельницький національний університет

Кафедра комп'ютерних наук та інформаційних технологій

ВІДГУК ОПОНЕНТА

на дипломну роботу магістра

Магістра *гр. КНм-19-1 Мостового Влада Вікторовича*

На тему: *Інформаційна технологія маркування та перевірки автентичності медичних зображень*

1. Актуальність і значення теми

Наразі багато все більше медичних закладів використовують різні системи для зберігання та передачі медичних зображень. Більшість відповідних систем є застарілими та не є зосередженими на безпеці даних що передаються між користувачами системи. Тому тема відповідної роботи є досить актуального адже пропонується методика для рішення проблематики захисту даних впроваджуючи високий рівень надійності та безпеки.

2.Оцінка якості та достовірності проведених досліджень.

Отримані результати показують себе на відповідному рівні із існуючими результатами, наведеними в наукових роботах і виданнях.

3. Оцінка запропонованих заходів та пропозицій, практичної цінності та ефективності.

Проведені дослідження та запропонована інформаційна технологія представляє науково-прикладну цінність, та є ефективним дослідженням в галузі медичних інформаційних систем, а саме роботи з медичними зображеннями. Отримані результати можна використати з метою покращення існуючих програмних продуктів та систем.

4. Загальний висновок та оцінка

Робота виконана в повному обсязі. Досліджені та проаналізовані дані за допомогою комплексу входять в рамки допустимих відхилень. Пояснювальна записка оформлена в відповідності з нормами. За своєю структурою, практичними цінностями, поставленій меті та вирішеними задачами робота відповідає вимогам, що пред'являються до освітньо-кваліфікаційного рівня «магістр», а її автор Мостовий В.В. заслуговує присвоєння кваліфікації магістра з комп'ютерних наук та інформаційних технологій.

Робота заслуговує на оцінку «*відмінно*».

Опонент *Дихо О.В., д.т.н., проф., зав.каф. ТАМ ХНУ*