

Міністерство освіти і науки України

Луцький національний технічний університет

(повне найменування закладу вищої освіти)

Факультет комп'ютерних та інформаційних технологій

(повне найменування факультету)

Кафедра комп'ютерної інженерії та безпеки

(повне найменування кафедри)

**КВАЛІФІКАЦІЙНА РОБОТА
ЗА СТУПЕНЕМ ВИЩОЇ ОСВІТИ «БАКАЛАВР»**

**ЦИФРОВА ОБРОБКА ЗОБРАЖЕНЬ МЕТОДАМИ MATLAB
ДЛЯ ЇХ ПЕРЕДАЧІ ПО МЕРЕЖІ**

**DIGITAL IMAGE PROCESSING USING MATLAB FOR
NETWORK TRANSMISSION**

спеціальність 123 Комп'ютерна інженерія

(шифр і назва спеціальності)

освітня програма Комп'ютерна інженерія

(назва освітньої програми)

Виконав: здобувач вищої освіти
групи KI-41
Потерайко Назар Анатолійович

(підпис)

Керівник:
к.т.н., доцент
Пех Петро Антонович

(підпис)

Кваліфікаційну роботу
допущено до захисту
« » червня 2026 р.
Гарант освітньої програми:
к.т.н., доцент
Лавренчук Світлана Василівна

(підпис)

Луцьк – 2026 року

ЛУЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

Факультет комп'ютерних та інформаційних технологій

Кафедра комп'ютерної інженерії та безпеки

Ступінь вищої освіти: бакалавр

Галузь знань: 12 Інформаційні технології

Спеціальність: 123 Комп'ютерна інженерія

Освітня програма: «Комп'ютерна інженерія»

ЗАТВЕРДЖУЮ

Завідувач кафедри

доц. Т. Терлецький

« 23 » 12 2025 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧУ ВИЩОЇ ОСВІТИ

Потерайко Назар Анатолійович

(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи Цифрова обробка зображень засобами MATLAB для їх передачі по мережі

Керівник роботи к.т.н., доцент Пех Петро Антонович

затверджені наказом закладу вищої освіти від «20» грудня 2025 року № 536/01-02

2. Строк подання здобувачем вищої освіти кваліфікаційної роботи 28.05.2026 р.

3. Вихідні дані до роботи Джерелом розробки є науково-технічна література та публікації в періодичних виданнях з даного питання, опубліковані зарубіжні та вітчизняні роботи в даній області, різні інтернет-ресурси технічного спрямування

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити):

Вступ

Огляд існуючих рішень

Методи та інструменти вирішення питань дослідження

Розроблення апаратної та програмної частини проекту

Висновки

5. Перелік графічного (ілюстративного) матеріалу:

Функціональна схема системи обробки зображень

Інтерфейс програмної системи у середовищі MATLAB

Результати обробки та JPEG-стиснення зображень

Графіки залежності метрик PSNR, SSIM та MSE від рівня якості JPEG

Результати передачі зображень у хмарне середовище Google Drive

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис	
		завдання видав	завдання прийняв
Огляд існуючих рішень	Пех П.А., доцент		
Методи та інструменти вирішення питань дослідження	Пех П.А., доцент		
Розроблення апаратної та програмної частини проекту	Пех П.А., доцент		
Нормоконтроль	Багнюк Н. В., доцент		
Гарант ОП	Лавренчук С. В., доцент		
Показник запозичень тексту		%	
Академічна доброчесність	Міскевич О. І., ст. викладач		

7. Дата видачі завдання 23.12.2025 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1.	Огляд літератури із досліджуваної проблеми, аналіз предметної області та наявних рішень	до 10.02.2026 р.	
2.	Огляд існуючих рішень	до 02.03.2026 р.	
3.	Методи та інструменти вирішення питань дослідження	до 02.04.2026 р.	
4.	Розроблення апаратної та програмної частини проекту	до 10.04.2026 р.	
5.	Представлення остаточного варіанту кваліфікаційної роботи керівникові	до 01.05.2026 р.	
6.	Нормоконтроль	до 23.05.2026 р.	
7.	Інструментальна перевірка на академічний плагіат	до 22.05.2026 р.	
8.	Здача кваліфікаційної роботи та всіх супровідних документів на кафедрі	до 28.05.2026 р.	

Здобувач вищої освіти

(підпис)

Назар ПОТЕРАЙКО

(прізвище, ініціали)

Керівник кваліфікаційної роботи

(підпис)

Петро ПЕХ

(прізвище, ініціали)

АНОТАЦІЯ

Потерайко Н. А. Цифрова обробка зображень методами MATLAB для їх передачі по мережі. Рукопис.

Кваліфікаційна робота бакалавра ОП «Комп'ютерна інженерія» спеціальності 123 Комп'ютерна інженерія. Луцький національний технічний університет Луцьк, 2026.

Кваліфікаційна робота складається зі вступу, трьох розділів, висновків, списку використаних джерел та додатків.

Перший розділ присвячено огляду існуючих методів цифрової обробки зображень. У розділі обґрунтовано актуальність теми, розглянуто основи цифрової обробки зображень, методи попередньої обробки, JPEG-стиснення та передачі графічної інформації комп'ютерними мережами. Також проведено аналіз сучасних програмних засобів для реалізації систем цифрової обробки зображень.

У другому розділі здійснено вибір та обґрунтування методів і програмних інструментів для реалізації системи. Розглянуто алгоритми цифрової обробки, методи JPEG-стиснення, моделі передачі даних у середовищі MATLAB та можливості використання Image Processing Toolbox для роботи із графічними даними.

Третій розділ присвячено розробленню програмної системи обробки, стиснення та передачі зображень до хмарного середовища Google Drive. Реалізовано графічний інтерфейс користувача, функції попередньої обробки, JPEG-стиснення, передачі файлів та оцінювання якості зображень за допомогою метрик PSNR, SSIM та MSE. Проведено експериментальне дослідження впливу параметрів JPEG-стиснення на якість зображення та розмір файлу.

Ключові слова: цифрова обробка зображень, MATLAB, JPEG, PSNR, SSIM, MSE, фільтрація, хмарне середовище, Google Drive, передача зображень.

ANNOTATION

Poteraiko N. Digital Image Processing Using MATLAB Methods for Network Transmission. Manuscript.

Qualifying work of a bachelor of EP «Computer Engineering» specialty 123 Computer Engineering. Lutsk National Technical University. Lutsk, 2026.

The qualification work consists of an introduction, three chapters, conclusions, a list of references, and appendices.

The first chapter is devoted to the review of existing digital image processing methods. The chapter substantiates the relevance of the topic, considers the basics of digital image processing, methods of preliminary image enhancement, JPEG compression, and transmission of graphical information through computer networks. In addition, an analysis of modern software tools for implementing digital image processing systems is carried out.

The second chapter presents the selection and justification of methods and software tools for system implementation. Algorithms of digital image processing, JPEG compression methods, data transmission models in the MATLAB environment, and the capabilities of the Image Processing Toolbox for working with graphical data are considered.

The third chapter is devoted to the development of a software system for image processing, compression, and transmission to the Google Drive cloud environment. A graphical user interface, functions for preliminary image processing, JPEG compression, file transmission, and image quality evaluation using PSNR, SSIM, and MSE metrics were implemented. An experimental study of the influence of JPEG compression parameters on image quality and file size was also conducted.

Keywords: digital image processing, MATLAB, JPEG, PSNR, SSIM, MSE, filtering, cloud environment, Google Drive, image transmission.

ЗМІСТ

ВСТУП.....	8
РОЗДІЛ 1 ОГЛЯД ІСНУЮЧИХ РІШЕНЬ	11
1.1 Основи цифрової обробки зображень	11
1.2 Методи попередньої обробки зображень	12
1.3 Методи стиснення зображень	13
1.4 Передача зображень по комп'ютерній мережі.....	16
1.5 Існуючі програмні рішення та системи	17
РОЗДІЛ 2 МЕТОДИ ТА ІНСТРУМЕНТИ ВИРІШЕННЯ ПИТАНЬ ДОСЛІДЖЕННЯ	19
2.1 Постановка задачі	19
2.2 Алгоритми цифрової обробки	20
2.3 Методи стиснення для передачі	24
2.4 Моделі передачі даних у MATLAB	27
2.5 Інструменти MATLAB	29
РОЗДІЛ 3 РОЗРОБЛЕННЯ АПАРАТНОЇ ТА ПРОГРАМНОЇ ЧАСТИНИ ПРОЕКТУ.....	32
3.1 Загальна архітектура та принцип роботи системи	32
3.2 Завантаження та первинний аналіз зображення	33
3.3 Попередня обробка зображення	34
3.4 Стиснення зображення та його вплив на якість.....	38
3.5 Передача зображення у хмарне середовище Google Drive.....	39
3.6 Оцінювання якості зображення	41
3.7 Комплексний результат роботи системи	43
3.8 Результати експерименту	46
3.9 Порівняння методів фільтрації	50
3.10 Результати експерименту	51

ВИСНОВКИ.....	54
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	56

ВСТУП

Актуальність теми. Сучасний розвиток інформаційних технологій супроводжується стрімким зростанням обсягів графічної інформації, що передається комп'ютерними мережами та зберігається у хмарних середовищах. Цифрові зображення широко використовуються у мультимедійних системах, веб-технологіях, системах дистанційного навчання, медичних інформаційних системах, соціальних мережах та сервісах хмарного зберігання даних. У зв'язку з цим особливої актуальності набувають задачі ефективної обробки, стиснення та передачі графічної інформації.

Передача зображень через мережу супроводжується значними витратами ресурсів, зокрема пропускної здатності каналу зв'язку, обсягу пам'яті та часу передачі. Для зменшення навантаження на мережу використовуються алгоритми цифрової обробки та JPEG-стиснення, які дозволяють значно скоротити розмір файлів при збереженні прийнятної якості зображення.

Важливу роль у сучасних системах передачі даних відіграють хмарні технології, що забезпечують можливість віддаленого зберігання, синхронізації та доступу до файлів із різних пристроїв. Одним із найбільш популярних хмарних сервісів є Google Drive, який дозволяє автоматизувати процес передачі та зберігання графічної інформації.

Актуальність теми також обумовлена необхідністю оцінювання ефективності методів стиснення та передачі зображень. Для цього використовуються спеціальні метрики якості, зокрема PSNR, SSIM та MSE, які дозволяють кількісно оцінити ступінь втрат після обробки та стиснення.

Для реалізації систем цифрової обробки зображень доцільним є використання середовища MATLAB, яке містить потужні інструменти для математичного моделювання, аналізу сигналів, роботи із графічними даними та створення інтерактивних програмних додатків.

Таким чином, розробка програмної системи цифрової обробки, JPEG-стиснення та передачі зображень до хмарного середовища Google Drive є актуальним науково-практичним завданням.

Об'єкт дослідження. Об'єктом дослідження є процес цифрової обробки, стиснення та передачі зображень у комп'ютерних системах.

Предмет дослідження. Предметом дослідження є методи цифрової обробки зображень, алгоритми JPEG-стиснення, оцінювання якості графічних даних та механізми передачі файлів до хмарного середовища Google Drive.

Мета дослідження. Метою дослідження є аналіз та реалізація методів цифрової обробки, стиснення і передачі зображень із використанням середовища MATLAB та хмарних технологій зберігання даних.

Мета дипломної роботи. Метою дипломної роботи є розробка програмної системи обробки, JPEG-стиснення та передачі зображень до хмарного середовища Google Drive із можливістю оцінювання якості отриманих результатів за допомогою метрик PSNR, SSIM та MSE.

Для досягнення поставленої мети необхідно вирішити такі задачі:

- проаналізувати існуючі методи цифрової обробки зображень;
- дослідити алгоритми попередньої обробки та JPEG-стиснення;
- реалізувати систему передачі зображень до хмарного середовища Google Drive;
- розробити програмний додаток у середовищі MATLAB;
- реалізувати розрахунок метрик якості PSNR, SSIM та MSE;
- провести експериментальне дослідження ефективності роботи системи;
- оцінити вплив параметрів JPEG-стиснення на якість зображення та розмір файлу.

Методи дослідження. Методи дослідження базуються на використанні методів цифрової обробки зображень, алгоритмів JPEG-стиснення, математичного

аналізу, методів оцінювання якості графічних даних та програмної реалізації у середовищі MATLAB.

Практичне значення отриманих результатів. Практичне значення отриманих результатів полягає у створенні програмної системи, яка дозволяє виконувати обробку, стиснення та передачу зображень до хмарного середовища Google Drive із подальшим оцінюванням якості отриманих результатів. Розроблена система може бути використана у навчальних, дослідницьких та мультимедійних застосуваннях, пов'язаних із передачею графічної інформації комп'ютерними мережами.

Структура дипломної роботи. Дипломна робота складається зі вступу, трьох розділів, висновків та списку використаних джерел. У першому розділі проведено аналіз сучасних методів цифрової обробки, стиснення та передачі зображень. У другому розділі розглянуто методи та інструменти реалізації програмної системи у середовищі MATLAB. У третьому розділі представлено програмну реалізацію системи, результати експериментальних досліджень, аналіз метрик якості та процес передачі зображень до хмарного середовища Google Drive.

РОЗДІЛ 1

ОГЛЯД ІСНУЮЧИХ РІШЕНЬ

1.1 Основи цифрової обробки зображень

Цифрова обробка зображень є важливою складовою сучасних інформаційних технологій, оскільки забезпечує отримання, перетворення, аналіз і передачу візуальних даних у формі, придатній для комп'ютерного опрацювання. Цифрове зображення подається як двовимірний масив пікселів, де кожному елементу відповідає значення яскравості або набору кольорових компонент.

У сучасних дослідженнях підкреслюється, що повний цикл обробки включає введення, попереднє покращення, сегментацію, аналіз ознак та передачу результатів [14].

Типовий процес обробки зображень включає такі етапи:

- введення даних;
- попередня обробка;
- сегментація;
- аналіз;
- кодування та передача.

Кожен з цих етапів має суттєвий вплив на кінцевий результат. Зокрема, недостатньо якісна попередня обробка може призвести до накопичення помилок на наступних етапах.

Для представлення кольорових зображень використовуються різні колірні простори. Найбільш поширеним є RGB, проте для обробки часто використовуються HSV або YCbCr.

Це пояснюється тим, що такі простори дозволяють розділити яскравість та кольорову складову, що підвищує ефективність алгоритмів обробки. У середовищі MATLAB зображення інтерпретується як матриця чисел, що дозволяє виконувати над ним різноманітні математичні операції. Використання спеціалізованих

інструментів, таких як Image Processing Toolbox, значно спрощує реалізацію складних алгоритмів.

Додатково слід зазначити, що важливу роль відіграє дискретизація та квантування сигналу. Саме ці процеси визначають точність представлення зображення у цифровій формі. При недостатній розрядності можливе виникнення спотворень, що негативно впливають на якість обробки.

Важливою характеристикою цифрових зображень також є просторова роздільна здатність, яка визначає кількість пікселів, що припадають на одиницю площі зображення. Зі збільшенням роздільної здатності підвищується деталізація об'єктів, однак одночасно зростає обсяг даних, що потребує більшої обчислювальної потужності та ресурсів пам'яті при обробці й передачі інформації.

Крім того, суттєве значення має глибина кольору, яка визначає кількість бітів, що використовуються для представлення кожного пікселя. Збільшення глибини кольору дозволяє отримати більш плавні переходи між відтінками та покращити якість зображення. Однак це також призводить до збільшення обсягу інформації, що передається та зберігається.

У сучасних системах цифрової обробки вибір параметрів дискретизації та квантування здійснюється з урахуванням необхідного балансу між якістю відображення та ефективністю використання ресурсів комп'ютерної системи [10].

1.2 Методи попередньої обробки зображень

Попередня обробка є одним із найважливіших етапів цифрової обробки зображень, оскільки саме вона визначає якість подальших перетворень.

Основною метою цього етапу є усунення шумів, покращення контрасту та підготовка зображення до стискання або аналізу.

Для усунення шумів використовуються різні методи:

- середнє фільтрування;

- медіанне фільтрування;
- гаусове фільтрування.

Медіанна фільтрація є ефективною для усунення імпульсних шумів, оскільки дозволяє зберегти контури об'єктів. Гаусове згладжування, у свою чергу, використовується для пригнічення високочастотних перешкод [3].

Окрім фільтрації, важливим є покращення контрасту. Для цього застосовуються методи гістограмного вирівнювання.

Ще одним важливим аспектом є масштабування. Зменшення розміру зображення дозволяє значно скоротити обсяг даних, що передаються мережею. Проте цей процес потребує обережності, оскільки надмірне зменшення може призвести до втрати деталей.

Також до методів попередньої обробки відносять геометричні перетворення, такі як поворот, масштабування та зсув, які дозволяють адаптувати зображення до конкретних умов використання [2].

1.3 Методи стиснення зображень

Стиснення зображень є одним із ключових етапів цифрової обробки, особливо у випадках, коли необхідно передавати великі обсяги даних через мережу з обмеженою пропускною здатністю. Основною метою стиснення є зменшення обсягу даних при збереженні максимально можливої якості зображення.

Існують два основні типи стиснення:

- без втрат;
- з втратами.

Безвтратні методи стиснення дозволяють повністю відновити початкове зображення. До таких форматів належить PNG, який широко використовується для зображень з чіткими контурами, текстом або графікою. Основною перевагою цього підходу є відсутність втрат якості, однак ступінь стиснення є обмеженим.

Стиснення з втратами, навпаки, передбачає часткове видалення інформації, яка є менш значущою для людського сприйняття. Найбільш поширеним форматом такого типу є JPEG. Його популярність зумовлена здатністю забезпечувати високий ступінь стиснення при відносно невеликих втратах якості.

Алгоритм JPEG базується на дискретному косинусному перетворенні (DCT), яке дозволяє представити зображення у вигляді набору частотних компонент [4].

Далі виконується квантування, під час якого відбувається зменшення точності представлення даних. Після квантування застосовується ентропійне кодування, яке дозволяє додатково зменшити обсяг даних.

Окрім класичного формату JPEG, у сучасних системах використовуються й інші методи стиснення графічної інформації. Формат PNG застосовується переважно для зображень із чіткими контурами та текстовими елементами, оскільки забезпечує стиснення без втрат. Формат WebP поєднує переваги методів стиснення з втратами та без втрат, забезпечуючи менший розмір файлу при збереженні високої якості зображення.

Формат JPEG2000 використовує вейвлет-перетворення замість дискретного косинусного перетворення, що дозволяє покращити якість стиснення при високих коефіцієнтах стискання. Проте через більшу складність реалізації та меншу підтримку в програмному забезпеченні він використовується значно рідше.

Для задач, пов'язаних із передачею зображень мережею, формат JPEG залишається одним із найбільш ефективних рішень завдяки підтримці більшістю програмних платформ та можливості керування співвідношенням між якістю та розміром файлу.

Таким чином, JPEG забезпечує ефективне поєднання різних методів стиснення. На рисунку 1.1 показані основні етапи стиснення зображення у форматі JPEG.

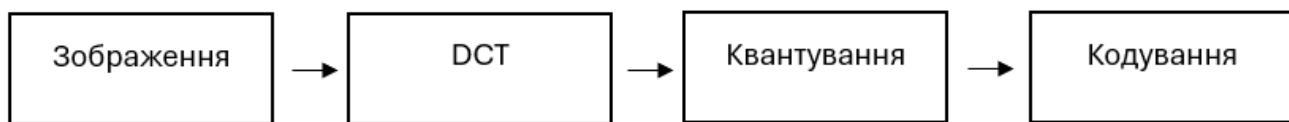


Рисунок 1.1 – Основні етапи стиснення зображення у форматі JPEG

Як показано на рисунку 1.1, процес JPEG-стиснення включає дискретне косинусне перетворення, квантування та кодування, що дозволяє значно зменшити розмір зображення.

Сучасні формати, такі як JPEG2000 та WebP, пропонують покращені алгоритми стиснення. Вони використовують вейвлет-перетворення або більш ефективні методи кодування, що дозволяє досягти кращих результатів у порівнянні з класичним JPEG [15].

Для оцінки ефективності стиснення використовуються спеціальні метрики, такі як PSNR, MSE та SSIM. Вони дозволяють кількісно оцінити ступінь втрати якості.

Таким чином, вибір методу стиснення є критично важливим етапом, що визначає ефективність всієї системи передачі зображень.

У таблиці 1.1 представлено порівняння найбільш поширених форматів зображень за основними характеристиками, зокрема можливістю стиснення, наявністю втрат якості та особливостями використання у системах цифрової обробки та передачі зображень.

Таблиця 1.1 – Порівняння форматів зображень

Формат	Стиснення	Втрати	Особливості
JPEG	Так	Так	Фото
PNG	Так	Ні	Без втрат
BMP	Ні	Ні	Великий розмір

Як видно з таблиці 1.1, різні формати зображень мають власні переваги та недоліки залежно від сфери використання.

Формат JPEG забезпечує значне зменшення розміру файлів за рахунок використання стиснення з втратами, що робить його ефективним для фотографій та передачі графічної інформації мережею. Формат PNG дозволяє зберігати якість без втрат, однак потребує більшого обсягу пам'яті. Формат BMP характеризується відсутністю стиснення, що призводить до значного розміру файлів.

1.4 Передача зображень по комп'ютерній мережі

Передача зображень через комп'ютерні мережі є складною задачею, що включає не лише передачу даних, але й забезпечення їх цілісності, швидкості доставки та відновлення на приймальній стороні. Ефективність передачі значною мірою залежить від обраного протоколу та способу кодування даних.

Основними протоколами передачі даних є TCP та UDP. Протокол TCP (Transmission Control Protocol) забезпечує надійну передачу даних шляхом встановлення з'єднання між відправником і отримувачем. Він гарантує доставку пакетів у правильному порядку та забезпечує контроль помилок.

Протокол UDP (User Datagram Protocol), у свою чергу, є менш надійним, але більш швидким, оскільки не потребує встановлення з'єднання та підтвердження отримання даних. Це робить його доцільним для застосувань, де важлива швидкість, наприклад у потоковому відео. Для передачі зображень у більшості випадків використовується TCP, оскільки він гарантує цілісність даних. Втрата навіть одного байта може призвести до пошкодження всього зображення.

У середовищі MATLAB реалізація передачі даних може здійснюватися за допомогою функцій `tcpclient` та `tcpserver`. Вони дозволяють організувати клієнт-серверну взаємодію між двома вузлами мережі.

Процес передачі включає кілька етапів:

- кодування зображення у байтовий потік;
- передача даних через мережу;

- прийом даних;
- декодування та відновлення зображення.

Додатково можуть використовуватися функції `webread` та `webwrite`, які дозволяють здійснювати передачу через HTTP-протокол.

Важливим аспектом є оцінка швидкості передачі та затримок. Ці параметри впливають на ефективність системи та визначають її придатність для практичного використання.

На ефективність передачі графічної інформації впливають характеристики мережевого середовища. До основних параметрів належать пропускна здатність каналу, затримки передачі, втрати пакетів та стабільність мережевого з'єднання. При передачі великих зображень або значної кількості файлів низька швидкість каналу може призвести до суттєвого збільшення часу передачі.

Додаткову роль відіграє використання хмарних технологій, які дозволяють здійснювати централізоване зберігання та синхронізацію даних. Використання хмарних сервісів забезпечує можливість доступу до інформації незалежно від місця перебування користувача та використовуваного пристрою.

Таким чином, передача зображень є складним процесом, що потребує узгодження методів стиснення, протоколів передачі та алгоритмів відновлення [4].

1.5 Існуючі програмні рішення та системи

Сучасні задачі цифрової обробки зображень можуть вирішуватися за допомогою різних програмних засобів. Найбільш поширеними є MATLAB, OpenCV, Python та інші спеціалізовані платформи.

MATLAB відзначається високим рівнем інтеграції, що дозволяє виконувати всі етапи обробки зображень у межах одного середовища. Наявність Image Processing Toolbox значно спрощує реалізацію складних алгоритмів [9].

OpenCV є популярною бібліотекою з відкритим кодом, яка широко використовується у задачах комп'ютерного зору. Вона підтримує велику кількість мов програмування, зокрема C++ та Python. Основною перевагою OpenCV є висока продуктивність та гнучкість.

Python також є популярним середовищем для обробки зображень завдяки таким бібліотекам, як Pillow, scikit-image та NumPy.

Він дозволяє швидко реалізовувати алгоритми, однак часто потребує використання декількох бібліотек одночасно.

У порівнянні з іншими середовищами MATLAB має такі переваги:

- зручний інтерфейс;
- наявність готових функцій;
- інтеграція з математичними інструментами;
- можливість швидкого прототипування.

Недоліками є комерційна ліцензія та обмежена гнучкість у порівнянні з відкритими платформами.

У межах даної дипломної роботи вибір MATLAB є обґрунтованим, оскільки він дозволяє об'єднати обробку, стиснення та передачу зображень у межах одного середовища.

РОЗДІЛ 2

МЕТОДИ ТА ІНСТРУМЕНТИ ВИРІШЕННЯ ПИТАНЬ ДОСЛІДЖЕННЯ

2.1 Постановка задачі

Метою дипломної роботи є розроблення програмної системи цифрової обробки зображень засобами MATLAB для їх підготовки та подальшої передачі комп'ютерною мережею. Основним завданням є забезпечення оптимального співвідношення між якістю зображення та обсягом передаваних даних.

Це означає, що система повинна не тільки зменшувати розмір даних, але й зберігати їх інформативність.

Для досягнення поставленої мети необхідно вирішити такі задачі:

- обґрунтувати вибір методів попередньої обробки;
- реалізувати алгоритми стиснення зображень;
- організувати передачу даних мережею;
- забезпечити відновлення зображення на приймальній стороні;
- оцінити якість отриманих результатів.

Для реалізації систем цифрової обробки зображень можуть використовуватися різні підходи до побудови архітектури програмного забезпечення. Найбільш поширеними є локальна обробка даних, клієнт-серверна архітектура та хмарно-орієнтовані системи.

Локальна модель характеризується високою швидкістю виконання операцій та незалежністю від мережевого з'єднання, проте має обмеження щодо віддаленого доступу до результатів. Клієнт-серверний підхід дозволяє забезпечити централізовану обробку та передачу даних, однак потребує додаткових ресурсів для підтримки серверної частини.

Хмарні технології забезпечують можливість зберігання та доступу до інформації з різних пристроїв, однак ефективність їх використання залежить від швидкості мережевого з'єднання.

У таблиці 2.1 наведено порівняльний аналіз основних підходів до організації систем цифрової обробки зображень.

Таблиця 2.1 – Порівняння архітектур програмних систем

Архітектура	Переваги	Недоліки
Локальна	висока швидкість роботи	відсутність віддаленого доступу
Клієнт-серверна	централізована обробка	потребує сервера
Хмарна	доступ з різних пристроїв	залежність від мережі

Як видно з таблиці 2.1, кожен із підходів має власні переваги та недоліки. Для реалізації системи цифрової обробки та передачі зображень було обрано клієнт-серверну архітектуру, оскільки вона забезпечує централізовану обробку даних та дозволяє організувати надійну передачу інформації між вузлами системи. Для програмної реалізації обрано середовище MATLAB, яке надає вбудовані засоби для обробки зображень, передачі даних та візуалізації результатів. Такий підхід забезпечує зручність проведення експериментальних досліджень та спрощує реалізацію алгоритмів.

2.2 Алгоритми цифрової обробки

У межах дослідження використовуються базові алгоритми цифрової обробки зображень, які забезпечують підготовку даних до передачі. До них належать фільтрація, покращення контрасту та масштабування.

Першим етапом є усунення шумів. Наявність шуму може суттєво знизити ефективність стиснення та ускладнити подальший аналіз. Для цього застосовуються різні методи фільтрації.

Кожен із методів фільтрації має власні особливості застосування та демонструє різну ефективність залежно від типу шуму, структури зображення та вимог до якості кінцевого результату.

Середній фільтр характеризується простотою реалізації та високою швидкістю обробки, однак його використання може призводити до втрати дрібних деталей через ефект розмиття.

Медіанний фільтр забезпечує ефективне усунення імпульсних шумів і дозволяє зберігати контури об'єктів, проте має більшу обчислювальну складність.

Гаусовий фільтр дозволяє отримати компроміс між рівнем пригнічення шуму та збереженням структури зображення. Завдяки використанню вагових коефіцієнтів центральні пікселі мають більший вплив на результат, що дозволяє уникнути надмірного розмиття зображення.

У таблиці 2.2 представлено порівняння основних методів фільтрації зображень за типом шуму, перевагами та недоліками їх використання в системах цифрової обробки зображень.

Таблиця 2.2 – Порівняльний аналіз методів фільтрації

Метод фільтрації	Тип шуму	Переваги	Недоліки	Доцільність застосування
Average	Слабкий випадковий	Проста реалізація, висока швидкість	Розмиття деталей зображення	Низька
Median	Імпульсний	Збереження контурів об'єктів	Більший час обробки	Середня
Gaussian	Універсальний	Баланс між шумозаглушенням і деталізацією	Незначне розмиття	Висока

Як видно з таблиці 2.2, різні методи фільтрації характеризуються різною ефективністю залежно від типу шуму та особливостей вхідного зображення. Середній фільтр є найпростішим у реалізації, однак має суттєвий недолік у вигляді втрати деталізації. Медіанний фільтр демонструє кращі результати при усуненні імпульсних шумів, проте потребує більшого часу обробки. Гаусовий фільтр забезпечує найбільш збалансовані характеристики та дозволяє ефективно зменшувати шум при мінімальному погіршенні якості зображення.

На основі проведеного аналізу для реалізації розроблюваної системи було обрано гаусове згладжування, оскільки воно забезпечує оптимальне співвідношення між якістю отриманого результату та обчислювальною складністю алгоритму.

Наступним етапом є покращення контрасту. Для цього використовуються методи нормалізації та гістограмного вирівнювання. Вони дозволяють підвищити деталізацію та зробити зображення більш придатним для подальшої обробки [2].

Масштабування зображення також є важливим етапом. Зменшення роздільної здатності дозволяє значно скоротити обсяг даних, що передаються, проте потребує вибору оптимального коефіцієнта масштабування.

Окрім методів фільтрації, важливе значення мають алгоритми покращення якості зображення. Найбільш поширеними підходами є нормалізація яскравості, гістограмне вирівнювання та адаптивне покращення контрасту. Нормалізація дозволяє виконати перетворення діапазону яскравості зображення до заданих меж. Гістограмне вирівнювання забезпечує більш рівномірний розподіл яскравості, що дозволяє підвищити деталізацію окремих ділянок зображення. Адаптивні методи покращення контрасту забезпечують локальне підсилення деталей та демонструють вищу ефективність у випадку нерівномірного освітлення.

У таблиці 2.3 наведено порівняльний аналіз основних методів покращення зображень за їх перевагами, недоліками та доцільністю використання у системах цифрової обробки графічної інформації.

Таблиця 2.3 – Порівняння методів покращення зображення

Метод	Переваги	Недоліки	Доцільність застосування
Нормалізація	Простота реалізації	Незначне покращення контрасту	Середня
Гістограмне вирівнювання	Підвищення деталізації та контрасту	Може підсилювати шуми	Висока
Адаптивне покращення	Локальне підсилення деталей	Вища обчислювальна складність	Висока

Як видно з таблиці 2.3, розглянуті методи покращення зображення відрізняються за принципом роботи, обчислювальною складністю та впливом на якість кінцевого результату. Нормалізація характеризується простотою реалізації, однак забезпечує обмежене покращення характеристик зображення.

Гістограмне вирівнювання дозволяє підвищити деталізацію та покращити контрастність, хоча в окремих випадках може призводити до підсилення шумів. Адаптивні методи демонструють високу ефективність при роботі із зображеннями з нерівномірним освітленням, проте характеризуються більшою обчислювальною складністю.

На основі проведеного порівняльного аналізу методів покращення якості зображень для реалізації розроблюваної системи доцільно використовувати гістограмне вирівнювання. На відміну від простого методу нормалізації, який забезпечує лише загальне коригування діапазону яскравості, гістограмне вирівнювання дозволяє підвищити деталізацію зображення та покращити контрастність. Адаптивні методи демонструють високу ефективність, проте

потребують більших обчислювальних ресурсів. Тому використання гістограмного вирівнювання забезпечує оптимальне співвідношення між якістю результатів та складністю реалізації в середовищі MATLAB.

2.3 Методи стиснення для передачі

Стиснення зображень є ключовим етапом у системах передачі даних, оскільки саме воно визначає обсяг інформації, яка передається мережею. Основною метою стиснення є зменшення розміру файлу при збереженні максимально можливої якості зображення, достатньої для подальшого використання. У межах даної роботи доцільним є використання алгоритму JPEG, який належить до методів стиснення з втратами.

Його популярність зумовлена високою ефективністю та широкою підтримкою в програмних середовищах. Алгоритм JPEG базується на дискретному косинусному перетворенні (DCT), яке дозволяє представити зображення у вигляді набору частотних компонент. Це дає змогу виділити найбільш значущі елементи зображення та зменшити вплив менш важливих деталей.

Дискретне косинусне перетворення виконує перехід від просторового представлення зображення до частотного. У результаті цього енергія зображення концентрується у невеликій кількості низькочастотних коефіцієнтів, тоді як високочастотні компоненти можуть бути частково видалені без значного погіршення візуального сприйняття.

Після виконання перетворення здійснюється процес квантування, під час якого частина коефіцієнтів округлюється або зменшується. Саме цей етап визначає основну втрату інформації та безпосередньо впливає на ступінь стиснення.

Процес стиснення включає такі етапи:

- розбиття зображення на блоки фіксованого розміру;
- застосування DCT до кожного блоку;

- квантування коефіцієнтів;
- ентропійне кодування.

Квантування є найбільш важливим етапом, оскільки саме на ньому відбувається основна втрата інформації. Зменшення точності представлення коефіцієнтів дозволяє досягти значного зменшення обсягу даних [13].

У середовищі MATLAB реалізація JPEG-стиснення може здійснюватися за допомогою функції `imwrite` із заданим параметром якості. Це дозволяє експериментально досліджувати вплив рівня стиснення на якість зображення [6].

Для оцінки ефективності стиснення використовуються метрики якості:

- MSE (середньоквадратична помилка);
- PSNR (пікове співвідношення сигнал/шум);
- SSIM (структурна подібність).

Метрики PSNR, SSIM та MSE дозволяють кількісно оцінити вплив стиснення на якість зображення [16].

Для вибору найбільш доцільного методу стиснення було проведено порівняльний аналіз сучасних форматів зображень, які застосовуються у системах передачі графічних даних. Основними критеріями порівняння є тип стиснення, рівень втрати якості, ступінь зменшення розміру файлу та підтримка сучасними програмними платформами. У таблиці 2.4 наведено порівняльний аналіз форматів стиснення зображень за основними характеристиками та особливостями використання.

Таблиця 2.4 – Порівняльний аналіз форматів стиснення зображень

Формат	Тип стиснення	Втрати якості	Ступінь стиснення	Підтримка платформ
JPEG	З втратами	Так	Високий	Висока
PNG	Без втрат	Ні	Середній	Висока

Продовження таблиці 2.4

Формат	Тип стиснення	Втрати якості	Ступінь стиснення	Підтримка платформ
WebP	Комбінований	Так/Ні	Високий	Середня
JPEG2000	З втратами	Так	Високий	Низька

Як видно з таблиці 2.4, розглянуті формати мають різні характеристики щодо ступеня стиснення та якості отриманого результату. Формат PNG забезпечує відсутність втрат інформації, однак характеризується відносно невеликим коефіцієнтом стиснення. Формат WebP дозволяє отримати менший розмір файлу при високій якості зображення, проте підтримується не всіма програмними системами. JPEG2000 забезпечує покращені характеристики стиснення, однак має складнішу реалізацію та обмежене використання у сучасному програмному забезпеченні.

На основі проведеного порівняльного аналізу форматів стиснення для реалізації розроблюваної системи було обрано формат JPEG. Порівняно з PNG, який забезпечує збереження даних без втрат, JPEG дозволяє значно зменшити розмір файлу при незначному погіршенні якості зображення. Формат WebP також забезпечує високий ступінь стиснення, однак його підтримка окремими програмними засобами може бути обмеженою. JPEG2000 має покращені характеристики стиснення, проте характеризується більш складною реалізацією та менш широкою підтримкою.

Проведений аналіз показав, що використання сучасних методів стиснення дозволяє досягти ефективного балансу між якістю отриманого зображення та обсягом переданих даних. Вибір відповідного формату й алгоритму обробки залежить від вимог конкретної системи, зокрема швидкості передачі, рівня втрат інформації та доступних програмних засобів. Важливим аспектом також є

можливість реалізації обраних алгоритмів у програмному середовищі для подальшого моделювання та експериментальних досліджень.

2.4 Моделі передачі даних у MATLAB

Передача зображень через комп'ютерні мережі є важливою складовою розроблюваної системи. Ефективність передачі залежить від обраної моделі взаємодії, протоколів зв'язку та способів представлення даних.

У межах даної роботи використовується клієнт-серверна модель передачі даних. У цій моделі один вузол (клієнт) формує та надсилає дані, а інший (сервер) приймає їх, обробляє та відображає результат.

Для реалізації передачі використовується протокол TCP, який забезпечує:

- встановлення з'єднання між вузлами;
- контроль цілісності даних;
- впорядковану передачу інформації.

Це особливо важливо при передачі зображень, оскільки навіть незначні втрати даних можуть призвести до спотворення результату [14].

У середовищі MATLAB передача даних реалізується за допомогою функцій `tcpclient` та `tcpserver`. Вони дозволяють створювати з'єднання між двома програмними модулями та здійснювати обмін даними у вигляді байтових потоків.

Процес передачі включає такі етапи:

- перетворення зображення у двійковий формат;
- передача даних через мережу;
- прийом даних на стороні сервера;
- відновлення зображення.

Для представлення зображення у вигляді байтового потоку можуть використовуватися функції кодування, що дозволяють передавати графічні дані у зручному форматі [11].

Додатково можливе використання HTTP-протоколу за допомогою функцій webread та webwrite, що дозволяє реалізувати передачу даних через веб-сервіси.

Важливим аспектом є оцінка часу передачі, затримок та пропускну здатності. Ці параметри визначають ефективність системи та її придатність до практичного використання.

Для передачі даних у комп'ютерних мережах можуть використовуватись різні транспортні протоколи. Найбільш поширеними є TCP та UDP, які відрізняються механізмом передачі, швидкістю роботи та рівнем надійності. Вибір протоколу залежить від вимог до швидкості передачі та цілісності інформації.

У таблиці 2.5 наведено порівняльний аналіз основних транспортних протоколів передачі даних. Крім основних характеристик, важливими параметрами оцінювання протоколів є затримка передачі, стійкість до втрати пакетів та ефективність використання мережевих ресурсів.

Таблиця 2.5 – Порівняльний аналіз транспортних протоколів передачі даних

Протокол	Переваги	Недоліки	Доцільність використання
TCP	Контроль цілісності, надійна передача	Більші затримки	Висока
UDP	Висока швидкість передачі	Можлива втрата пакетів	Середня

Як видно з таблиці 2.5, протокол TCP забезпечує більш надійну передачу інформації завдяки механізмам контролю цілісності та повторної передачі пакетів у випадку втрат. Протокол UDP характеризується меншою затримкою та вищою швидкістю передачі, однак не гарантує доставку всіх пакетів даних.

На основі проведеного аналізу для реалізації розроблюваної системи доцільно використовувати протокол TCP, оскільки навіть незначна втрата частини графічних даних може призвести до спотворення отриманого результату.

Таким чином, за результатами проведеного порівняльного аналізу для реалізації розроблюваної системи було обрано клієнт-серверну модель передачі даних на основі протоколу TCP. Обраний підхід забезпечує високу надійність передачі графічної інформації, контроль цілісності даних та мінімізацію ризику втрати важливих елементів зображення.

2.5 Інструменти MATLAB

Для реалізації системи цифрової обробки та передачі зображень використовується середовище MATLAB, яке забезпечує широкий спектр інструментів для роботи з графічними даними.

Основним інструментом є Image Processing Toolbox, який містить функції для:

- зчитування та запису зображень;
- фільтрації;
- покращення контрасту;
- геометричних перетворень;
- аналізу якості зображень.

Зокрема, функції `imread` та `imwrite` використовуються для роботи з файлами, `imfilter` - для фільтрації, `imadjust` та `histeq` - для покращення контрасту [5].

Для реалізації алгоритмів стиснення можуть використовуватися стандартні засоби MATLAB або власні алгоритми, що дозволяє проводити експериментальні дослідження [1].

Для передачі даних застосовуються функції:

- `tcpclient`;
- `tcpserver`;

– webread;

–webwrite.

Вони забезпечують можливість створення клієнт-серверної архітектури та передачі даних у мережі.

Окрім цього, MATLAB надає засоби візуалізації результатів, що дозволяє порівнювати зображення до та після обробки.

Окремою перевагою MATLAB є можливість створення інтерактивних програмних інтерфейсів за допомогою App Designer. Даний інструмент дозволяє реалізувати зручне графічне середовище для керування алгоритмами обробки, налаштування параметрів та відображення результатів у режимі реального часу.

Для оцінювання доцільності використання MATLAB у роботах із цифрової обробки зображень варто враховувати особливості інших програмних середовищ.

Наприклад, Python із бібліотеками OpenCV та NumPy характеризується відкритим програмним кодом і широкими можливостями для реалізації алгоритмів машинного навчання. Середовище MATLAB, у свою чергу, має вбудовані інструменти для математичного моделювання, аналізу та візуалізації результатів, що значно спрощує процес розробки та експериментального дослідження алгоритмів.

У таблиці 2.6 наведено порівняльну характеристику програмних середовищ для обробки зображень.

Таблиця 2.6 – Порівняння програмних середовищ для обробки зображень

Середовище	Переваги	Недоліки
MATLAB	Вбудовані інструменти, зручна візуалізація	Платне програмне забезпечення
Python + OpenCV	Безкоштовність, гнучкість	Складніше налаштування
ImageJ	Простота роботи	Менша функціональність

Як видно з таблиці 2.6, середовище MATLAB має перевагу завдяки наявності вбудованих інструментів для математичного моделювання, аналізу та візуалізації результатів. Python з бібліотекою OpenCV забезпечує високу гнучкість і є безкоштовним рішенням, однак потребує додаткового налаштування середовища. ImageJ характеризується простотою використання, проте має обмежений набір функцій для реалізації складних алгоритмів обробки зображень.

Таким чином, за результатами проведеного порівняльного аналізу програмних середовищ для реалізації розроблюваної системи було обрано середовище MATLAB.

Даний вибір обумовлений наявністю вбудованих інструментів для цифрової обробки зображень, засобів моделювання, візуалізації результатів та підтримкою швидкої реалізації алгоритмів без необхідності додаткового налаштування середовища [9].

РОЗДІЛ 3. РОЗРОБЛЕННЯ АПАРАТНОЇ ТА ПРОГРАМНОЇ ЧАСТИНИ ПРОЕКТУ

3.1 Загальна архітектура та принцип роботи системи

У межах виконання кваліфікаційної роботи було розроблено програмну систему для обробки, стиснення та передачі зображень із використанням середовища MATLAB. Основною метою розробки є створення інструменту, який дозволяє реалізувати повний цикл обробки зображення та провести аналіз впливу окремих етапів на якість кінцевого результату.

Система реалізована у вигляді інтерактивного додатку з графічним інтерфейсом користувача, створеного за допомогою App Designer. Такий підхід дозволяє поєднати обчислювальні можливості MATLAB із наочним відображенням результатів, що є особливо важливим у задачах дослідження.

Архітектура системи побудована за послідовним принципом та включає такі основні етапи:

- введення зображення;
- попередня обробка;
- стиснення;
- передача;
- оцінка якості.

Кожен етап виконує окрему функцію, проте їх взаємодія формує єдиний технологічний процес. Такий підхід дозволяє не лише реалізувати функціонал системи, але й дослідити вплив кожного етапу на якість зображення.

Розроблений інтерфейс дозволяє керувати параметрами обробки та спостерігати результати в реальному часі, що значно спрощує проведення експериментів. Загальний вигляд розробленого програмного додатка наведено на рисунку 3.1. Інтерфейс містить основні елементи керування, область для відображення зображень, поле вибору якості JPEG та блок виведення метрик якості.

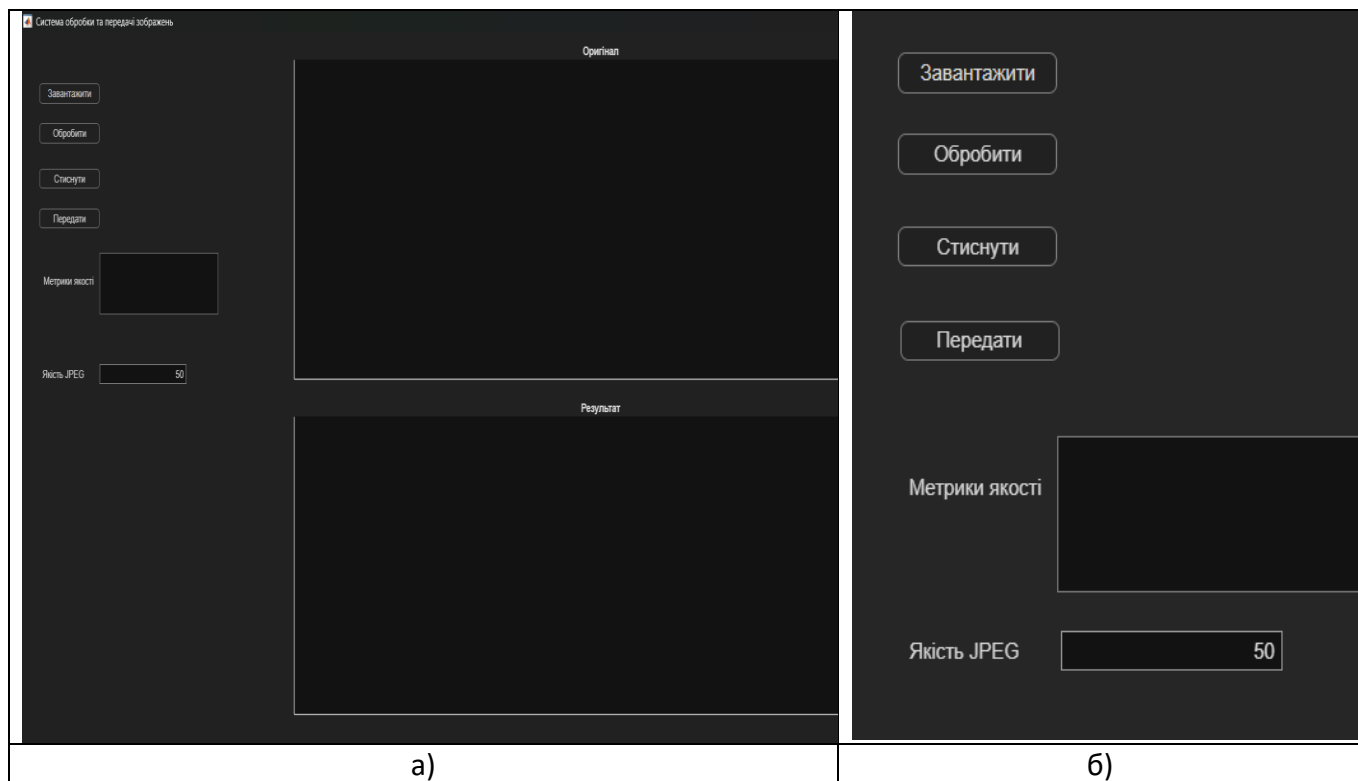


Рисунок 3.1 – Головне вікно програмного додатка

а) загальний вигляд головного вікна

б) інтерфейс головного вікна

Як видно з рисунка 3.1, інтерфейс побудований таким чином, щоб користувач міг послідовно виконати всі етапи роботи системи: завантаження, обробку, стиснення, передачу та аналіз результатів.

3.2 Завантаження та первинний аналіз зображення

Першим етапом роботи системи є завантаження зображення користувачем. Після вибору файлу зображення відображається у графічному інтерфейсі, що дозволяє здійснити його первинний аналіз.

На даному етапі оцінюються такі характеристики:

- наявність шумів;
- рівень контрасту;

- деталізація;
- загальна якість зображення.

Первинний аналіз є важливим, оскільки дозволяє визначити необхідність подальшої обробки та обрати відповідні методи. Результат завантаження початкового зображення у систему представлено на рисунку 3.2.

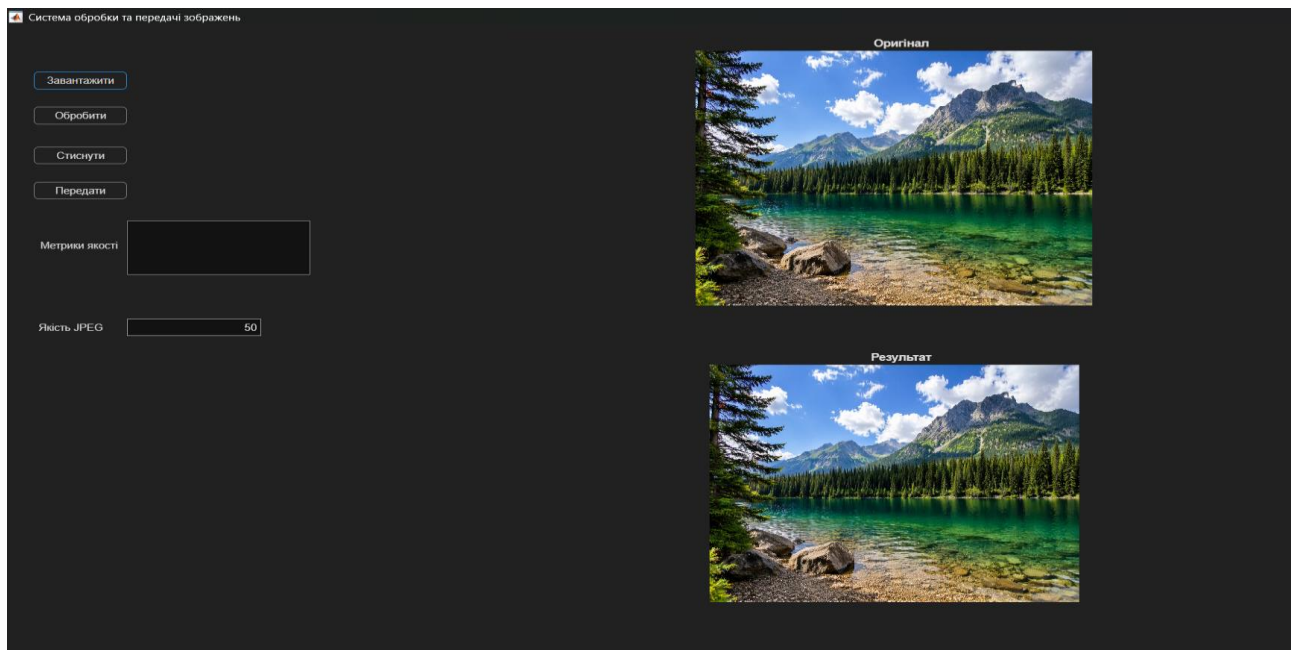


Рисунок 3.2 – Завантаження зображення в систему

На рисунку 3.2 показано процес завантаження зображення в програмну систему. Після вибору користувачем графічного файлу здійснюється його зчитування та відображення у вікні програми для подальшої обробки. На даному етапі зображення стає доступним для виконання операцій попередньої обробки, налаштування параметрів стиснення та подальшої передачі.

3.3 Попередня обробка зображення

Попередня обробка зображення є важливим етапом підготовки даних, оскільки якість початкового зображення безпосередньо впливає на ефективність

подальших алгоритмів стиснення та передачі. Наявність шумів, низького контрасту або надлишкової деталізації може призводити до збільшення обсягу переданих даних та погіршення кінцевого результату. Саме тому використання засобів попередньої обробки дозволяє підвищити ефективність функціонування всієї системи. У межах роботи реалізовано наступні етапи обробки:

- перетворення у відтінки сірого;
- фільтрація шумів;
- підвищення контрасту;
- масштабування.

Перетворення зображення у відтінки сірого дозволяє суттєво спростити подальшу обробку даних, оскільки замість трьох кольорних каналів (червоного, зеленого та синього) використовується лише один канал яскравості. Це зменшує кількість інформації, яку необхідно аналізувати та обробляти, що знижує обчислювальне навантаження на систему.

Використання відтінків сірого дозволяє зосередити обробку саме на цих важливих елементах. Фільтрація шумів реалізована за допомогою гаусового згладжування. Обраний метод дозволяє ефективно зменшити випадкові шуми без суттєвого спотворення структури зображення. Це особливо важливо перед стисненням, оскільки наявність шумів негативно впливає на ефективність алгоритмів стиснення.

Покращення контрасту здійснюється за допомогою вирівнювання гистограми, що дозволяє підвищити інформативність зображення та зробити деталі більш виразними.

Масштабування зображення дозволяє зменшити роздільну здатність і, відповідно, обсяг даних, що передаються.

Для реалізації попередньої обробки зображення у середовищі MATLAB було використано фрагмент програмного коду, який виконує перетворення зображення у

відтінки сірого, фільтрацію шумів, покращення контрасту та масштабування. Код реалізації наведено нижче (лістинг 3.1).

Лістинг 3.1 – Код реалізації попередньої обробки зображення

```
% Перетворення у відтінки сірого
gray = rgb2gray(app.Image);
% Гаусова фільтрація шумів
filtered = imgaussfilt(gray, 2);
% Покращення контрасту
enhanced = histeq(filtered);
% Масштабування зображення
resized = imresize(enhanced, 0.5);
% Збереження результату
app.Processed = resized;
% Відображення результатів
imshow(app.Image, 'Parent', app.UIAxes);
imshow(resized, 'Parent', app.UIAxes2);
title(app.UIAxes, 'Оригінал');
title(app.UIAxes2, 'Після обробки');
```

кінець лістингу 3.1

Наведений фрагмент програмного коду реалізує послідовне виконання основних етапів попередньої обробки зображення. На першому етапі здійснюється перетворення кольорового зображення у відтінки сірого, що дозволяє зменшити кількість оброблюваних даних та спростити подальші обчислення.

Далі виконується згладжування за допомогою гаусового фільтра, яке забезпечує зменшення рівня випадкових шумів без суттєвої втрати важливих деталей зображення. Після цього застосовується гістограмне вирівнювання для підвищення контрастності та покращення видимості окремих елементів зображення.

Завершальним етапом є масштабування, що дозволяє зменшити розмір графічних даних для подальшої передачі мережею та знизити обчислювальні витрати при подальшій обробці.

Результат виконання попередньої обробки зображення наведено на рисунку 3.3.

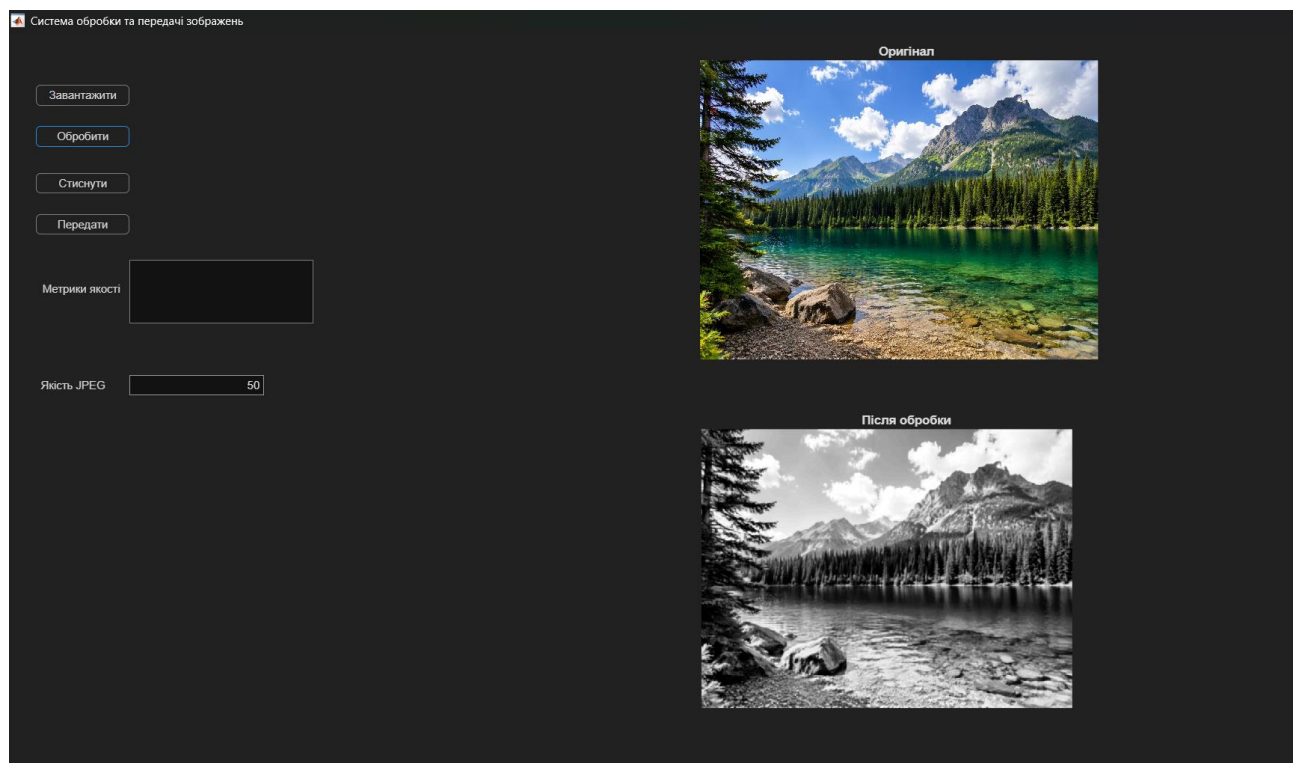


Рисунок 3.3 – Результат попередньої обробки зображення

Аналіз результатів показує, що після обробки зменшується рівень шуму, покращується контраст та підвищується чіткість зображення. Це створює оптимальні умови для подальшого стиснення. Додатково варто зазначити, що попередня обробка зображення сприяє підвищенню ефективності подальших етапів роботи системи. Таким чином, попередня обробка є важливим етапом, який дозволяє не лише покращити візуальні характеристики зображення, а й оптимізувати подальші процеси аналізу, стиснення та передачі графічної інформації.

3.4 Стиснення зображення та його вплив на якість

Стиснення зображення виконується з метою зменшення обсягу даних перед передачею. У роботі використано алгоритм JPEG, який є стандартом для обробки графічної інформації.

Алгоритм JPEG базується на дискретному косинусному перетворенні, що дозволяє виділити основні частоти зображення. Менш значущі компоненти можуть бути зменшені або відкинуті, що і забезпечує ефект стиснення.

Важливим параметром є рівень якості. Зміна цього параметра дозволяє керувати співвідношенням між якістю та розміром файлу.

Для реалізації стиснення зображення у середовищі MATLAB використано фрагмент програмного коду, який виконує JPEG-стиснення із заданим параметром якості. Код реалізації наведено нижче (лістинг 3.2).

Лістинг 3.2 – Код реалізації стиснення зображення

```
q = round(app.QualityEditField.Value);  
imwrite(app.Processed, ...  
    'app_compressed.jpg', ...  
    'Quality', q);  
info = dir('app_compressed.jpg');  
fileSizeKB = info.bytes / 1024;  
img = imread('app_compressed.jpg');  
app.Compressed = img;  
global CompressedData  
CompressedData = img;  
imshow(app.Compressed, ...  
    'Parent', app.UIAxes2);  
title(app.UIAxes, 'До стиснення');  
title(app.UIAxes2, 'Після стиснення');
```

кінець лістингу 3.2

Результат стиснення зображення із заданим параметром якості JPEG наведено на рисунку 3.4.

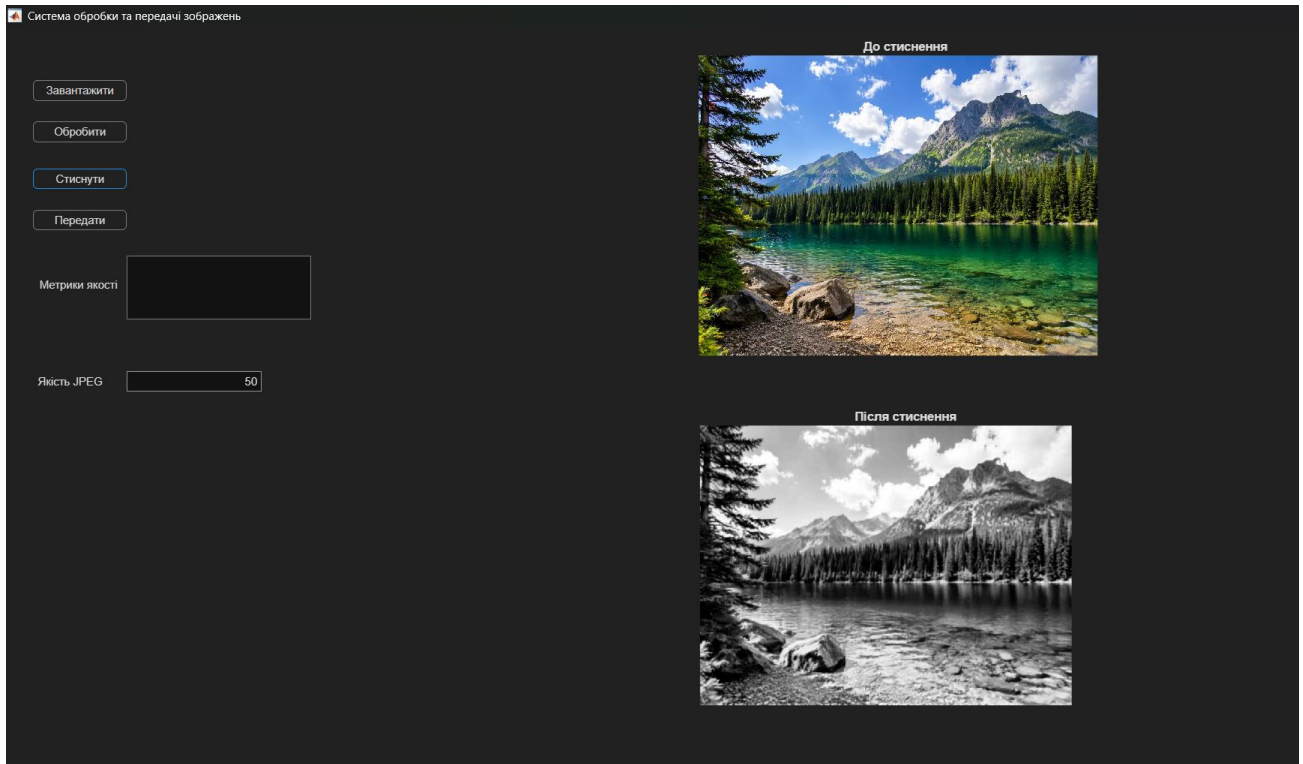


Рисунок 3.4 – Результат стиснення зображення

З рисунка 3.4 видно, що після стиснення зображення зберігає основну структуру, однак його розмір зменшується, що є важливим для подальшої передачі.

3.5 Передача зображення у хмарне середовище Google Drive

Передача зображення є важливим етапом розробленої системи, оскільки саме на цьому етапі забезпечується обмін даними між локальним програмним модулем та віддаленим хмарним сховищем.

У межах даної роботи реалізовано передачу стисненого зображення до сервісу Google Drive засобами середовища MATLAB.

Перед початком передачі система виконує попередню обробку та JPEG-стиснення зображення із заданим параметром якості. Після цього сформований файл автоматично копіюється до локальної папки, синхронізованої з Google Drive. Завдяки цьому забезпечується автоматичне завантаження зображення до хмарного середовища.

Для реалізації передачі використовується функція `copyfile`, яка виконує копіювання сформованого JPEG-файлу до директорії `MATLAB_CLOUD`. Дана папка синхронізується із сервісом Google Drive, що дозволяє автоматично передавати файл у хмарне сховище.

Після завершення передачі система відображає повідомлення про успішне завантаження файлу до Google Drive, а також виводить основні характеристики зображення та параметри передачі. Результат виконання передачі наведено на рисунку 3.5.

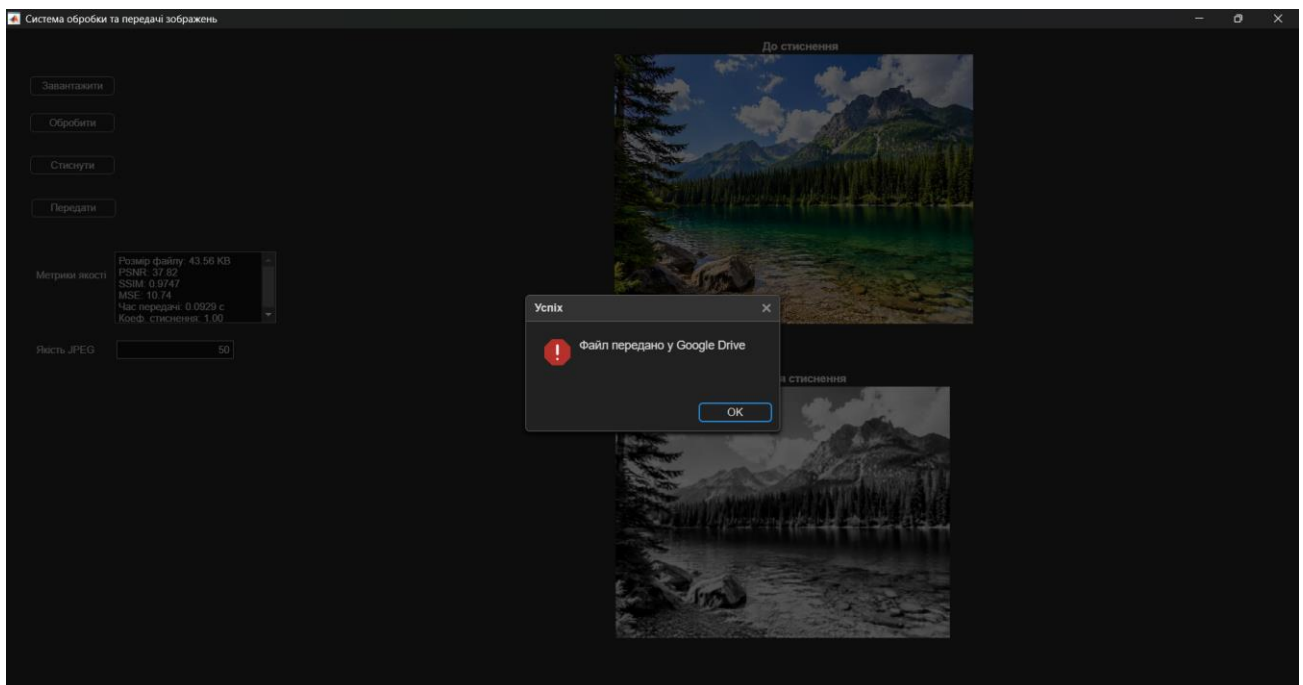


Рисунок 3.5 – Повідомлення про успішну передачу зображення до Google Drive

Після копіювання файлу до синхронізованої директорії зображення автоматично з'являється у хмарному середовищі Google Drive та стає доступним для подальшого перегляду й використання на інших пристроях.

На рисунку 3.6 показано результат успішної синхронізації зображення у хмарному сховищі Google Drive.



Рисунок 3.6 – Відображення переданого зображення у Google Drive

Отримані результати підтверджують працездатність реалізованої системи передачі даних та можливість використання хмарних технологій для організації зберігання й передачі графічної інформації. Реалізований підхід дозволяє наблизити систему до реальних умов функціонування сучасних мережевих сервісів та підвищує практичну цінність розробленого програмного забезпечення.

3.6 Оцінка якості зображення

Після виконання стиснення та передачі зображення система автоматично виконує оцінювання якості отриманого результату. Для цього у програмному середовищі MATLAB реалізовано розрахунок основних метрик якості зображення,

що дозволяють оцінити ступінь втрат після JPEG-стиснення та ефективність передачі файлу до хмарного середовища.

У процесі роботи система визначає такі показники:

- розмір стисненого файлу;
- коефіцієнт стиснення;
- час передачі файлу;
- значення PSNR;
- значення SSIM;
- значення MSE.

Розмір файлу дозволяє оцінити ефективність стиснення та обсяг даних, необхідний для передачі через мережу. Коефіцієнт стиснення показує співвідношення між початковим та стисненим зображенням, що дає змогу визначити ступінь зменшення обсягу даних.

Час передачі є важливою характеристикою продуктивності системи, оскільки демонструє швидкість копіювання файлу до хмарного сховища Google Drive. Визначення цього параметра здійснюється за допомогою вбудованих функцій MATLAB tic та toc.

Для оцінювання якості відновленого зображення використовуються класичні метрики цифрової обробки зображень. Метрика PSNR (Peak Signal-to-Noise Ratio) характеризує рівень спотворень між початковим та стисненим зображенням. Чим більше значення PSNR, тим вищою є якість отриманого результату.

Метрика SSIM (Structural Similarity Index) використовується для оцінки структурної подібності між двома зображеннями. Значення SSIM наближається до одиниці у випадку високої схожості між оригіналом та стисненим зображенням.

Метрика MSE (Mean Squared Error) визначає середньоквадратичну похибку між пікселями початкового та обробленого зображень. Менше значення MSE свідчить про менший рівень втрат інформації після стиснення.

Після завершення всіх етапів система автоматично виводить розраховані параметри у спеціальному інформаційному полі інтерфейсу користувача. Приклад відображення метрик якості наведено на рисунку 3.7.

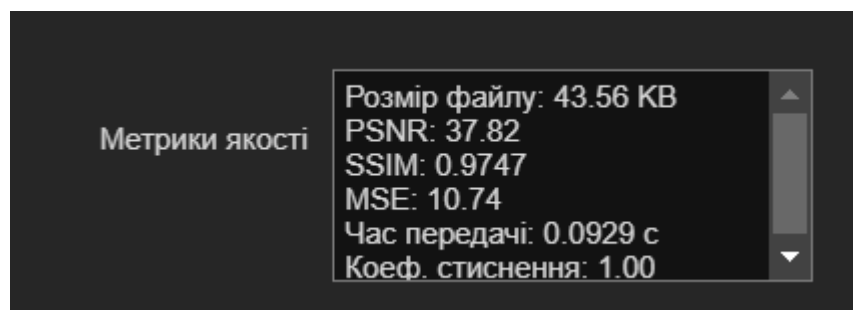


Рисунок 3.7 – Відображення метрик якості та параметрів передачі зображення

Отримані результати демонструють, що реалізована система забезпечує ефективне стиснення зображень із збереженням прийнятної якості та можливістю швидкої передачі до хмарного середовища. Використання комплексного набору метрик дозволяє об'єктивно оцінити якість роботи розробленого програмного забезпечення та ефективність застосованих алгоритмів обробки даних.

3.7 Комплексний результат роботи системи

Після виконання всіх етапів роботи програмна система формує комплексний результат, який дозволяє провести інтегральний аналіз ефективності реалізованих алгоритмів цифрової обробки зображень, JPEG-стиснення та передачі даних до хмарного середовища Google Drive.

Загальний вигляд програмної системи після завершення повного циклу обробки наведено на рисунку 3.8.

У межах одного програмного інтерфейсу користувач отримує можливість одночасно переглядати результати локальної обробки зображення, параметри стиснення, числові значення метрик якості та результат передачі файлу до хмарного

сховища. Це дозволяє комплексно оцінювати ефективність роботи системи, аналізувати вплив методів цифрової обробки на якість зображення та контролювати процес передачі графічної інформації у віддалене середовище зберігання даних.

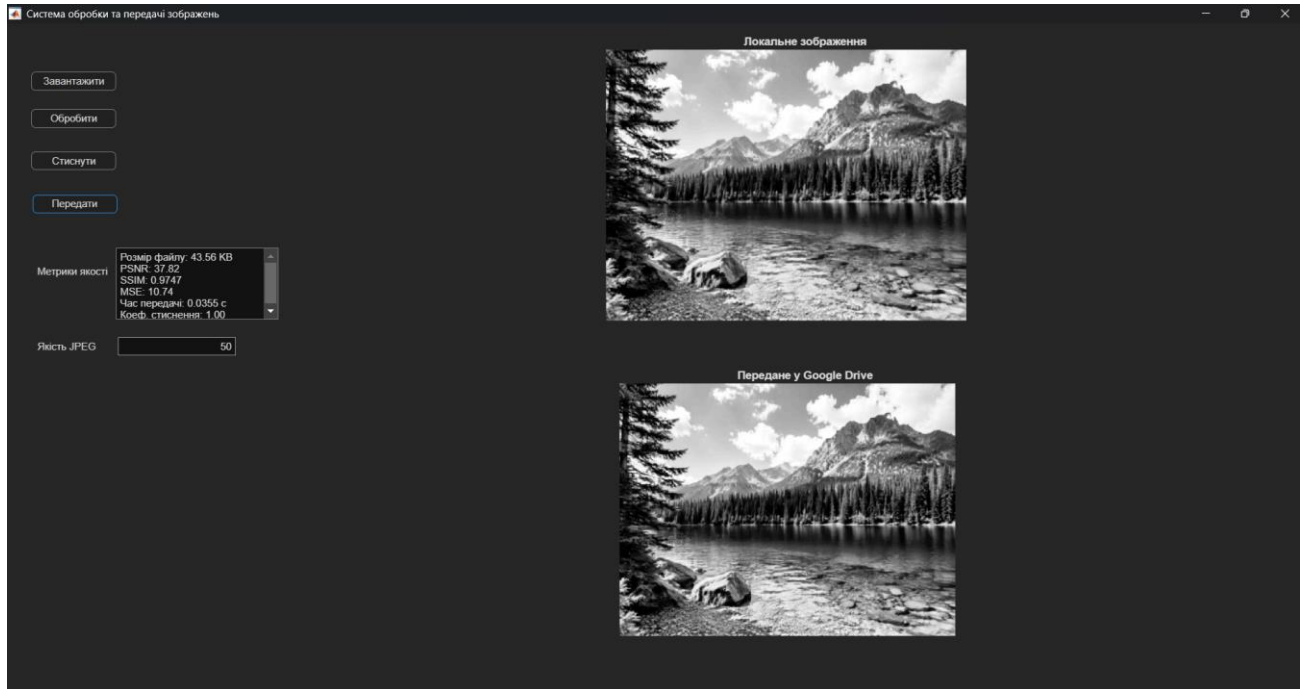


Рисунок 3.8 – Загальний вигляд програмної системи після виконання повного циклу обробки та передачі зображення

У межах реалізованої системи всі основні етапи роботи поєднані в одному графічному інтерфейсі користувача, що значно спрощує процес аналізу отриманих результатів та забезпечує наочність дослідження.

Після завершення обробки користувач отримує можливість одночасно переглядати локальне зображення після обробки та результат передачі файлу до хмарного середовища Google Drive.

На даному етапі система відображає:

- локальне зображення після виконання цифрової обробки;
- зображення, передане до Google Drive;
- значення метрик оцінювання якості;

- параметри передачі файлу;
- результати оцінювання ефективності стиснення.

Під час роботи система автоматично виконує попередню обробку зображення, фільтрацію шумів, покращення контрастності та JPEG-стиснення із заданим параметром якості. Після завершення стиснення сформований файл копіюється до локальної папки, синхронізованої із сервісом Google Drive, що забезпечує автоматичне завантаження даних до хмарного середовища.

Однією з основних переваг реалізованої системи є можливість одночасного аналізу як візуальних, так і числових характеристик результату обробки. У спеціальному інформаційному полі програма автоматично виводить основні параметри роботи системи, серед яких:

- розмір стисненого файлу;
- значення PSNR;
- значення SSIM;
- значення MSE;
- час передачі файлу;
- коефіцієнт стиснення.

Використання метрики PSNR дозволяє оцінити рівень спотворень між початковим та стисненим зображенням.

Метрика SSIM характеризує структурну подібність між зображеннями, а MSE визначає середньоквадратичну похибку між відповідними пікселями. Час передачі використовується для оцінювання швидкодії системи під час копіювання файлу до хмарного сховища.

Як видно з рисунка 3.8, система забезпечує повну інтеграцію всіх етапів цифрової обробки зображень у межах єдиного програмного середовища. Користувач має можливість оперативно оцінювати результати роботи алгоритмів, аналізувати вплив параметрів стиснення на якість зображення та контролювати процес передачі даних до хмарного середовища.

Реалізований підхід дозволяє значно підвищити наочність дослідження, оскільки користувач одночасно отримує доступ як до графічних результатів, так і до числових характеристик ефективності роботи системи.

Це створює можливість комплексного аналізу впливу методів цифрової обробки на якість зображення та ефективність його передачі.

Отримані результати підтверджують працездатність розробленої системи та демонструють ефективність інтеграції методів цифрової обробки зображень із сучасними хмарними технологіями передачі та зберігання даних. Реалізоване програмне забезпечення може бути використане як основа для подальшої розробки систем дистанційної передачі графічної інформації, мультимедійних сервісів та хмарних платформ обробки зображень.

3.8 Аналіз експериментальних результатів

У процесі дослідження було проведено серію експериментів для визначення впливу параметра якості JPEG на характеристики зображення після стиснення. Для цього використовувалося одне й те саме тестове зображення, до якого послідовно застосовувалися різні рівні якості JPEG. Після виконання стиснення для кожного значення параметра проводився розрахунок метрики PSNR, яка дозволяє оцінити рівень втрат якості між оригінальним та відновленим зображенням.

Отримані результати були використані для побудови графічної залежності між параметром якості JPEG та значенням PSNR. Аналіз такої залежності дозволяє визначити, як змінюється якість зображення при різних ступенях стиснення, а також оцінити ефективність використання алгоритму JPEG у системах цифрової обробки та передачі графічної інформації.

Під час експерименту особлива увага приділялася пошуку оптимального співвідношення між розміром файлу та якістю відновленого зображення. Це має важливе практичне значення для систем передачі даних, оскільки надмірне

стиснення дозволяє зменшити обсяг переданої інформації, проте може призвести до суттєвих втрат деталей зображення.

Для наочного представлення результатів було побудовано графік залежності показника PSNR від рівня якості JPEG. Отриманий графік наведено на рисунку 3.9.

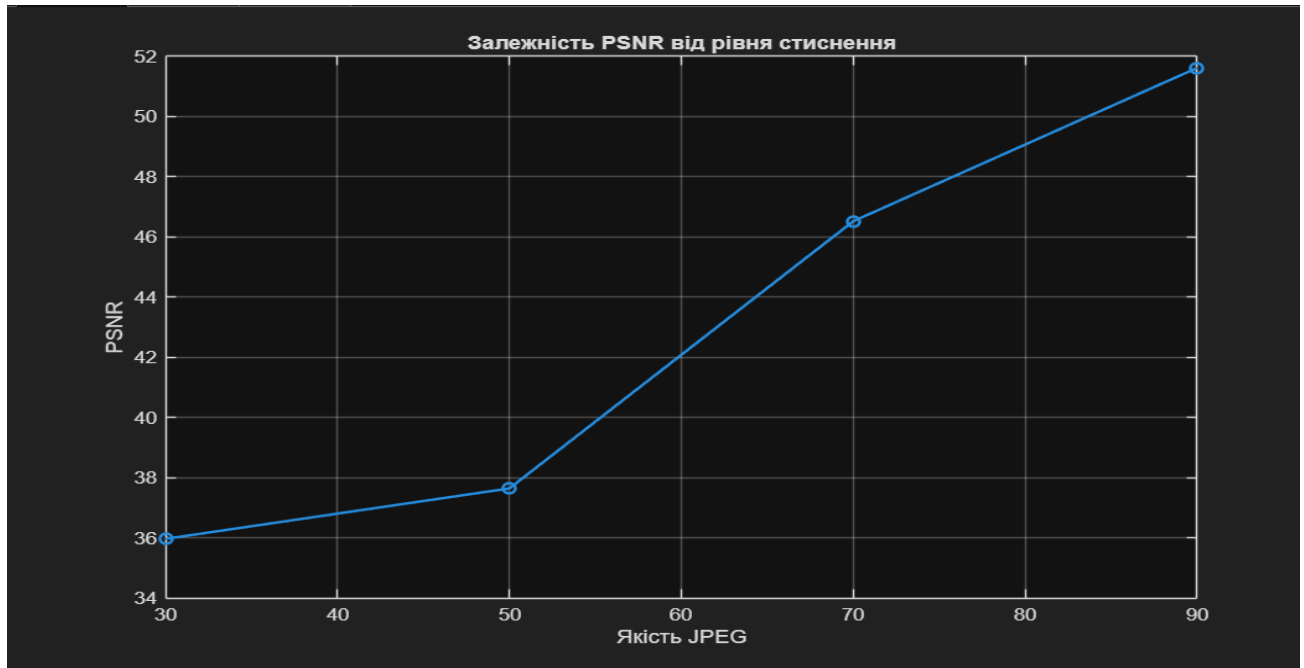


Рисунок 3.9 – Залежність PSNR від рівня якості JPEG

Як видно з рисунка 3.9, зі збільшенням значення параметра якості JPEG спостерігається зростання показника PSNR. Це свідчить про зменшення рівня спотворень у зображенні. При високих значеннях якості (близько 90) показник PSNR досягає максимальних значень, що відповідає мінімальним втратам якості. Водночас при зменшенні параметра якості до значень 30-50 спостерігається різке зниження PSNR, що вказує на суттєве погіршення якості зображення.

Під час експериментального дослідження також було проаналізовано залежність розміру файлу від рівня якості JPEG-стиснення. Зміна параметра якості безпосередньо впливає на обсяг графічних даних, що передаються мережею.

Також на швидкість передачі інформації та необхідний обсяг пам'яті для зберігання файлів. На рисунку 3.10 представлено залежність розміру файлу від рівня якості JPEG.

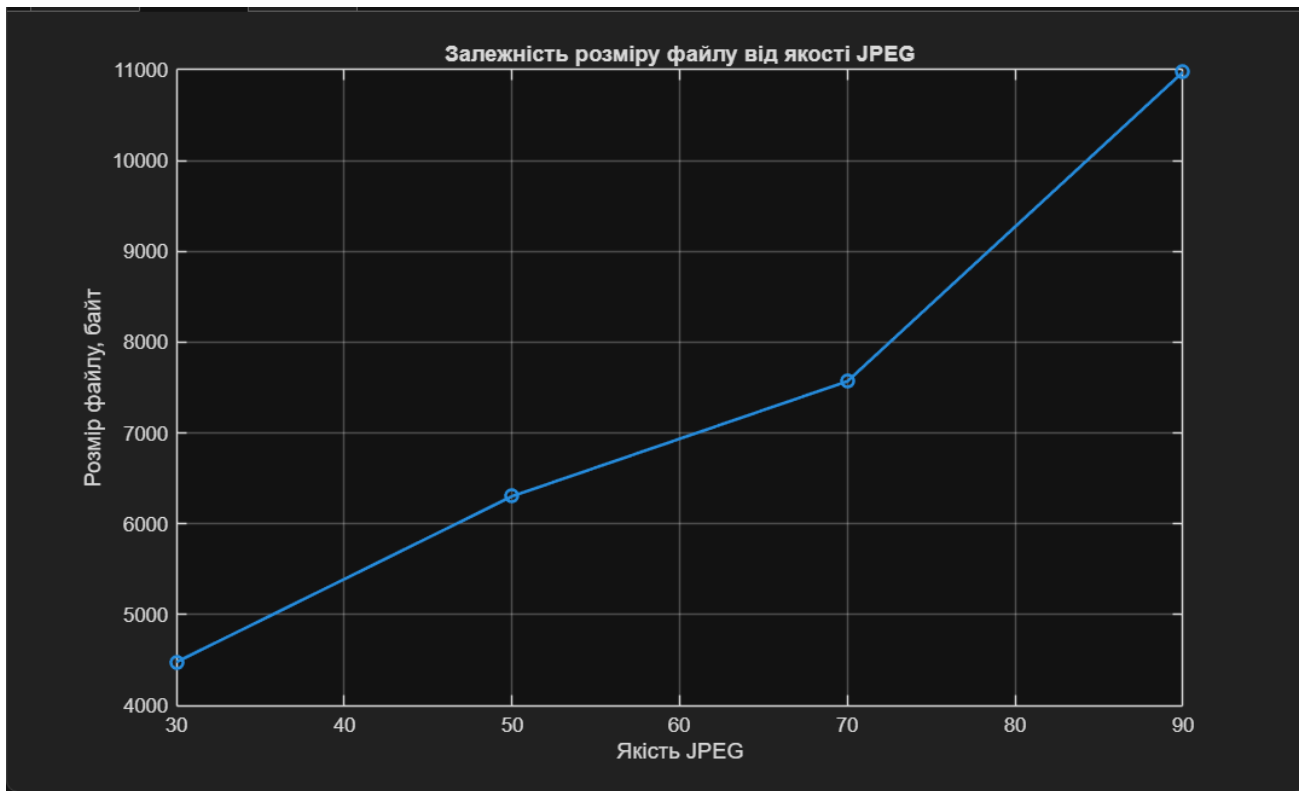


Рисунок 3.10 – Залежність розміру файлу від якості JPEG

Аналіз графіка на рисунку 3.10 показує, що зі збільшенням параметра якості JPEG значно зростає розмір файлу. При максимальній якості обсяг даних є найбільшим, тоді як при зменшенні якості файл суттєво зменшується.

Для більш детального оцінювання якості відновленого зображення було проведено аналіз показника SSIM, який характеризує ступінь структурної подібності між оригінальним та стисненим зображенням. На відміну від метрики PSNR, показник SSIM враховує особливості людського візуального сприйняття та дозволяє оцінити збереження структурних елементів, контрасту й яскравості зображення після стиснення.

У процесі експерименту значення параметра якості JPEG змінювалися у визначеному діапазоні, після чого для кожного випадку виконувалося обчислення показника SSIM.

Отримані результати були використані для побудови графіка залежності SSIM від рівня якості JPEG. На рисунку 3.11 наведено залежність показника SSIM від рівня якості JPEG.

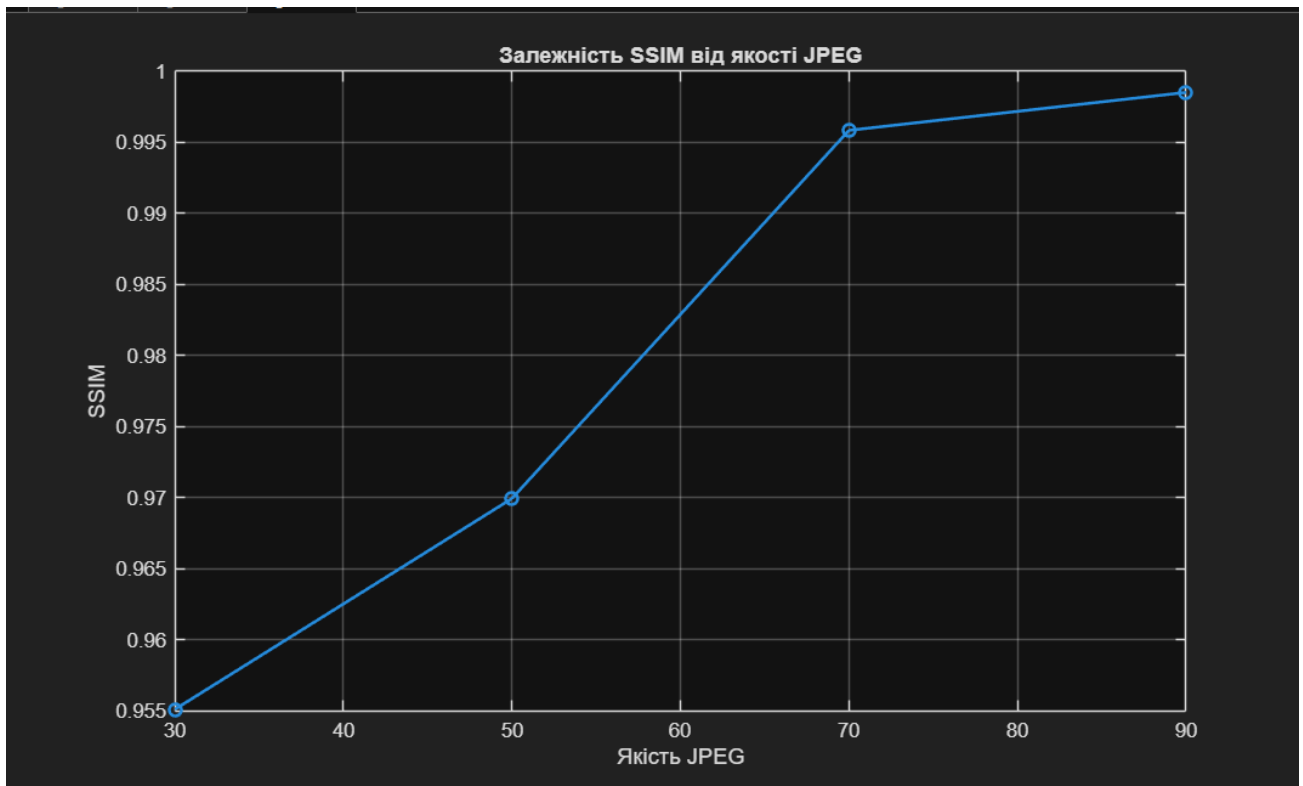


Рисунок 3.11 – Залежність SSIM від рівня якості JPEG

Як видно з рисунка 3.11, показник SSIM зменшується значно повільніше порівняно з PSNR. Це означає, що навіть при значному зменшенні якості JPEG структура зображення зберігається на достатньо високому рівні.

Отримані результати свідчать про те, що людське сприйняття якості зображення менш чутливе до деяких видів спотворень, які суттєво впливають на PSNR, але незначно змінюють SSIM.

Таким чином, проведений аналіз підтверджує наявність компромісу між якістю зображення та обсягом даних. З одного боку, підвищення якості JPEG забезпечує покращення візуальних характеристик, однак призводить до збільшення розміру файлу. З іншого боку, зменшення якості дозволяє значно скоротити обсяг даних, проте супроводжується зниженням якості.

3.9 Порівняння методів фільтрації

Для візуального порівняння ефективності різних методів фільтрації було проведено обробку одного й того ж зображення із застосуванням середнього, медіанного та гаусового фільтрів.

Кожен метод фільтрації використовує власний принцип обробки графічних даних, що безпосередньо впливає на кінцеву якість зображення. Відмінності між алгоритмами проявляються у ступені пригнічення шумів, рівні збереження контурів об'єктів та деталізації окремих елементів зображення. Результати застосування різних методів наведено на рисунку 3.12.

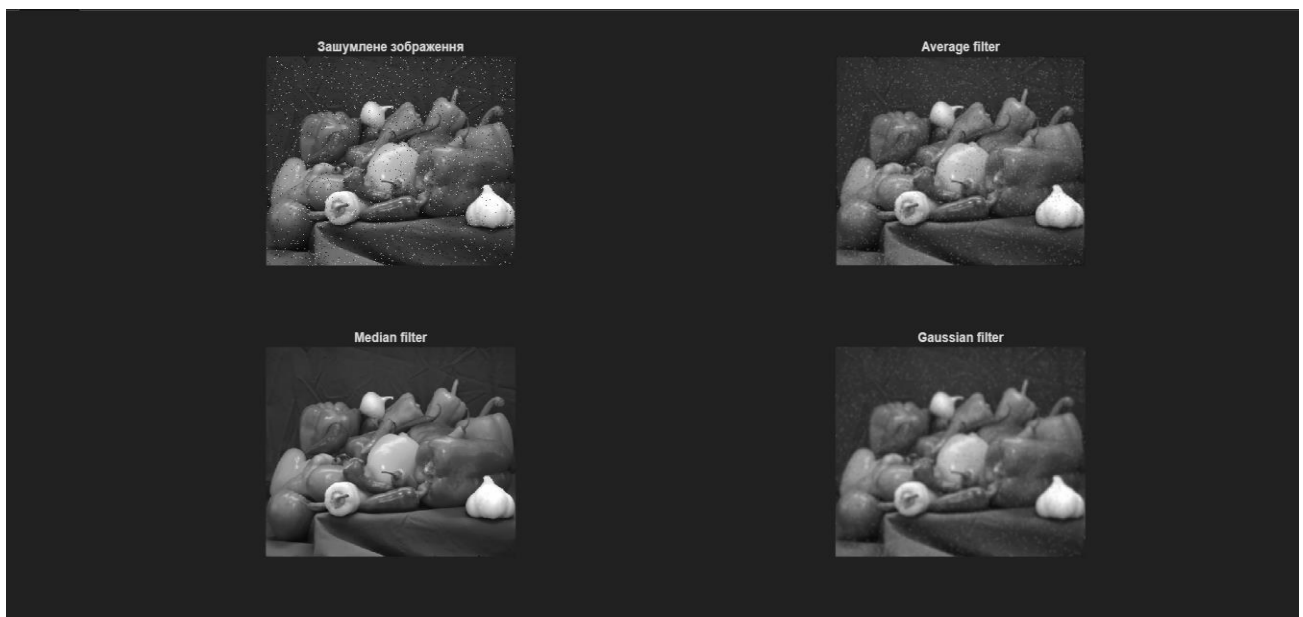


Рисунок 3.12 – Порівняння методів фільтрації

Як видно з рисунка 3.12, кожен із розглянутих методів по-різному впливає на структуру зображення, рівень шумів та збереження деталей.

Середній фільтр забезпечує ефективне згладжування шуму шляхом усереднення значень сусідніх пікселів. Однак такий підхід призводить до значного розмиття зображення, зокрема втрати дрібних деталей і згладжування контурів об'єктів.

Медіанний фільтр демонструє кращі результати при усуненні імпульсних шумів (типу «сіль і перець»). На відміну від середнього фільтра, він замінює значення пікселя на медіану значень у заданому вікні, що дозволяє ефективно усувати шум без значного впливу на контури.

Гаусовий фільтр забезпечує компроміс між згладжуванням шуму та збереженням деталей. Його особливістю є використання вагових коефіцієнтів, що розподіляються за нормальним (гаусовим) законом. Це дозволяє сильніше враховувати центральні пікселі та зменшити вплив віддалених, що знижує рівень розмиття порівняно із середнім фільтром. .

Порівняльний аналіз показує, що вибір методу фільтрації залежить від типу шуму та вимог до якості обробки. Середній фільтр є простим у реалізації, але призводить до втрати деталей. Медіанний фільтр є найбільш ефективним для усунення імпульсних шумів і збереження контурів. Гаусовий фільтр забезпечує збалансований результат і може використовуватися як універсальний метод попередньої обробки. Таким чином, результати дослідження підтверджують, що використання медіанного або гаусового фільтра є більш доцільним у задачах підготовки зображень до стиснення та передачі.

3.10 Результати експерименту

У ході експериментального дослідження було проведено серію тестувань для оцінювання впливу параметра якості JPEG на характеристики зображення після

стиснення. Основною метою експерименту є визначення залежності між рівнем стиснення, розміром отриманого файлу та показниками якості відновленого зображення.

Для проведення дослідження використовувалось одне тестове зображення, до якого застосовувалося JPEG-стиснення з різними значеннями параметра якості. Після кожного етапу стиснення здійснювалося обчислення основних метрик оцінювання якості, зокрема PSNR, SSIM та MSE.

Також визначався розмір сформованого файлу, що дозволило оцінити ефективність стиснення з точки зору економії дискового простору та можливостей передачі даних мережею.

Показник PSNR використовується для кількісної оцінки рівня спотворень після стиснення: чим більше його значення, тим ближчим є відновлене зображення до оригіналу. Метрика SSIM характеризує ступінь збереження структури та візуального сприйняття зображення, а MSE дозволяє оцінити середньоквадратичну помилку між оригінальним і стисненим зображеннями. Комплексне використання цих показників забезпечує більш повний аналіз ефективності JPEG-стиснення.

Кількісні результати експерименту при різних значеннях якості JPEG наведено у таблиці 3.1.

Таблиця 3.1 – Результати експериментального дослідження

Quality	Size (bytes)	PSNR	SSIM	MSE
90	10975	51,619	0,99852	0,44788
70	7570	46,523	0,99585	1,4481
50	6303	37,642	0,96993	11,191
30	4478	35,966	0,95509	16,463

Порівняльний аналіз показує, що:

- при зниженні якості з 90 до 30 розмір файлу зменшується більш ніж у 2 рази;
- показник PSNR знижується, що свідчить про втрату якості;
- SSIM залишається на достатньо високому рівні, що означає збереження структури.

Отримані результати підтверджують, що при зниженні якості зображення розмір файлу суттєво зменшується, проте при цьому спостерігається поступове зниження показників якості. Найбільш доцільним є використання середніх значень якості, які забезпечують оптимальний баланс між розміром файлу та якістю зображення.

ВИСНОВКИ

У результаті виконання дипломної роботи було розроблено програмну систему цифрової обробки, стиснення та передачі зображень засобами середовища MATLAB. Основною метою роботи було створення системи, яка забезпечує підготовку графічних даних до ефективної передачі мережею із збереженням прийнятної якості зображення та можливістю оцінювання результатів обробки.

У процесі виконання роботи було проведено аналіз сучасних методів цифрової обробки зображень, досліджено алгоритми фільтрації, покращення контрастності та JPEG-стиснення, а також розглянуто особливості передачі графічної інформації у комп'ютерних мережах та хмарних середовищах зберігання даних.

На основі проведеного аналізу було обґрунтовано вибір середовища MATLAB як основного інструмента реалізації системи. Використання MATLAB та пакета Image Processing Toolbox дозволило об'єднати всі етапи обробки, стиснення, передачі та аналізу зображень у межах єдиного програмного середовища.

У межах практичної частини роботи реалізовано інтерактивний програмний додаток із графічним інтерфейсом користувача, створеним засобами App Designer. Розроблений інтерфейс забезпечує можливість завантаження зображення, виконання попередньої обробки, JPEG-стиснення, передачі файлу до Google Drive та автоматичного оцінювання якості отриманого результату.

У роботі реалізовано алгоритми:

- перетворення зображення у відтінки сірого;
- гаусового згладжування;
- покращення контрастності;
- масштабування;
- JPEG-стиснення із регулюванням параметра якості.

Для організації передачі даних було використано механізм синхронізації локальної папки із сервісом Google Drive. Це дозволило реалізувати автоматичне передавання стисненого зображення до хмарного середовища та забезпечити доступ до файлів із різних пристроїв.

У процесі дослідження було проведено оцінювання ефективності системи за допомогою метрик PSNR, SSIM та MSE. Додатково визначались розмір файлу, коефіцієнт стиснення та час передачі зображення до хмарного середовища. Отримані результати підтвердили, що реалізована система забезпечує ефективне стиснення графічних даних при збереженні достатнього рівня якості зображення.

Проведене експериментальне дослідження показало, що зі збільшенням параметра якості JPEG покращуються показники PSNR та SSIM, однак одночасно збільшується розмір файлу. Це підтверджує наявність компромісу між якістю зображення та ефективністю стиснення.

Розроблена система демонструє можливість ефективної інтеграції методів цифрової обробки зображень із сучасними хмарними технологіями передачі та зберігання даних. Практичне значення отриманих результатів полягає у можливості використання розробленого програмного забезпечення для організації передачі графічної інформації у комп'ютерних мережах, мультимедійних системах та хмарних сервісах.

Таким чином, поставлену мету дипломної роботи досягнуто, а всі визначені завдання виконано в повному обсязі.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Azad R., Kazerouni A., Heidari M. et al. Advances in Medical Image Analysis with Vision Transformers: A Comprehensive Review. 2023. P. 1-9. URL: <https://arxiv.org/pdf/2301.03505> (date of access: 28.03.2026).
2. Christudhas C., et al. Recent trends in pixel-based image enhancement techniques: A review // Results in Engineering. 2025. P. 1-9. URL: <https://www.sciencedirect.com/science/article/pii/S2590123025005596> (date of access: 28.03.2026).
3. Guo J., Li Y., Ling H. A Survey on Image Enhancement for Low-Light Images // Displays. 2023. Vol. 77. URL: <https://pmc.ncbi.nlm.nih.gov/articles/PMC10070385/> (date of access: 05.04.2026).
4. Jacob J. Discrete Cosine Transform in JPEG Compression. 2021. P. 1-5. URL: <https://arxiv.org/abs/2102.06968> (date of access: 28.03.2026).
5. Jamil S., Hamidouche W., Déforges O. Review of Image Quality Assessment Methods for Compressed Images // Journal of Imaging. 2024. Vol. 10(5). P. 2-15. URL: <https://www.mdpi.com/2313-433X/10/5/113> (date of access: 06.04.2026).
6. Kosek M., Shreedhar T., Bajpai V. Beyond QUIC v1: A First Look at Recent Transport Layer IETF Standardization Efforts. 2021. P. 1-6. URL: <https://arxiv.org/pdf/2102.07527> (date of access: 06.04.2026).
7. MathWorks. Digital Image Processing. URL: <https://www.mathworks.com/discovery/digital-image-processing.html> (date of access: 04.04.2026).
8. MathWorks. Image Processing Toolbox Documentation. URL: <https://www.mathworks.com/help/images/> (date of access: 03.04.2026).
9. MathWorks. System Design and Simulation with MATLAB and Simulink. URL: <https://www.mathworks.com/solutions/system-modeling-simulation.html> (date of access: 03.04.2026).

10. MathWorks. TCP/IP Communication in MATLAB Documentation. 2024. URL: <https://www.mathworks.com/help/matlab/network-communication.html> (date of access: 03.04.2026).
11. MathWorks. TCP/IP Communication in MATLAB. URL: <https://www.mathworks.com/help/instrument/tcpip.html> (date of access: 28.03.2026).
12. McAndrew A. An Introduction to Digital Image Processing with MATLAB. P. 158-172. URL: <https://bit.ly/43mHasV> (date of access: 05.04.2026).
13. Perez-Meana H., Nakano-Miyatake M. Digital Image Processing: Technologies and Applications // Applied Sciences. 2025. Vol. 15(23). P. 1-6. URL: <https://www.mdpi.com/2076-3417/15/23/12709> (date of access: 28.03.2026).
14. Taubman D. S., Marcellin M. W. JPEG2000: Image Compression Fundamentals, Standards and Practice. Springer, 2022. P. 179 - 188. URL: <https://bit.ly/4dWtVVT> (date of access: 05.04.2026).
15. Кобилін О.А. Методи цифрової обробки зображень: навчальний посібник. Харків: ХНУРЕ, 2021. С. 21 - 55. URL: <https://bit.ly/3RVmpC4> (дата звернення: 04.04.2026).