

Вінницький національний технічний університет
(повне найменування вищого навчального закладу)

Факультет інтелектуальних інформаційних технологій та автоматизації
(повне найменування інституту, факультету)

Автоматизації та інтелектуальних інформаційних технологій
(повна назва кафедри)

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА
на тему:

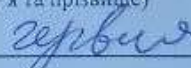
**«Розробка модифікованого методу стиснення зображень на основі
алгоритму JPEG в системах передачі інформації»**

Виконала: студентка 2-го курсу,
групи АКІТ-22мз
спеціальності 151 – Автоматизація та
комп'ютерно-інтегровані технології
(шифр і назва спеціальності)


Аліна ГРИНЬ
(ім'я та прізвище)

Керівник: д.т.н., доцент каф. АІТ


Володимир ГАРМАШ
(ім'я та прізвище)

« 17 »  2024 р.

Опонент: д.т.н., доцент каф. КСУ


В'ячеслав КОВТУН
(ім'я та прізвище)

« 18 »  2024 р.

Допущено до захисту
Завідувач кафедри АІТ
д.т.н., проф.

Олег БІСКАЛО
(ім'я та прізвище)
« 19 »  2024 р.

ВІННИЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра автоматизації та інтелектуальних інформаційних технологій
Рівень вищої освіти II-й (магістерський)
Галузь знань – 15 Автоматизація та приладобудування
Спеціальність 151 Автоматизація та комп'ютерно-інтегровані технології
Освітньо-професійна програма Інтелектуальні комп'ютерні системи управління

 **ЗАТВЕРДЖУЮ**
Завідувач кафедри АІТ
д.т.н., проф.
Олег БІСКАЛО
«12» 03 2024 року

ЗАВДАННЯ
НА МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

Гринь Аліні Олександрівні
(прізвище, ім'я, по батькові)

1. Тема роботи

Розробка модифікованого методу стиснення зображень на основі алгоритму JPEG в системах передачі інформації

керівник роботи к.т.н., доцент каф. АІТ Гармаш В.В.

Затверджені наказом ВНТУ від "11" березня 2024 року №81

2. Термін подання студентом роботи "13" червня 2024 року

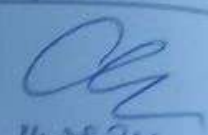
3. Вихідні дані до роботи:

тип цифрового зображення – півтонові та кольорові; мінімальна роздільна здатність зображення 128 x 128 пікселів, максимальна роздільна здатність 2048 x 2048 пікселів; коефіцієнт стиснення 2 – 200. Виграш у стисненні від 0,5% до 20% у порівнянні з класичним алгоритмом JPEG.

4. Зміст текстової частини: Вступ. Аналіз сучасних методів стиснення зображень. Застосування ортогональних перетворень для стиснення зображень. Модифікація алгоритму JPEG. Економічний розділ. Висновки.

5. Перелік ілюстративного матеріалу (з точним зазначенням обов'язкових креслень): основні етапи стиснення цифрових зображень, схема стиснення зображення по стандарту JPEG, алгоритм стиснення зображень, експериментальні дослідження, середньоквадратичне відхилення значень пікселів стислого зображення від оригіналу, порівняння методів за критерієм якості PSNR.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	виконання прийняв
1-3	к.т.н. доцент каф. АІТ Володимир ГАРМАШ	 12.03.24	 12.05.24
4	к.е.н, професор кафедри ЕПВМ Володимир КОЗЛОВСЬКИЙ	 20.05.24	 14.05.24

7. Дата видачі завдання “ 12 ” березня 2024 року

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів магістерської дипломної роботи	Строк виконання етапів роботи	При- мітка
1	Аналіз сучасних методів стиснення зображень	12.03.2024 – 26.03.2024	
2	Застосування ортогональних перетворень для стиснення зображень	27.03.2024 – 15.04.2024	
3	Модифікація алгоритму JPEG	16.04.2024 – 02.05.2024	
4	Експериментальні дослідження	03.05.2024 – 17.05.2024	
5	Економічний розділ	20.05.2024 – 27.05.2024	
6	Оформлення пояснювальної записки	28.05.2024 – 05.06.2024	
7	Попередній захист роботи	06.06.2024	
8	Остаточний захист роботи	20.06.2024	

Студент


(підпис)

Аліна ГРИНЬ

Керівник роботи


(підпис)

Володимир ГАРМАШ

АНОТАЦІЯ

УДК 004.932

Гринь А. О. Розробка модифікованого методу стиснення зображень на основі алгоритму JPEG в системах передачі інформації. Магістерська кваліфікаційна робота зі спеціальності 151 – Автоматизація та комп'ютерноінтегровані технології, освітня програма – Інформаційні системи і Інтернет речей. Вінниця: ВНТУ, 2024. 88 с.

На укр. мові. Бібліограф.: 26 назва; рис.: 21; таб.: 7.

Стиснення зображень є важливим та актуальним процесом для підвищення ефективності використання комунікаційних й інформаційно-обчислювальних ресурсів різноманітних систем.

Робота присвячена розробці модифікованого методу стиснення зображень на основі алгоритму JPEG в системах передачі інформації. Метою роботи є підвищення коефіцієнту стиснення цифрових зображень шляхом модифікації алгоритму стиснення JPEG, що дозволяє ефективно стискати зображення при мінімальному спотворенні.

В роботі проведено аналіз існуючих методів стиснення інформації, розглянуто можливість модифікації алгоритму JPEG для стиснення інформації та його переваги. Для стандартного алгоритму JPEG з постійною таблицею Хаффмана побудована модифікація JPEG-MOD. Вона має таку ж трудомісткість, але дає вииграш в стисненні до 20%. Розроблено алгоритмічне та програмне забезпечення, яке демонструє переваги запропонованого методу. Результати експериментів підтверджують ефективність запропонованого підходу.

Ключові слова: стиснення, зображення; алгоритм JPEG, коефіцієнт стиснення.

ABSTRACT

A. O. Gryn. Development of a modified image compression method based on the JPEG algorithm in information transmission systems. Master's thesis on specialty 151 - Automation and computer-integrated technologies, educational program - Information systems and the Internet of Things. Vinnytsia: VNTU, 2024. 88 p.

In Ukrainian language Bibliography: 26 titles; Fig.: 21; tab.: 7.

Image compression is an important and relevant process for increasing the efficiency of the use of communication and information-computing resources of various systems.

The work is devoted to the development of a modified image compression method based on the JPEG algorithm in information transmission systems. The aim of the work is to increase the compression ratio of digital images by modifying the JPEG compression algorithm, which allows for efficient image compression with minimal distortion.

The paper analyzes the existing methods of information compression, considers the possibility of modifying the JPEG algorithm for information compression and its advantages. A modification of JPEG-MOD was built for the standard JPEG algorithm with a constant Huffman table. It has the same labor intensity, but gives a gain in compression of up to 20%. Algorithmic and software have been developed that demonstrate the advantages of the proposed method. The results of the experiments confirm the effectiveness of the proposed approach.

Keywords: compression, image; JPEG algorithm, compression ratio.

ЗМІСТ

ВСТУП.....	4
1 АНАЛІЗ СУЧАСНИХ МЕТОДІВ СТИСНЕННЯ	
ЗОБРАЖЕНЬ.....	6
1.1 Представлення цифрових зображень	6
1.2 Класифікація методів стиснення. Основні характеристики	8
1.3 Алгоритми стиснення зображень без втрат	11
1.4 Алгоритми стиснення з втратами	13
1.5 Вибір і обґрунтування аналогу	22
1.6 Висновки до розділу.....	23
2 ЗАСТОСУВАННЯ ОРТОГОНАЛЬНИХ ПЕРЕТВОРЕНЬ ДЛЯ	
СТИСНЕННЯ ЗОБРАЖЕНЬ.....	24
2.1 Перетворення Адамара.....	24
2.2 Перетворення Карунена-Лоева	27
2.3 Дискретне косинусне перетворення	29
2.4 Дискретне вейвлет-перетворення	31
2.5 Висновки до розділу	34
3 МОДИФІКАЦІЯ АЛГОРИТМУ JPEG	35
3.1 Baseline JPEG	35
3.2 Алгоритм JPEG-MOD	40
3.3 Методика порівняння	41
3.4 Аналіз результатів	42
4 ЕКОНОМІЧНИЙ РОЗДІЛ	45
4.1 Технологічний аудит розробленого модифікованого методу стиснення зображень на основі алгоритму JPEG в системах передачі інформації.....	45
4.2 Розрахунок витрат на розроблення модифікованого методу стиснення зображень	50
4.3 Прогнозування економічного ефекту від можливої комерціалізації розробки.....	55

4.4 Висновки до економічного розділу	63
ВИСНОВКИ	64
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	65
ДОДАТКИ	69
Додаток А (обов’язковий) – Технічне завдання	70
Додаток Б (обов’язковий) – Ілюстративна частина	71
Додаток В (обов’язковий) Лістинг модуля програми.....	80
Додаток Г (обов’язковий) Протокол перевірки магістерської кваліфікаційної роботи на наявність текстових запозичень	87

ВСТУП

Актуальність теми

Сьогодні одним із ключових факторів, що впливають на частоту відвідування сайту користувачами, є швидкість його завантаження. Особливу увагу слід приділити зображенням, оскільки вони складають значну частину контенту. Зображення повинні бути достатньо високої якості, але при цьому мати невеликий розмір файлу. Це особливо важливо з огляду на сучасні тенденції, оскільки все більше користувачів використовують мобільний інтернет з низькою швидкістю. Розвиток сучасних інформаційних систем і мереж призвів до широкого використання цифрових зображень. Сьогодні багато галузей техніки, пов'язаних з отриманням, обробкою, зберіганням і передачею інформації, значною мірою орієнтуються на системи, де інформація представлена у вигляді зображень. Стиснення зображень є важливим для підвищення ефективності використання комунікаційних і інформаційно-обчислювальних ресурсів різноманітних систем.

Метою даної роботи є підвищення коефіцієнта стиснення цифрових зображень шляхом модифікації алгоритму JPEG, що дозволяє ефективно стискати зображення з мінімальним спотворенням.

Об'єктом дослідження є процес стиснення цифрових зображень.

Предметом дослідження є алгоритм JPEG для стиснення цифрових зображень.

Для досягнення поставленої мети необхідно розв'язати наступні **задачі**:

- Провести аналіз існуючих методів стиснення зображень.
- Розробити модифікований метод стиснення на основі алгоритму JPEG.
- Зробити практичну реалізацію та провести експериментальні дослідження розробленого методу.

Методи дослідження. Для вирішення поставлених задач використовувалися методи лінійної алгебри та цифрової обробки зображень

у процесі стиснення цифрових кольорових зображень, а також методи статистичного аналізу для обробки і тестування отриманих результатів.

Наукова новизна(інноваційність) одержаних результатів:

Удосконалено метод стиснення зображень на основі алгоритму JPEG за рахунок зміни схеми кодування, що дозволило підвищити ступінь стискання зображень в середньому на 15 – 20% порівняно з базовим кодеком JPEG залежно від необхідної якості та вихідного розміру зображення.

Практичне значення одержаних у роботі результатів полягає в тому, що на основі запропонованого методу розроблено та програмно реалізовано алгоритм стиснення цифрових зображень, який дозволяє підвищити ступінь стиснення зображень при збереженні прийнятної якості. Запропонований метод може використовуватися для пришвидшення завантаження сайтів в Інтернеті та в системах передачі інформації.

Апробація результатів дослідження.

Представлені в роботі результати апробовані в результаті участі в конференції ВНТУ: «Науково-технічна конференція факультету інтелектуальних інформаційних технологій та автоматизації (2024)».

Публікації. Гринь Аліна Олександрівна, Гармаш Володимир Володимирович «Метод стиснення зображень на основі модифікованого алгоритму JPEG», «Науково-технічна конференція факультету інтелектуальних інформаційних технологій та автоматизації (2024)», 2024.
URL: <http://surl.li/ulyug>

1. АНАЛІЗ СУЧАСНИХ МЕТОДІВ СТИСНЕННЯ ЗОБРАЖЕНЬ

1.1. Представлення цифрових зображень

Комп'ютерне зображення в його цифровому представленні є набором значень інтенсивностей світлового потоку, розподілених по кінцевій площі.

Для простоти розглянемо спочатку монохромні зображення. Інтенсивність випромінювання світлової енергії з одиниці поверхні в точці з координатами (ξ, η) зображення можна представити деяким числом $B(\xi, \eta)$. Одиничний елемент зображення, що характеризується певним значенням (ξ, η) , називається пікселем, а величина $z = f(\xi, \eta)$ – яскравістю [7].

Перш ніж розглянути алгоритми стиснення зображень, необхідно визначити що надалі буде розумітися під зображенням.

З математичної точки зору, зображення в градаціях сірого можна представити як дійсну функцію I двох дійсних змінних x і y . Функція $I(x, y)$ зображення в загальному випадку визначається в прямокутній області, але для зручності досліджень в роботі всі зображення визначаються в квадратних областях, тобто $x \in [0; W]$, а $y \in [0; H]$, де W – ширина зображення, а H – висота зображення і $W = H$.

Всі зображення можна поділити на дві групи: з палітрою і без неї. У зображень з палітрою у пікселі зберігається число – індекс в деякому одновимірному векторі кольорів, званому палітрою. Палітри зазвичай бувають 8, 16 і 256 кольорів[7].

Зображення без палітри зазвичай бувають в певній системі представлення кольору або в градаціях сірого. В градаціях сірого значення кожного з пікселів визначається як яскравість точки. Найбільш часто зустрічаються зображення з 2-ма, 16-ма і 256-ю рівнями сірого.

Якщо зображення відображається в якійсь системі представлення кольору, то кожен її піксель є структурою, що описує компоненти кольору. Найбільш поширеною системою представлення кольору, що

використовується в електронних та комп'ютерних системах, є система RGB. У цій системі колір визначається як комбінація червоного, зеленого та синього кольору. І на кожному з складових припадає по одному байту. Існують і інші системи представлення кольору, такі як CMYK, CIE, YUV і YCrCb[4].

Для того, щоб коректніше оцінити ступінь стиснення зображення та застосовності того чи іншого алгоритму стиснення до даного зображення вводиться поняття класу зображення.

Під класом цифрового зображення розуміється сукупність зображень, для яких застосування алгоритму стиснення дає якісно однаковий результат. Наприклад, для одного класу зображень алгоритм стиснення забезпечує високий коефіцієнт стиснення, тоді як для іншого класу зображень він може навіть збільшувати об'єм стиснутого файлу [11].

У процесі роботи із зображеннями додатки, які здійснюють обробку, висувають різні вимоги до алгоритмів стиснення. Через специфіку цих додатків такі вимоги можуть іноді суперечити одна одній. У загальному випадку можна виділити наступні основні вимоги до алгоритмів стиснення зображень:

- високий ступінь компресії;
- висока якість стисненого зображення (дана вимога суперечить виконанню попередньої вимоги, тому завжди доводиться шукати компроміс між ступенем стиснення і якістю відновленого зображення);
- висока швидкість компресії (дана вимога актуально для додатків, що займаються кодуванням зображень в реальному масштабі часу: цифрових фотоапаратів, відеокамер);
- висока швидкість декомпресії (дана вимога актуально майже для всіх додатків);
- можливість показати приблизне зображення, не чекаючи повної його завантаження (дана вимога актуально для мережеских додатків і для додатків, що займаються передачею великих зображень);

- врахування специфіки зображення (дану вимога реалізують алгоритми стиснення, засновані на визначенні «області особливого призначення» (ROI - regionsofinterest)).

1.2. Класифікація методів стиснення. Основні характеристики

Всі методи стиснення інформації ґрунтуються на простому припущенні, що набір даних завжди містить надлишкові елементи. Стиснення досягається шляхом пошуку та кодування цих надлишкових елементів. Потік даних про зображення містить значну кількість зайвої інформації, яку можна усунути практично без помітних для ока спотворень. При цьому розрізняють два типи надмірності.

Статистична надлишковість пов'язана з кореляцією і передбачуваністю даних. Ця надмірність може бути усунена без втрати інформації, при цьому вихідні дані можуть бути повністю відновлені. Найбільш відомі методи ефективного кодування символів ґрунтуються на знанні частоти кожного символу, присутнього в повідомленні. Знаючи ці частоти, створюють таблицю кодів, яка володіє наступними властивостями:

- різні коди можуть мати різну кількість біт;
- коди символів, які зустрічаються з більшою частотою, мають менше біт, ніж коди символів з меншою частотою;
- хоча коди мають різну бітову довжину, вони можуть бути відновлені єдиним чином, тобто коди будуються як префіксні.

Цими властивостями володіє відомий алгоритм Хаффмана[7].

Візуальна (суб'єктивна) надлишковість, яку можна усунути з частковою втратою даних, мало впливають на якість відтворення зображень; це – інформація, яку можна вилучити з зображення, не порушуючи візуальне сприйняття якості зображень.

Завдання стиснення зображення складається з двох основних частин: кодування і декодування. Якщо декодоване зображення завжди в точності відповідає кодованому зображенню, то такий алгоритм кодування-декодування називається алгоритмом стиснення без втрат. Якщо декодувати зображення відрізняється від кодованого, то подібний алгоритм називають алгоритмом стиснення з втратами. Загальна схема процесу стиснення зображення представлена на рисунку 1.1.

Розглянемо етапи процедури стиснення даних в загальному вигляді. Будь-який метод стиснення реалізує три основних етапи (рисунок 1.1.):

- кодування або первинне стиснення;
- вторинне стиснення;
- декодування або відновлення зображення.

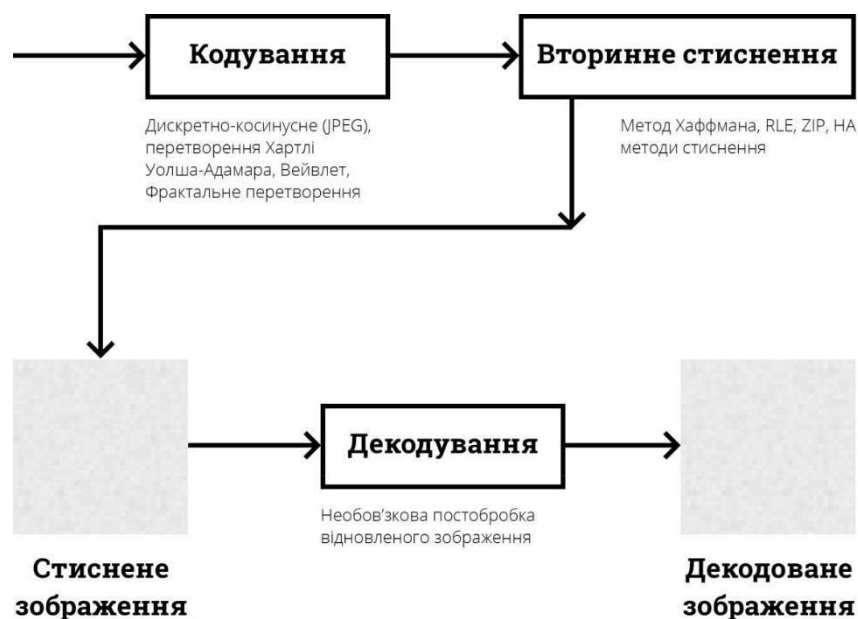


Рисунок 1.1 – Основні етапи стиснення цифрових зображень

На першому етапі виконується перетворення вихідних даних з однієї форми подання в іншу. Зокрема, при стисненні зображень залежно від виду алгоритму стиснення може бути виконаний перехід від початкового зображення до наступних видів подання (таблиця 1.1).

На другому етапі компоненти перетворення квантуються і зводяться до виду зручного для статистичного кодування, а потім кодуються. На цьому етапі забезпечується ущільнення інформаційного потоку.

Таблиця 1.1 – Переходи від початкового зображення

Вихідне зображення	Перетворене зображення
Матриця пікселів (значення інтенсивності)	Матриця компоненту спектру (спектральні перетворення)
	Набір коефіцієнтів перетворення (фрактальне стиснення)
	Опис об'єктів перетворення ()стиснення з розпізнаванням

Класифікація найбільш поширених методів стиснення наведена на рисунку 1.2.

Стиснення без втрат. Результатом стиснення без втрат завжди є зниження обсягу вихідного потоку інформації без зміни його інформативності, тобто без втрати інформаційної структури. Більше того, з вихідного потоку, за допомогою відновлюючого алгоритму, можна отримати вхідний [7].

Стиснення з втратами. Під стисненням з втратами мається на увазі таке перетворення вхідного потоку даних, при якому вихідний потік, що базується на певному форматі інформації, представляє досить схожий за зовнішніми характеристиками на вхідний потік об'єкт, проте відрізняється від нього об'ємом. Ступінь подібності вхідного і вихідного потоків визначається ступенем відповідності деяких властивостей об'єктів, що представляється даним потоком інформації.

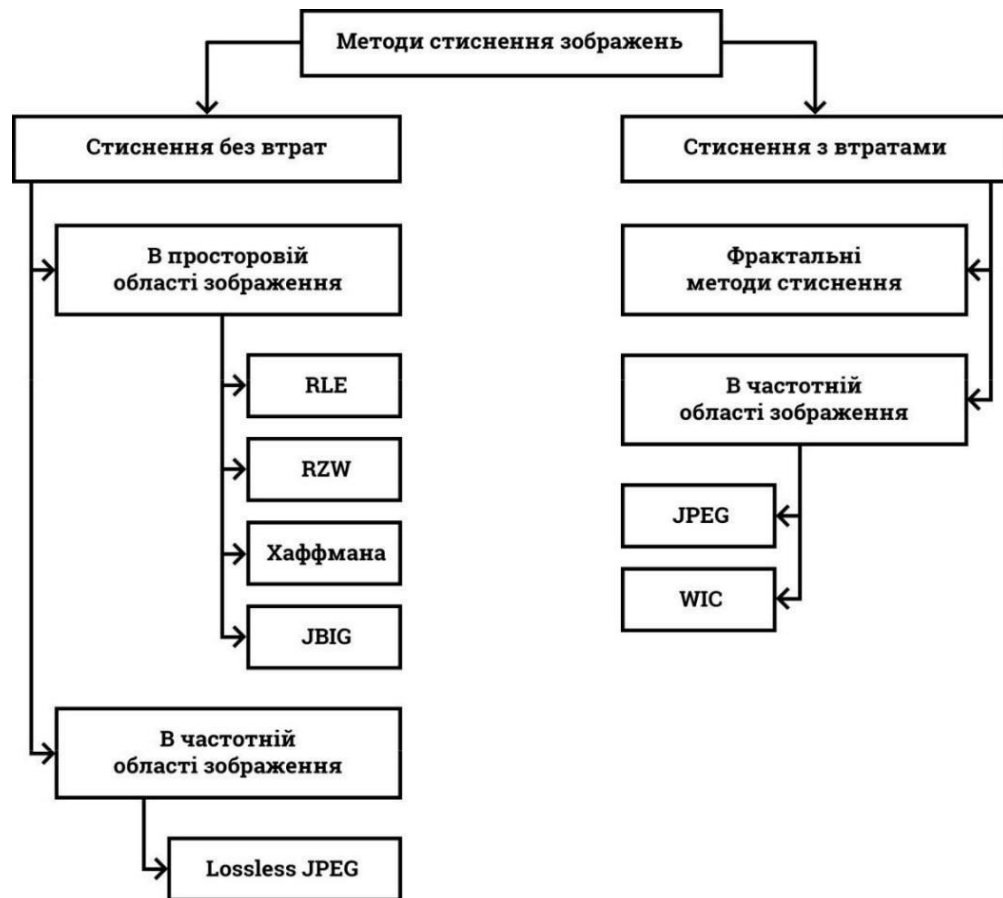


Рисунок 1.2 – Класифікація методів стиснення зображень

Для симетричних методів стиснення процедури стиснення і відновлення однотипні. Час стиснення і відновлення для таких методів порівнянні. Для несиметричних методів процедура стиснення відрізняється від процедури відновлення і зазвичай займає більше машинного часу.

1.3. Алгоритми стиснення зображень без втрат

Найбільш відомий і простий алгоритм стиснення інформації без втрат – це кодування серій послідовностей (Run Length Encoding – RLE)[9]. Суть даного підходу полягає в заміні ланцюжків або серій повторюваних байтів або їх послідовностей на один кодуєчий байт і лічильник числа їх повторень.

Проблема всіх аналогічних методів полягає лише у визначенні способу, за допомогою якого алгоритм розпакування міг би відрізнити в результуючому потоці байтів кодовану серію від інших – некованих послідовностей байтів. Рішення проблеми досягається зазвичай постановкою міток на початку кодованих ланцюжків. Такими позначками можуть бути, наприклад, характерні значення бітів в першому байті кованоїсерії, значення першого байтакованої серії і т.п. Кращий, середній і найгірший коефіцієнти стиснення $1/32$, $1/2$, $2/1$.

Дані методи, як правило, досить ефективні для стиснення растрових графічних зображень (BMP, PCX, TIFF), так як останні містять досить довгісерії повторюваних послідовностей байтів.

Недоліком методу RLE є досить низька ступінь стиснення або вартість кодування файлів з малим числом серій і, що ще гірше— з малим числом повторюваних байтів в серіях. До позитивних сторін алгоритму, мабуть, можна віднести тільки те, що він не вимагає додаткової пам'яті при роботі, і швидко виконується. Цікава особливість групового кодування у форматі PCX полягає в тому, що ступінь архівації для деяких зображень може бути істотно підвищений всього лише за рахунок зміни порядку кольорів у палітрі зображень.

Кодування Хаффмана частіше за інших вживається для текстових файлів, яке полягає в тому, що символи тексту замінюються ланцюжками біт різної довжини. Методика Хаффмана гарантує однозначну побудову коду з найменшим для даного розподілу ймовірностей середнім числом символів на букву[7].

Щодо стиснення зображень, то в основі такого методу лежить облік частоти появи однакових байт в зображенні. При цьому пікселям вихідного зображення, які зустрічаються більше число разів, зіставляється код меншої довжини, а пікселям, що зустрічаються рідко – код більшої довжини (тобто формується префіксний код змінної довжини). Для збору статистики

потрібно два проходи по файлу – один для перегляду та збору статистичної інформації, другий – для кодування. Коефіцієнти стиснення: $1/8$, $2/3$, 1 [11].

При використанні такого методу потребується запис у файл і таблиці відповідності кодованих пікселів та кодуючих ланцюжків. Таке кодування застосовується в якості останнього етапу архівації в JPEG. Методи Хаффмана дають досить високу швидкість і помірно добру якість стиснення.

Основним недоліком даного методу є залежність ступеню стиснення від близькості ймовірностей символів до величини 2^{-M} , оскільки кожен символ кодується цілим числом біт. Так, при кодуванні даних з двохсимвольним алфавітом стиснення завжди відсутнє, тому незважаючи на різні ймовірності появи символів у вхідному потоці алгоритм фактично зводить їх до $1/2$. Такий алгоритм реалізований у форматі TIFF.

1.4. Алгоритми стиснення з втратами

JPEG – один з найпоширеніших і досить потужних алгоритмів, являє собою метод стиснення зображень, що реалізується різними способами [7,9]. Працює він як на чорно-білих, так і на повнокольорових зображеннях.

Стандарт JPEG (JointPhotographicExpertsGroup - Об'єднана експертна група по фотографії) – формат зберігання фотографічних зображень, що відрізняється гарною якістю відновленого зображення. Процес стиснення зображення JPEG досить складний і часто для досягнення прийнятної продуктивності вимагає спеціальної апаратури. Схема процедури стиснення зображень за стандартом JPEG наведена на рисунку 1.3.

Кодування зображень за стандартом JPEG зазвичай починається з перетворення кольорового простору з RGB в YUV (відоме також під назвою YCbCr). Кольорове зображення традиційно може розглядатися як результат складання трьох компонент:

$$X_{ijC} = a_1 X_{ij(R)} + a_2 X_{ij(G)} + a_3 X_{ij(B)} \quad (1.1)$$

У цьому виразі a_1, a_2, a_3 — калориметричні коефіцієнти.

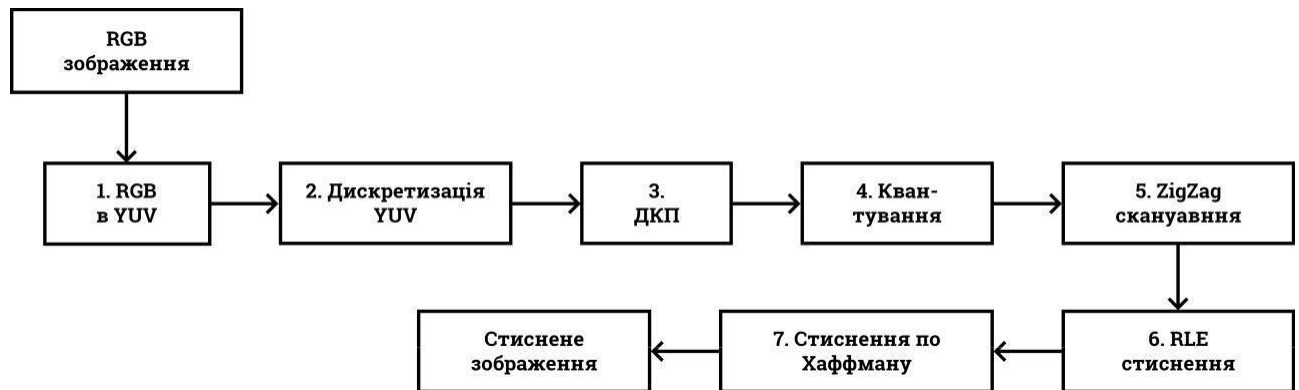


Рисунок 1.3 – Основні етапи процедури стиснення за стандартом JPEG

У типових зображеннях у форматі RGB є істотна кореляція між кольоровими компонентами і з точки зору стиснення зображення цей формат є завідомо надлишковим. Як відомо, в стандартах телевізійного мовлення використовується інше представлення зображень, при якому також використовуються 3 компоненти сигналу, але при цьому ці компоненти майже некорельовані один з одним. Компоненти R , G і B перетворюються в яскравісну компоненту Y і дві різнокольорові компоненти U і V , формату YUV.

Основу фрактального підходу становить постулат про те, що зображення реального світу мають афінну надмірність[7]. Інакше кажучи, існує набір афінних коефіцієнтів, що описують обертання, стиснення, розширення, спотворення форми, зрушення об'єктів зображення.

Найбільш поширеним прикладом фрактальної зображення, згенерованого за допомогою IFS є зображення папороті (рисунок 1.4), використане для створення даного зображення, складається з 4-х афінних перетворень. Кожне перетворення кодується ліченими байтами, хоча вихідне зображення може бути будь-якого розміру. Таким чином, можна зробити

висновок, що фрактальна компресія – це пошук самоподібних областей та визначення для них параметрів афінних перетворень.

Проста IFS, що породила зображення папороті не придатна для стиснення довільних зображень, оскільки вони зазвичай подаються у градаціях сірого, а не з значень 0 або 1 і тому прості IFS використовуються тільки для самоподібних зображень, тобто коли зображення будуються з елементів, що є копією цілого зображення.



Рисунок 1.4 – Зображення папороті, що згенеровано за допомогою IFS

В основі фрактального стиснення лежать кілька основних визначень і теорем. Розглянемо їх коротко.

Перетворення. Перетворення зіставляє точки в одному просторі точку в іншому просторі, можливо в тому ж самому просторі. Перетворення називається відображенням f і записується як $f: X_1 \rightarrow X_2$ якщо воно переводить простір X_1 в простір X_2 .

Перетворення в метричному просторі (X, d) називається стискуючим відображенням, якщо існує константа s , $0 \leq s < 1$ така що:

$$d(f(x_1), f(x_2)) \leq s \cdot d(x_1, x_2) \quad (1.2)$$

де $d(x_1, x_2)$ - відстань від точки x_1 до точки x_2 в просторі X .

Константа s називається коефіцієнтом стиснення відображення f .

На рисунку 1.5. показаний приклад стиснення відображення (R_2, d_2) , застосованого до безлічі точок в R_2 . Тут дане перетворення застосовувалося більше одного разу: спочатку $f(x)$ обчислювалося для точки x , а потім перетворення f застосовувалося до результату перетворення: $f(f(x))$. Таке послідовне, багаторазове застосування перетворення називають ітераціями і позначають як: f^n , тобто $f(\dots f(x) \dots)$, де f застосовується n раз.

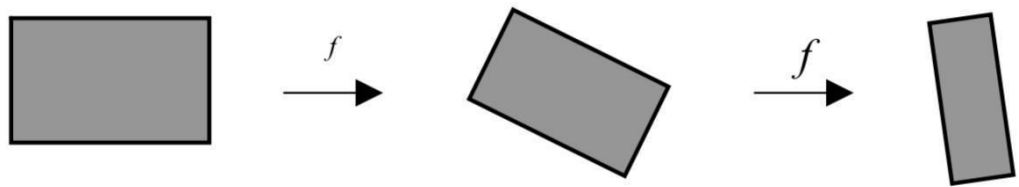


Рисунок 1.5 – Стискуjące відображення для точок у просторі R^2

Система ітеруючих функцій (IFS) складається з повного метричного простору (X, d) і кінцевого безлічі стискають відображень $w_n: X_1 \rightarrow X_2$ з коефіцієнтами стиснення s_n . Таким чином, IFS можна позначити таким чином: $\{X, w_n: n = 1, 2, \dots, N\}$, або якщо розглядати простір точок в R_2 , то просто $\{w_n\}$.

У теорії фрактального стиснення доводиться теорема колажу, яку можна сформулювати наступним чином.

Нехай L є двійкове зображення $L \subseteq R_2$ і нехай w_n є стискають відображення, такі що: $U_n^N = \bigcup_{n=1}^N w_n(L)$

Покривають L майже точно. Можна вважати таке $w_n(L)$ зменшеною копією L . Тоді теорема колажу стверджує, що аттрактор A (аттрактором IFS називається зображення, яке є єдиною нерухомою точкою IFS) системи $\{w_n\}$ близький до L . «колаж» є набір областей $w_n(L)$.

Щоб на практиці застосувати теорему колажу (рисунок 1.6), необхідно вибрати перетворення, які будуть стискаючими відображеннями. Одним з таких перетворень є афінне перетворення: $T: R^2 \rightarrow R^2$:

$$T\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} \tilde{x} \\ \tilde{y} \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix} \quad (1.3)$$

де $a, b, c, d, e, f \in \mathbb{R}$. Афінні перетворення можуть здійснювати поворот, переміщення й масштабування (рисунок 1.7).

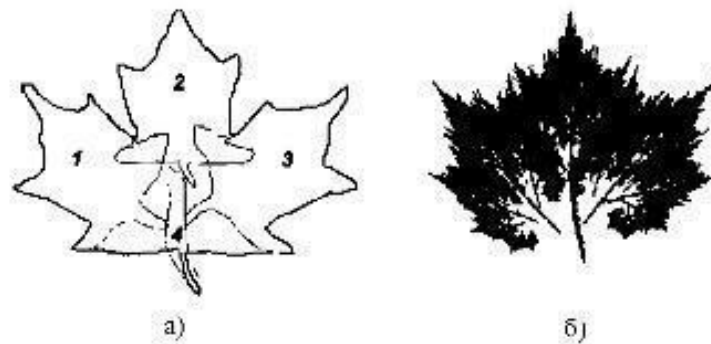


Рисунок 1.6 – Ілюстрація теореми колажу. (а) вихідне зображення і чотири фрагменти зображення; (б) зображення-аттрактор

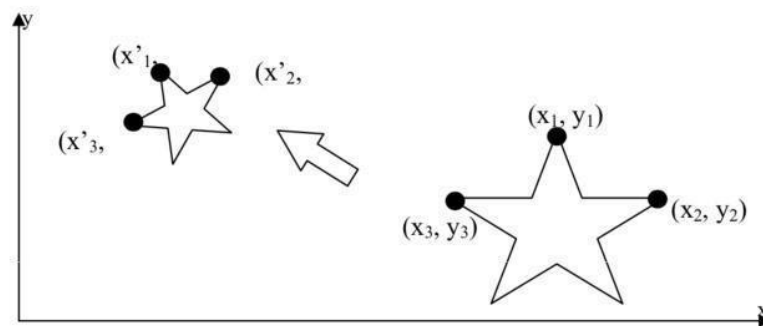


Рисунок 1.7 – Дія афінного перетворення на точки в просторі R^2

Щоб створити IFS зображення – потрібно знайти хоча б деяку приблизну самоподібність в зображенні. Потім потрібно виділити точки і перетворення IFS для кожної з фігур подібності на зображенні. Коли точки і перетворення для IFS визначені, то обчислюються коефіцієнти афінного перетворення, використовуючи систему рівнянь, засновану на виразі (1.3).

Існує два алгоритми побудови зображення-аттрактора за допомогою IFS. Один з них це пряме застосування теореми про стискають відображеннях, а інший - застосування так званої «гри хаосу». Детермінований алгоритм для побудови зображення є аттрактором IFS, до будь-якого початкового зображення B застосовує теорему про стискають відображеннях. Алгоритм будує послідовність зображень A_n , багаторазово застосовуючи IFS відображення $W = \{w_1, w_2, \dots, w_n\}$:

$$A_n = W^{on}(B) \quad (1.4)$$

Детермінований алгоритм корисний з точки зору навчання, оскільки дозволяє бачити результат перетворення на кожному кроці ітерації (рисунок 1.8).

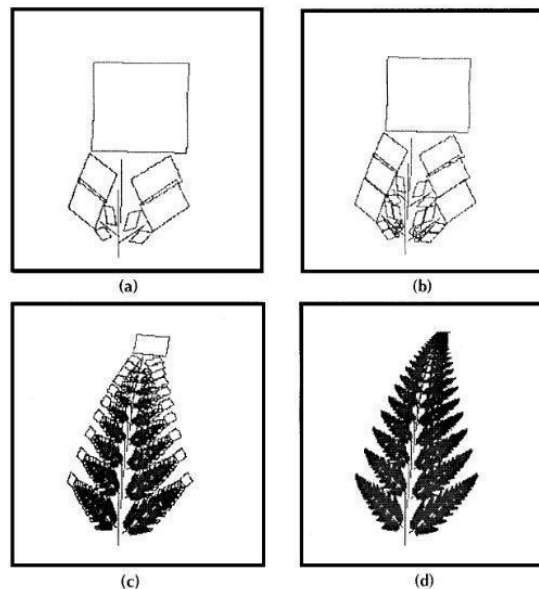


Рисунок 1.8 – Детермінований алгоритм, застосований до IFS папороті.

Вид зображення A_n після: а) 2-х, б) 3-х, с) 10-і, d) 30-й ітерацій.

Імовірнісний алгоритм пов'язує з кожним афінним перетворенням w_i в IFS ймовірність p_i . Ці ймовірності визначають, наскільки щільно кожна частина зображення-аттрактора покрита точками. Імовірнісний алгоритм

створює зображення-атрактори високої якості швидше, ніж детермінований алгоритм. Це відбувається не тільки через те, що імовірнісний алгоритм на кожному кроці ітерації виконує меншу роботу, а й сама виконана робота дає кращий результат.

Реальні зображення не володіють властивістю глобальної самоподібності, яка притаманна зображенням IFS. До того ж реальні зображення представлені в градаціях сірого, а не в двійкових значеннях, як зображення IFS. Тому при фрактальному кодуванні реальних зображень намагаються знайти безліч стискають перетворень, які відображають доменні блоки (які можуть перекриватися) в безліч рангових блоків, які покривають зображення. Рангові блоки можуть бути однакового розміру, але частіше використовується адаптивне розбиття з перемінним розміром блоку (одним з поширених методів адаптивного розбиття є метод квадродерева).

Базовий алгоритм фрактального кодування зображення виконується таким чином:

- Зображення f розбивається на не перекриваються рангові блоки $\{R_i\}$. У самому простому випадку блоки можуть являти собою прямокутники, але можуть бути й інші форми.
- Зображення покривається послідовністю доменних блоків, які можуть перетинатися. Домени можуть бути різного розміру, і їх кількість може обчислюватися сотнями і тисячами.
- Для кожного рангового блоку знаходять домен і відповідне перетворення, яке найкращим чином покриває ранговий блок. Зазвичай це афінні перетворення.
- Якщо достатньо точної відповідності не вийшло, то розбиваємо рангові блоки на менші рангові блоки. Даний процес продовжується до тих пір поки не отримують прийнятної відповідності або розмір рангового блоку досягає певних значень.

В даний час відомі наступні різновиди такого алгоритму [9].

Алгоритм Фішера. Ідея алгоритму полягає в тому, щоб певним чином класифікувати D -блоки і R -блоки, а пошук близького D -блоку виробляти в тому ж класі, до якого відноситься рангова область. Робиться це в такий спосіб [9].

Вихідні блоки розбиваються на чотири частини. Для кожної з частин підраховується середнє значення A_i і дисперсія V_i пікселів. Далі блоки класифікуються за наступним принципом. Визначимо три базових типи блоків:

$$\begin{aligned} \text{тип } 1: A_1 \geq A_2 \geq A_3 \geq A_4, \\ \text{тип } 2: A_1 \geq A_2 \geq A_4 \geq A_3, \\ \text{тип } 3: A_1 \geq A_4 \geq A_2 \geq A_3. \end{aligned} \quad (1.5)$$

Зрозуміло, що будь-який блок за допомогою відповідного афінного перетворення квадрата в квадрат можна привести до виду, відповідному одному з вказаних типів. Після того, як ми зафіксували три основні класи, блоки класифікуються за дисперсії. Таким чином, в кожному з трьох класів з'являються 24 підкласу, разом 72 класу. Пошук близького до R -блоку D -блоку проводиться перебором у відповідному класі.

Популяцією $\Pi = (\chi^1, \chi^2, \dots, \chi^M)$ чисельності M вважається вектор простору B^M , координати якого називаються генотипами особин даної популяції.

Кроком ГА є перехід від поточного покоління до наступного, тобто отримання нової популяції Π_{t+1} з Π_t . У побудові чергової особини нової популяції беруть участь оператори кросинговеру (схрещування), мутації і випадковий оператор відбору, $Select: B^M \rightarrow \{1, \dots, M\}$ дія якого полягає у виборі номера особини батька при породженні чергового нащадка.

Для визначення ГА необхідно задати оператор кросинговеру $Cross: B \times B \rightarrow B \times B$ і оператор мутації $Mut: B \rightarrow B$.

Дія кросинговеру $(\chi', \tau') = \text{Cross}(\chi, \tau)$ полягає у виборі випадковим чином деякої позиції j , рівномірно розподіленої від 1 до $N-1$, після чого результат формується у вигляді

$$\chi' = (\chi_1, \chi_2, \dots, \chi_j, \tau_{j+1}, \dots, \tau_N), \tau' = (\tau_1, \tau_2, \dots, \tau_j, \chi_{j+1}, \dots, \chi_N). \quad (1.6)$$

Вплив кросинговеру регулюють за допомогою ймовірності P_{Cross} спрацьовування цього оператора (в іншому випадку все залишається без змін).

Оператор мутації в кожній позиції аргументу із заданою вірогідністю P_{mut} замінює її вміст на випадковий елемент двійкового алфавіту A , обраний відповідно з рівномірним розподілом (в іншому випадку все залишається без змін).

Цільова функція вихідної задачі, замінюється в ГА на функцію придатності генотипу $\Phi(\chi)$, де $\chi \in B$.

Процес роботи алгоритму є послідовна зміна поколінь, на кожному кроці якої популяція P_{t+1} наповнюється парами нащадків від особин популяції P_t за формулою:

$$(\chi^{t+1}_k, \chi^{t+1}_{k+1}) = \text{Mut}(\text{Cross}(\chi^t_{\text{Select}(P^t)}, \chi^t_{\text{Select}(P^t)})) \quad (1.7)$$

де $(\chi^{t+1}_k, \chi^{t+1}_{k+1})$ – особини з найменшою придатністю популяції P . Тобто індивіди витягуються попарно з P^t і після кросинговеру і мутації поміщаються в P^{t+1} . Зміна ймовірностей мутації і кросинговеру дозволяє регулювати роботу ГА і налаштовувати його на конкретні завдання.

Збільшення швидкодії алгоритму. Недоліком даного методу стиснення є дуже великий обсяг обчислень для кодування зображень. Для програм реалізують даний метод, застосовуються як звичайні методи збільшення швидкості програм: цілочисельна арифметика, асемблерна реалізація та інші,

так і деякі специфічні прийоми: використання афінних перетворень тільки одного типу, різні варіанти розбиття зображення на області R_i .

1.5 Вибір і обґрунтування аналогу

В якості аналога до розробленого алгоритму стиснення зображень на основі дискретних ортогональних функцій обрано метод стиснення зображень JPEG. JPEG – є методом стиснення, що дозволяє стискати дані багатоградаційних зображень з піксельною глибиною від 6 до 24 біт з задовільною швидкістю й ефективністю. Схема JPEG заснована на відкиданні інформації, яку важко помітити візуально. Міра стиснення зображення з фотографічною якістю може становити від 20:1 до 25:1 без помітної втрати якості. Звичайно ж, настільки високий показник стиснення супроводжується відмінністю від оригіналу, але вона настільки незначна, що якість зображення все-таки залишається досить високою. Зображення, що містять великі області одного кольору, стискаються дуже погано.

Недоліками аналогу є:

- при підвищенні ступені стиснення зображення розпадається на окремі квадрати (8 x 8, 16 x 16). Це пов'язано з тим, що відбуваються великі втрати в низьких частотах при квантуванні, і відновити вихідні дані стає неможливо.

- проявляється «ефект Гіббса» – ореоли на границях різких переходів кольорів [17].

1.6 Висновки до розділу

Методи стиснення зображення поділяються на дві великі групи – методи стиснення без втрат та методи стиснення з втратами.

Методи стиснення без втрат дозволяють отримати відновлене зображення ідентичне вихідному. Такі методи стиснення використовують тоді, коли ідентичність вхідного та вихідного зображень є критичною або коли самі шуми зображення є об'єктом дослідження – наприклад, в медицині та науковій діяльності. Методи стиснення без втрат дають в результаті невеликі коефіцієнти стиснення.

Методи стиснення з втратами використовують у випадках, коли зміни зображення, які непомітні людському оку є прийнятними фото- та відеокамери, особливо веб. Такі зміни відновленого зображення дають змогу досягти значно більших коефіцієнтів стиснення, ніж при стисненні без втрат.

Проаналізовано сучасні методи стиснення зображень та виконано вибір аналогу.

2. ЗАСТОСУВАННЯ ОРТОГОНАЛЬНИХ ПЕРЕТВОРЕНЬ ДЛЯ СТИСНЕННЯ ЗОБРАЖЕНЬ

У загальному вигляді процедура лінійного перетворення вектора $X = [x_1, x_2, \dots, x_n]^T$ запишеться в такий спосіб:

$$Y = HX, \quad (2.1)$$

де H – матриця перетворення розміром $N \times N$ елементів.

Для того, щоб перетворення (1.4) було зворотнім, тобто $X = H^{-1}Y$, необхідно і достатньо існування матриці H^{-1} . Серед операторів перетворення особливе місце займають оператори ортогонального перетворення, для яких справедливе умова $H^{-1} = H^T$, де T – знак транспонування.

У задачах кодування зображень вигляд оператора перетворення визначає якість відновленого зображення при заданих коефіцієнтах стиснення. Тому необхідно підібрати таке перетворення, щоб відновлене зображення якнайменше відрізнялося від вихідного відповідно до обраної міри втрат [12].

2.1. Перетворення Адамара

Простим методом перетворення вихідного сигналу є перетворення Адамара. Матриця перетворення H для двох випадкових величин буде мати вигляд:

$$H = \begin{vmatrix} 1 & 1 \\ 1 & -1 \end{vmatrix} \quad (2.2)$$

Якщо з елементів матриці H скласти базисні вектори $e_1 = [h_{11}, h_{12}] = [1, 1]$, $e_2 = [h_{21}, h_{22}] = [1, -1]$, то вони будуть характеризувати поворот ортогональної системи координат на 45° відносно одиничного базису (рисунок 2.1).

Якщо випадкові величини x_1 та x_2 мають кореляційну залежність, то проекція вектора $X = [x_1, x_2]^T$ на базисний вектор e_1 буде, в середньому, більше, ніж проекція цього ж вектора на базисний вектор e_2 . Завдяки цьому інформація буде зосереджувати в першому коефіцієнті перетворення. Другий коефіцієнт слугує для уточнення представлення вектора X у новому базисі.

При збільшенні розмірності простору більша частина інформації буде зосереджена в малому числі коефіцієнтів, які потрібно зберегти з найменшими втратами. Інші коефіцієнти можна або відкинути, або квантувати із більшим кроком.

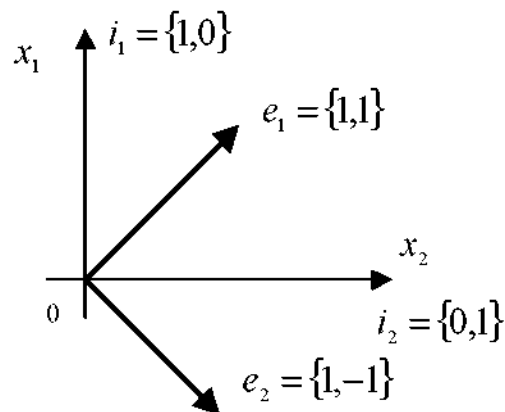


Рисунок 2.1 – Розташування базисних векторів перетворення Адамара щодо одиничного базису

Матрицю Адамара розмірності 4×4 елементів легко побудувати з матриці Адамара H розмірністю 2×2 елементи:

$$H_4 = \begin{vmatrix} H & H \\ H & -H \end{vmatrix} = \begin{vmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{vmatrix} \quad (2.3)$$

Користуючись співвідношенням (2.3), можна побудувати матрицю Адамара будь-якої розмірності $N = 2^n$, де n – будь-яке додатне число.

При аналізі зображень за допомогою даного перетворення спочатку обчислюються коефіцієнти по рядках зображення, а потім по стовпцях. У результаті виходить роздільне перетворення виду:

$$Y = HAH^T, \quad (2.4)$$

де A – $N \times N$ матриця вихідного зображення.

Так як матриця H являє собою оператор ортогонального перетворення, то зворотне перетворення з Y в A запишеться у вигляді:

$$A = H^T Y H \quad (2.5)$$

Перевагою такого перетворення є простота реалізації й низька обчислювальна складність. Розглянуте розкладання буде давати гарні результати для кусково-постійних функцій, тому що базисні функції перетворення Адамара виділяють постійні складові сигналу. У результаті для кусково-постійних сигналів більша частина коефіцієнтів розкладання перетворення Адамара буде близька до нуля, що приведе до зменшення ентропії перетворених даних і збільшенню коефіцієнта стиснення. Однак на практиці такі сигнали зустрічаються досить рідко, тому виникає необхідність визначити таке перетворення, що давало б оптимальне або близьке до оптимального розкладання сигналів, що більш точно характеризує реальні зображення. При цьому під оптимальністю розуміється мінімум середньої

дисперсії помилки відновлення, при відкиданні частини високочастотних коефіцієнтів розкладання [13].

2.2. Перетворення Карунена-Лоева

У випадку ортогонального перетворення з набором N базисних векторів $\phi_i, i = \overline{1, N}$ перетворення з Y в X можна записати як

$$X = A^T Y = \sum_{i=1}^N \phi_i y_i, \quad (2.6)$$

де $A - N \times N$ - матриця ортогонального перетворення.

З усієї множини значень $Y = [y_1, y_2, \dots, y_N]^T$ запам'ятовуються перші M , причому $M < N$. Після відновлення отримаємо оцінку вектора $\hat{X}$:

$$\hat{X} = \sum_{i=1}^M y_i \phi_i + \sum_{i=M+1}^N b_i \phi_i, \quad (2.7)$$

де b_i - деяке константне значення відкинутих коефіцієнтів розкладання

$$y_i, i = \overline{M+1, N}.$$

Помилка при відкиданні $N - M$ коефіцієнтів визначається як

$$\varepsilon = \Delta X = X - \sum_{i=1}^M y_i \phi_i - \sum_{i=M+1}^N b_i \phi_i = \sum_{i=M+1}^N (y_i - b_i) \phi_i, \quad (2.8)$$

а дисперсія цієї помилки

$$M\{\varepsilon^2\} = M\left\{\left(\sum_{i=M+1}^N (y_i - b_i)\phi_i\right)^2\right\} = M\left\{\sum_{i=M+1}^N \sum_{j=M+1}^N (y_i - b_i)(y_j - b_j)^T \phi_i \phi_j\right\} \quad (2.9)$$

Аналіз виразу (2.9) показує, що для мінімізації дисперсії помилки необхідно підібрати відповідні значення b_i та вектори ϕ_i .

Для знаходження оптимальних векторів $\phi_i, i = \overline{1, N}$, перепишемо вираз (2.9) з урахуванням $b_i = \phi_i^T M\{X\} = \phi_i^T \overline{X}$ та $\phi_i^T \phi_j = 0$ при $i \neq j$; у вигляді

$$\begin{aligned} M\{\varepsilon^2\} &= \sum_{i=M+1}^N M\{(y_i - b_i)(y_i - b_i)^T\} = \sum_{i=M+1}^N M\{(\phi_i^T X - \phi_i^T \overline{X})(\phi_i^T X - \phi_i^T \overline{X})^T\} = \\ &= \sum_{i=M+1}^N \phi_i^T M\{(X - \overline{X})(X - \overline{X})^T\} \phi_i = \sum_{i=M+1}^N \phi_i^T R_{xx} \phi_i \end{aligned} \quad (2.10)$$

де $R_{xx} = M\{(X - \overline{X})(X - \overline{X})^T\}$ представляє собою коваріаційну матрицю. Для мінімізації (2.10) застосовується метод множників Лагранжа. В результаті отримаємо

$$R_{xx} \phi_i = \beta_i \phi_i \quad (2.11)$$

Вираз (2.11) визначає вектори ϕ_i та значення β_i як власні вектори і власні значення кореляційної матриці R_{xx} . Таким чином матриця перетворення A , складена з власних векторів кореляційної матриці R_{xx} , дозволяє виконати перетворення Карунена-Лоева (ПКЛ) [14], яке мінімізує середню дисперсію помилки при відновленні сигналу по неповних даних.

Недоліком описаного методу є необхідність апіорних відомостей про кореляційну матрицю R_{xx} вихідного сигналу X , а також відносно висока обчислювальна складність. Так, у випадку застосування роздільного ПКЛ,

для обчислення всіх коефіцієнтів перетворення потрібно N^2 арифметичних операцій, де N кількість відліків вихідного сигналу X .

Таким чином, розглянутий метод може бути застосований у задачах, у яких апріорі відома кореляційна матриця R_{xx} , а також немає великих обмежень на обчислювальні ресурси.

2.3. Дискретне косинусне перетворення

Розглянуті способи перетворення зображень не повністю задовольняють вимогам що звичайно пред'являють до алгоритмів стиснення, таким як низька обчислювальна складність, високі коефіцієнти стиснення, гарна візуальна якість відновленого зображення. Ці вимоги обумовлюють вибір перетворень, які, з одного боку, мають низьку обчислювальну складність, а з іншого, добре описують сигнал, який кодується. Традиційно в якості одного з таких перетворень застосовують розкладання сигналу по косинусним гармонікам ряду Фур'є [15]. Даний метод одержав широке поширення й використовується в алгоритмі стиснення зображень JPEG.

Відомо, що ряд Фур'є для будь-якої безперервної дійсної й симетричної (парної) функції містить тільки дійсні коефіцієнти, що відповідають косинусним членам ряду. У відповідній інтерпретації цей результат можна поширити й на перетворення зображень. Існує простий спосіб одержання симетричних зображень, при якому до вихідного зображення впритул пристроїти його дзеркальні відбиття. При цьому з первісного масиву, що містить $N \times N$ елементів, виходить масив з $2N \times 2N$ елементів.

Елементи симетричного масиву пов'язані з елементами вихідного масиву співвідношеннями:

$$f_s(j, k) = \begin{cases} f(j, k) \text{ при } j \geq 0, k \geq 0, \\ f(-1-j, k) \text{ при } j \leq 0, k \geq 0, \\ f(j, -1-k) \text{ при } j \geq 0, k \leq 0, \\ f(-1-j, -1-k) \text{ при } j \leq 0, k \leq 0. \end{cases} \quad (2.12)$$

Побудований таким чином масив $f_s(j, k)$ симетричний відносно точки $j = -1/2, k = -1/2$. Перетворення Фур'є для випадку, коли початок координат знаходиться в центрі симетрії, запишеться як:

$$F_s(u, v) = \frac{1}{2N} \sum_{j=-N}^{N-1} \sum_{k=-N}^{N-1} f_s(j, k) \exp \left\{ -\frac{2\pi i}{2N} \left[u \left(j + \frac{1}{2} \right) + v \left(k + \frac{1}{2} \right) \right] \right\} \quad (2.13)$$

де $u, v = \overline{-N, N-1}$.

Оскільки масив $f_s(j, k)$ симетричний і складається з дійсних чисел, відношення (2.13) можна записати у вигляді

$$F(u, v) = \frac{2}{N} \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} f(j, k) \cos \left[\frac{\pi}{N} u \left(j + \frac{1}{2} \right) \right] \cos \left[\frac{\pi}{N} v \left(k + \frac{1}{2} \right) \right] \quad (2.14)$$

Так як базисна функція при $u=0, v=0$ представляє собою постійну складову й у два рази більше стосовно всіх інших базисних функцій парного косинусного перетворення, то при виконанні зворотного перетворення необхідно ввести нормуючий коефіцієнт $C(t)$, який визначається виразом

$$C(t) = \begin{cases} 1/2, t = 0 \\ 1, t \neq 0 \text{ \& } t \neq 0 \end{cases} \quad (2.15)$$

В цьому випадку зворотне перетворення запишеться наступним чином

$$f(j, k) = \frac{2}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} C(u)C(v)F(u, v) \cos\left[\frac{\pi}{N}u\left(j + \frac{1}{2}\right)\right] \cos\left[\frac{\pi}{N}v\left(k + \frac{1}{2}\right)\right] \quad (2.16)$$

Перевагою ДКП є те, що базисні функції цього перетворення належать поліному Чебишева й можуть бути обчислені рекурентно. Крім того, це дозволяє синтезувати алгоритми швидкого ортогонального перетворення.

Недоліком ДКП є нерівномірна збіжність на краях інтервалу розкладання N при відкиданні коефіцієнтів, що позначається на візуальній якості відновленого зображення у вигляді артефактів, тобто областей, які не були присутні раніше на вихідному зображенні.

Розглянутий метод одержав широке поширення завдяки наявності швидкого алгоритму обчислення коефіцієнтів. Крім того, він не вимагає апріорної інформації про вхідний сигнал, на відміну від ПКЛ. ДКП рекомендується застосовувати для вхідних сигналів, які можуть бути добре описані косинусними гармоніками, а також мають період рівний періоду розкладання N [16].

2.4. Дискретне вейвлет-перетворення

Дискретне вейвлет-перетворення (ДВП) дає сигналу багатомасштабне представлення засноване на ідеї обережного виявлення границь у вейвлет-теорії. Його переваги – трансляційна інваріантність, може охарактеризувати локальну форму нерегулярних структур розвитком локальних максимумів вейвлетів для звичайних сигналів [24].

Нехай $\Psi(x)$ – вейвлет функція та $f(x) \in L^2(R)$. Дилатацію $\Psi(x)$ з показником 2^j можна визначити як [24]:

$$\psi_{2^j}(x) = \frac{1}{2^j} \psi\left(\frac{x}{2^j}\right). \quad (2.17)$$

Вейвлет перетворення для $f(x)$ у масштабі 2^j та позиції x визначається згорткою [24]:

$$W_{2^j} f(u) = \left(f(x) * \psi_{2^j}(x)\right)(u) = \int_{-\infty}^{\infty} f(x) \frac{1}{\sqrt{2^j}} \psi\left(\frac{x-u}{2^j}\right) dx. \quad (2.18)$$

Це означає, що масштаб для неперервного вейвлет-перетворення дискретний вздовж діадної послідовності $\{2^j\}$, причому j належить цілочисельній множині Z .

Діадне вейвлет-перетворення – це послідовність функцій [24]:

$$Wf = \left(W_{2^j} f(x)\right)_{j \in Z}, \quad (2.19)$$

де W – оператор діадного вейвлет-перетворення.

Нехай функція $\phi(x)$ відповідна згладжувальна функція вейвлет-функції $\Psi(x)$ та S_{2^j} згладжувальний оператор, визначений як згортка:

$$S_{2^j} f(u) = \left(f(x) * \phi_{2^j}(x)\right)(u) = \int_{-\infty}^{\infty} f(x) \frac{1}{2^j} \phi\left(\frac{x-u}{2^j}\right) dx, \quad (2.20)$$

причому $\phi_{2^j}(x)$ визначений як

$$\phi_{2^j}(x) = \frac{1}{\sqrt{2^j}} \phi\left(\frac{x}{2^j}\right), \quad (2.21)$$

тоді, з вищезгаданого визначення, можна відмітити, що більше деталей $f(x)$ видалені S_{2^j} , коли масштаб 2^j стає більшим.

Якщо оригінальний сигнал є дискретною послідовністю $f = (f(n))_{n \in \mathbb{Z}}$ з кінцевою енергією, можна припустити, що фрагменти $a_0(n)$ вхідного дискретного сигналу не зовсім дорівнюють $f(n)$, але локальне середнє f по сусідству $t = n$. Таким чином, $a_0(n)$ може бути записане формулою

$$a_0(n) = \langle f(t), \phi(t-n) \rangle = \int_{-\infty}^{\infty} f(t) \phi(t-n) dt. \quad (2.22)$$

Для кожного $j > 0$ можна відмітити

$$S_{2^j} f = a_j(n) = \langle f(t), \phi_{2^j}(t-n) \rangle = \int_{-\infty}^{\infty} f(t) \phi_{2^j}(t-n) dt. \quad (2.23)$$

Діадне вейвлет-перетворення

$$W_{2^j} f = d_j(n) = Wf(n, 2^j) = \langle f(t), \psi_{2^j}(t-n) \rangle. \quad (2.24)$$

Для кожного масштабу 2^j послідовність дискретних сигналів $\left(S_{2^j} f, (W_{2^j} f)_{1 \leq j \leq J} \right)$ є дискретним діадним вейвлет-перетворенням $f = (f(n))$.

Отже, можна визначити діадне вейвлет-перетворення $(S_2 f, W_2 f)$ як (Sf, Wf) для функції $f(n)$, де Sf виражає низькочастотну інформацію $f(n)$, а Wf – для високочастотної інформації $f(n)$.

2.5 Висновки до розділу

Розглянуто застосування ортогональних дискретних перетворень для стиснення зображень. Оскільки в базовому кодеку JPEG використовується дискретне косинусне перетворення, саме його і обрано для вирішення задачі модифікації алгоритму стиснення зображень JPEG.

3. МОДИФІКАЦІЯ АЛГОРИТМУ JPEG

3.1 Baseline JPEG

Стиснення в базовому алгоритмі JPEG виконується згідно зі схемою рисунку 3.1. Попередньо зображення розбивається на блоки $b_{m,n}$ розміром 8×8 пікселів, де $m \geq 0$ – номер рядків, а $n \geq 0$ – номер стовпця, в яких знаходиться блок. Кожний блок утворює матрицю $X: \{x_{i,j}\}_{i,j=0}^7$, де $x_{i,j}$ – яскравість пікселя з індексами i і j всередині поточного блоку, яка обробляється у 4 етапи.

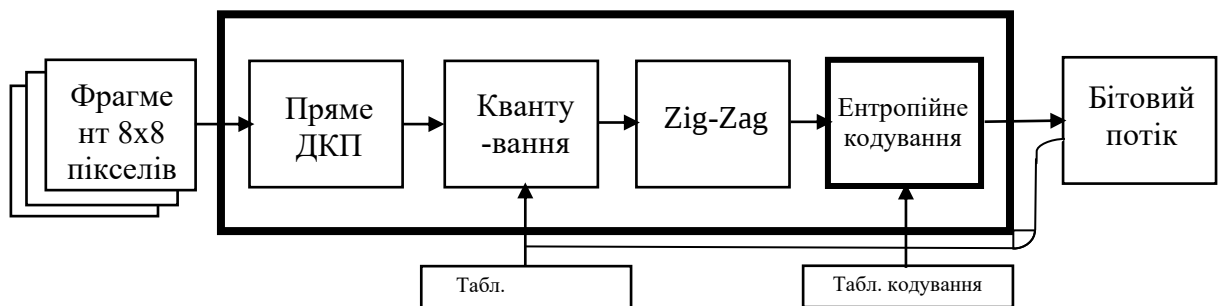


Рисунок 3.1 – Схема стиснення зображення по стандарту JPEG

Етап 1. Пряме ДКП. Блоки обробляються рядково зліва направо від лівого верхнього кута зображення. Всередині кожного блоку виконується дискретне косинусне перетворення (ДКП) та виходить нова матриця Y :

$$y_{ij} = \text{ДКП}(x_{ij}) \quad (3.1)$$

Етап 2. Квантування (огрубіння) коефіцієнтів ДКП. Мета даної процедури полягає в обнуленні або зменшенні високочастотних коефіцієнтів ДКП, що слабо впливають на якість зображення, яке сприймається

людиною. Для цього використовується спеціальна матриця квантування $Q: \{q_{ij}\}_{i,j=0}^8$ загальна для всіх блоків. Матриця Y поелементно ділиться на Q , що утворює нову матрицю $\tilde{Y}$:

$$\tilde{y}_{i,j} = \text{round}\left(\frac{y_{i,j}}{p \cdot q_{i,j}}\right), \quad (3.2)$$

де функція round означає заокруглення до найближчого цілого числа, а p – параметр, що задається користувачем. У baseline JPEG використовується вбудована матриця квантування, а параметр p задається користувачем перед стисненням. Цей параметр формально визначає якість одержуваного зображення.

Етап 3. "Зигзаг" упорядкування. Отримана ціла матриця $\tilde{Y}$ складається з елементів, що спадають зі зростанням кожного індексу. Елементи цієї матриці впорядковуються в одновимірний масив за правилом "зигзаг", як показано на рисунку 3.2. Спочатку виписується кутовий елемент $\tilde{y}_{0,0}$, потім 2 елементи першої побічної діагоналі, потім елементи наступної побічної діагоналі і так далі.

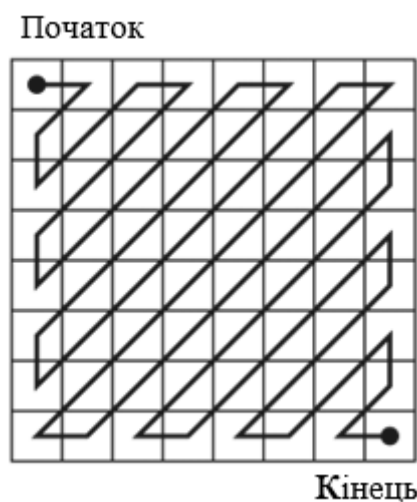


Рисунок 3.2 – Схема «зигзаг» впорядкування

Закон зменшення коефіцієнтів ДКП [6], процедура квантування (3.2) та спеціальна схема впорядкування «зигзаг» дають, як правило, не зростаючий вектор $V = \{\tilde{y}_k\}, \tilde{y}_k \geq \tilde{y}_{k+1}, 0 \leq k \leq 63$

Для елементів $\tilde{y}_0^{m,n}$ кожного блоку $b_{m,n}$, виробляється диференціальна імпульсно-кодова модуляція (ДІКМ). У поточному блоці $b_{m,n}$, замість істинного значення $\tilde{y}_0^{m,n}$ зберігається різниця з $\tilde{y}_0$ попереднього блоку. Для сусідніх блоків всередині рядка з номером m обчислюється різниця

$$\Delta \tilde{y}_0^{m,n} = \tilde{y}_0^{m,n} - \tilde{y}_0^{m,n-1}, m \geq 0, n > 0, \quad (3.3)$$

а для блоків на початку рядка попереднім вважається початковий блок попереднього рядка

$$\Delta \tilde{y}_0^{m,n} = \tilde{y}_0^{m,n} - \tilde{y}_0^{m-1,n}, m \geq 0, n > 0 \quad (3.4)$$

а перший елемент першого блоку $\tilde{y}_0^{m,n}$ залишається незмінним. Різниця середніх яркостей сусідніх блоків слабо змінюється у природних зображеннях, тому зберігати абсолютні значення $\tilde{y}_0$ не вигідно. Таким чином ДІКМ забезпечує більш економне представлення елементів $\tilde{y}_0$.

Етап 4. Ентропійне кодування. Заключний етап – це кодування векторів $V_{m,n}$ кожного блоку $b_{m,n}$, в бітові послідовності та їх запис у вихідний потік. Блоки обробляються окремо один від одного, але при цьому вважається, що вони підпорядковуються одній статистичній моделі. Тому для кодування використовуються загальні кодові таблиці. Важливо зазначити, що кодуються не самі елементи векторів, а їх особливий запис.

Перший елемент вектору $\Delta \tilde{y}_0$ кодується за принципом VLI (variable length integer) [7]. Таким чином виходить мінімальний бітовий код

$$\Delta \tilde{y}_0 \text{ VLI} \rightarrow R_0 \quad (3.5)$$

який не є префіксним, тобто може бути початком будь-якого іншого коду. Тому додатково записується інформація про довжину цього коду

$$L_0 = \text{length}(R_0), \quad (3.6)$$

де length – функція, що підраховує кількість біт у коді R_0 . Ця інформація необхідна для коректного зчитування кодів із потоку даних на етапі декодування. Число L_0 окремо представляється префіксним кодом

$$L_0 \text{ code table} \rightarrow L_0^{H1}, \quad (3.7)$$

який береться із спеціально підготовленої таблиці кодових слів $H1$. Серед елементів, що залишилися $y_k, 1 \leq k \leq 63$ кодуються тільки ненульові також як і в (3.5)

$$\tilde{y}_k \text{ VLI} \rightarrow R_k \quad (3.8)$$

а інформація про кількість попередніх нульових елементів записується в байт-пару $(Z_k, \text{length}(R_k))$

$$P_k = Z_k \ll 4 + \text{length}(R_k) \quad (3.9)$$

де Z_k – число нульових елементів попередніх y_k , length – функція з (3.6), операція $\ll$ – бітове зрушення у бік старших розрядів. Такі пари кодуються префіксними кодами, аналогічно (3.7)

$$P_k \text{ code table} \rightarrow P_k^{H2} \quad (3.10)$$

За допомогою іншої таблиці кодових слів $H2$.

Зауважимо, що на практиці Z_k може бути більше 15, отже, 4-х біт може бути недостатньо для запису цього числа. Тому послідовності з 15 нульових

елементів кодуються спеціальною парою (15,0). Якщо після якогось y_k всі елементи, що залишилися $y_{k+1} \dots y_{63}$ рівні нулю, то вони кодуються іншою спеціальною парою (0,0) – ознакою кінця поточного блоку $b_{m,n}$. У вихідний потік записуються коди

$$b_{m,n} \text{ entropy coding} \rightarrow (L_0^{H1} R_0) (P_{k_1}^{H2} R_{k_1}) \dots (P_{k_d}^{H2} R_{k_d}) \quad (3.11)$$

де $k_1, \dots, k_d$ – індекси ненульових y_k .

У baseline JPEG на етапі ентропійного кодування використовується алгоритм Хаффмана. Існують інші альтернативи кодування, наприклад арифметичне кодування, але вони виявляються більш ресурсоемними, хоча в деяких випадках дають більший стиск. Можливі два варіанти кодування: з використанням фіксованих кодових таблиць або оптимальних. Кодові таблиці H1 і H2 з (3.7) та (3.10) формуються з урахуванням статистичного аналізу даних, що стискають.

Як показує практика, такий аналіз може займати до половини часу всього кодування [8]. Тому користуються вбудованими таблицями Хаффмана, які оптимізовані для конкретного рівня якості p_z (2), що значно заощаджує часові ресурси. Однак для інших значень p ці таблиці виявляються неоптимальними і, як наслідок, виходить не найкраще стиснення.

В стандарті JPEG пропонується використовувати вбудовані таблиці Хаффмана для (3.7) та (3.10). Приклад цих таблиць будується модифікація JPEG-MOD.

3.2 Алгоритм JPEG-MOD

Нам вдалося знайти спосіб, як можна покращити характеристики базового кодека JPEG, який використовує вбудовані таблиці Хаффмана. Якщо користувач вимагає високої якості від стисненого зображення, то квантування (2) буде слабким і більшість елементів вектора $V_{m,n}$ у всіх блоках $b_{m,n}$ будуть m ненульовими. Майже завжди кілька перших елементів вектора будуть ненульовими. Тому стандартна схема кодування (3.8) - (3.10) буде надлишковою.

Пропонується змінити схему кодування так, щоб кілька перших елементів $\tilde{y}_k \neq 0, 1 \leq k < t \leq 63$ кодувалась так само, як і $\Delta\tilde{y}_0$ згідно з (3.5)-(3.7). В цьому випадку у вихідний потік, за аналогією з (3.11), записуються такі коди

$$b_m \text{ entropy coding} \rightarrow (L_0^{H1} R_0)(L_1^{H1} R_1) \dots (L_t^{H1} R_t)(P_{k_1}^{H2} R_{k_1}) \dots (P_{k_d}^{H2} R_{k_d}) \quad (3.12)$$

де $0 < t < k_1$ – індекси перших ненульових елементів вектора $V_{m,n}$.

Перші ненульові елементи $\tilde{y}_k$ утворюють початковий трикутник у вихідній матриці $\tilde{Y}$, (рисунок 3.2). Така зміна дозволяє заощаджувати до кількох біт на блок. При сильному квантуванні (при низькій якості отриманого зображення) зустрічаються блоки, в яких вектор $V_{m,n}$ складається тільки з нульових елементів, може бути винятком $\Delta\tilde{y}_0$. У цьому випадку у вихідний потік записується код Хаффмана для спеціальної пари (0,0) – ознаки кінця блоку. Стандартна таблиця Хаффмана кодує цю пару чотирма бітами [1010]. Якщо $\forall k, 1 \leq h \leq 63: \tilde{y}_h = 0$, тоді в вихідний потік записують коди

$$b_{m,n} \text{ entropy coding} \rightarrow (L_0^{H1} R_0)[1010], \quad (3.13)$$

а в іншому випадку, коли зустрічається кілька ненульових $\tilde{y}_k$ записуються

$$b_{m,n} \text{ entropy coding} \rightarrow (L_0^{H1} R_0) (P_{k_1}^{H2} R_{k_1}) \dots (P_{k_d}^{H2} R_{k_d}) [1010] \quad (3.14)$$

де $k_1, \dots, k_d$ – індекси ненульових $\tilde{y}_k$.

Пропонується використовувати лише 1 біт замість 4 для позначення кінця блоку. Якщо $\tilde{y}_k = 0, 1 \leq k \leq 63$, то записується «0»:

$$b_{m,n} \text{ entropy coding} \rightarrow (L_0^{H1} R_0) [0] \quad (3.15)$$

в іншому випадку – «1», а елементи, що залишилися, кодуються за стандартною

$$b_{m,n} \text{ entropy coding} \rightarrow (L_0^{H1} R_0) [1] (P_{k_1}^{H2} R_{k_1}) \dots (P_{k_d}^{H2} R_{k_d}) [1010], \quad (3.16)$$

Схема кодування (3.15)-(3.16) дозволяє заощаджувати до 3 біт на блок порівняно з (3.13)-(3.14). Обидві пропозиції (3.12) та (3.15) - (3.16) доповнюють одна одну та є ефективною модифікацією базового кодека JPEG.

3.3 Методика порівняння

Для проведення розрахунків було обрано загальнодоступну реалізацію алгоритму JPEG [9] мовою C++. Описані вище модифікації було реалізовано у цьому коді. Модифікація JPEG-MOD торкається лише етапу ентропійного кодування кодека baseline JPEG, що не впливає на якість стисненого зображення. Тому зображення, які отримують кодеками baseline JPEG і JPEG-MOD, ідентичні за якістю, але відрізняються ступенем стиснення.

Інші режими роботи кодека JPEG (progressive, optimized, arithmetic, lossless) є надбудовами над baseline JPEG. Тому порівняння цих режимів роботи кодека з JPEG-MOD даватиме такий самий результат, що і для baseline JPEG, але з урахуванням виграшу в стиску, який був отриманий в ході експериментів. Порівняння проводилося на 24 різних чорно-білих зображення з бази KODAK [10]. Кожне зображення бази представлене у чотирьох розмірах: 0.4, 0.6, 1.6 та 6.3 мегапікселя. Для оцінки якості стислих зображень використовувалися спеціальні метрики адекватні людському сприйняттю. Відомо близько 20 альтернативних математичних критеріїв оцінки якості спотвореного зображення, зокрема широко поширений критерій PSNR, який не враховує особливості людського зору. З результатів попередніх робіт [11] слід, що з оцінки якості зображення найкращим є математичний критерій VIF [12]. У ході експерименту проводилося кодування тестових зображень кодеками baseline JPEG та JPEG-MOD. Параметр якості p змінювався у всьому діапазоні допустимих значень. Для кожного тестового зображення обчислювалися ступінь стиснення як відношення розміру вихідного файлу до стиснутого, і якість за критерієм VIF.

3.4 Аналіз результатів

Отримані основні результати наведено на рисунку 3.3. По осі абсцис відкладено якість зображення за критерієм VIF у лінійному масштабі. Значення критерію VIF відповідають умовним суб'єктивними оцінками якості: погане $[0; 0.4)$, посереднє $[0.4; 0.6)$, добрий $[0.6; 0.8)$ та відмінне $[0.8; 1.0]$. На рисунку 3.4 наведено приклади стиснених зображень кожного діапазону якості зображень. На рисунку 3.3 ці діапазони відокремлені вертикальними пунктирами. По осі ординат відкладено виграш у стисненні R^* алгоритму JPEG-MOD у порівнянні зі стиском R стандартного JPEG у

відсотках. Кожна крива відповідає середньому виграшу у стисненні зображень однакового розміру, вказаного поруч із цією кривою.

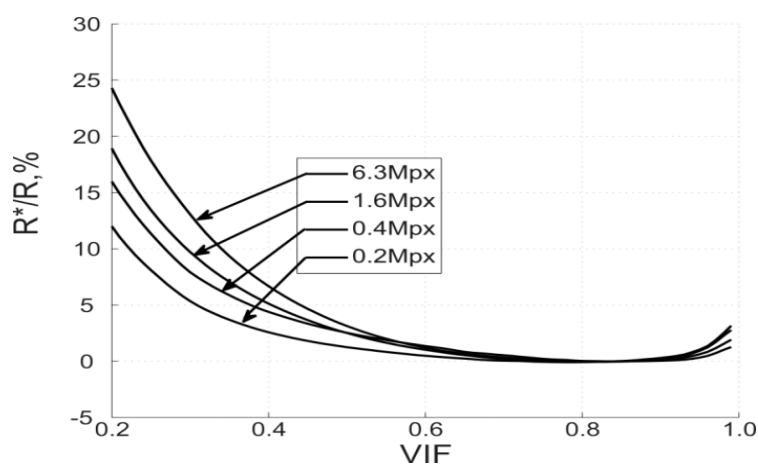


Рисунок 3.3 – Середній виграш у стисненні JPEG-MOD за 24 зображеннями; значення біля кривих – розмір зображення у мегапікселях



Рисунок 3.4 – Зображення №21 з бази KODAK після компресії для різних рівнів якості VIF (рядково зліва направо: 0.11, 0.21, 0.31, 0.46, 0.68 та 0.99)

Результат стиснення зображення з рисунку 3.4 представлений у таблиці 3.1. Для кожного рівня якості наведено бітові витрати на піксель при використанні кодека JPEG та JPEG-MOD. В останньому рядку таблиці наведено процентний приріст у стисканні запропонованої модифікації проти стандартного кодека. Виграш змінюється від ~0.4% до 20.4% в залежності від рівня якості стисненого зображення.

Таблиця 3.1 – Бітові витрати на піксель зображень з рисунку 3.4

Якість VIF	0.11	0.21	0.31	0.46	0.68	0.99
Кодек						
Baseline JPEG	0.159	0.218	0.349	0.541	0.939	3.46
JPEG-MOD	0.132	0.197	0.332	0.529	0.935	3.30
Виграш, %	20.4	10.6	5.12	2.26	0.427	4.85

Аналіз рисунку 3.3 та таблиці 3.1 дозволяє зробити такі висновки:

- У разі застосування запропонованих модифікацій виграш у стиску є у всьому діапазоні якості. За хорошої якості він складає ~0.5%, при високому до ~4.5%, а при середньому і поганому зростає до ~20%, в залежності від розміру зображення.
- При роботі із зображеннями розміром від 0.2 до 6.3 Мп виграш у стиску змінюється відповідно: від 2.0 до 4.0% у діапазоні середньої якості, від 5.0 до 20.0% у діапазоні поганої якості.
- З використанням інших таблиць Хаффмана, оптимізованих іншого рівня якості, форма кривих на рисунку 3 збережеться. При цьому мінімум зміститься в область якості, для якої оптимізація.

4 ЕКОНОМІЧНИЙ РОЗДІЛ

4.1 Технологічний аудит розробленого модифікованого методу стиснення зображень на основі алгоритму JPEG в системах передачі інформації

Як було зазначено раніше, одним з важливих факторів постійного відвідування користувачами різних сайтів є швидкість їх завантаження. Зображення, що їх бачить користувач, повинні бути достатньо високої якості, але в той же час мати невеликий розмір файлу. Оскільки останнім часом в практичній діяльності користувачів все частіше використовується мобільний інтернет, який має невелику швидкість, то вимога до зменшення розміру файлу стає все більш актуальною. Ця вимога посилюється також у зв'язку з тим, що розвиток сучасних інформаційних систем і мереж привів до широкого використання цифрових зображень. Не випадково багато користувачів, які мають відношення до отримання, обробки, зберігання і передачі інформації, значною мірою орієнтуються на розвиток систем, в яких інформація представлена у вигляді зображень. Тому стиснення зображень стає важливим фактором для підвищення ефективності використання комунікаційних й інформаційно-обчислювальних ресурсів різноманітних систем.

У зв'язку з цим, метою виконаної нами магістерської кваліфікаційної роботи було пропонування методу та розроблення алгоритму стиснення цифрових зображень, який дозволив би підвищити ступінь стискання зображень та зберегти їх прийнятну якість.

В результаті виконання нашої роботи було запропоновано модифікований метод стиснення зображень JPEG-MOD, який дає вигоду у стисненні зображень до 20% порівняно з базовим кодеком JPEG (залежно від необхідної якості та вихідного розміру зображення). Запропонований модифікований метод дає вигоду у стисненні зображень як з дрібними

деталлями, так і фонових зображень. Причому модифікований метод стиснення зображень JPEG-MOD не збільшує трудомісткість базового алгоритму JPEG при компресії та декомпресії зображень.

Для встановлення комерційного потенціалу розробленого нами модифікованого методу стиснення зображень було запрошено 3-х відомих експертів – кандидата технічних наук, професора Папінова В.М., кандидата технічних наук, доцента Кабачого В.В. та кандидатку технічних наук, доценткиню Богач І.В.

Встановлення комерційного потенціалу розробленого нами модифікованого методу стиснення зображень здійснювалося за загально визнаними критеріями, які наведені в таблиці 4.1.

Таблиця 4.1 – Рекомендовані критерії оцінювання комерційного потенціалу будь-якої розробки і їх бальна оцінка

Критерії оцінювання та бали (за 5-ти бальною шкалою)					
	0	1	2	3	4
Технічна здійсненність концепції:					
1	Достовірність концепції не підтверджена	Концепція підтверджена експертними висновками	Концепція підтверджена розрахунками	Концепція перевірена на практиці	Перевірено роботоздатність продукту в реальних умовах
Ринкові переваги (недоліки):					
2	Багато аналогів на малому ринку	Мало аналогів на малому ринку	Кілька аналогів на великому ринку	Один аналог на великому ринку	Продукт не має аналогів на великому ринку
3	Ціна продукту значно вища за ціни аналогів	Ціна продукту дещо вища за ціни аналогів	Ціна продукту приблизно дорівнює цінам аналогів	Ціна продукту дещо нижче за ціни аналогів	Ціна продукту значно нижче за ціни аналогів
4	Технічні та споживчі властивості продукту значно гірші, ніж в аналогів	Технічні та споживчі властивості продукту трохи гірші, ніж в аналогів	Технічні та споживчі властивості продукту на рівні аналогів	Технічні та споживчі властивості продукту трохи кращі, ніж в аналогів	Технічні та споживчі властивості продукту значно кращі, ніж в аналогів

Продовження таблиці 4.1

Критерії оцінювання та бали (за 5-ти бальною шкалою)					
	0	1	2	3	4
Ринкові перспективи					
5	Експлуатаційні витрати значно вищі, ніж в аналогів	Експлуатаційні витрати дещо вищі, ніж в аналогів	Експлуатаційні витрати на рівні експлуатаційних витрат аналогів	Експлуатаційні витрати трохи нижчі, ніж в аналогів	Експлуатаційні витрати значно нижчі, ніж в аналогів
6	Ринок малий і не має позитивної динаміки	Ринок малий, але має позитивну динаміку	Середній ринок з позитивною динамікою	Великий стабільний ринок	Великий ринок з позитивною динамікою
7	Активна конкуренція великих компаній на ринку	Активна конкуренція	Помірна конкуренція	Незначна конкуренція	Конкурентів немає
Практична здійсненність					
8	Відсутні фахівці як з технічної, так і з комерційної реалізації ідеї	Необхідно наймати фахівців або витратити значні кошти та час на навчання наявних фахівців	Необхідне незначне навчання фахівців та збільшення їх штату	Необхідне незначне навчання фахівців	Є фахівці з питань як з технічної, так і з комерційної реалізації ідеї
9	Потрібні значні фінансові ресурси, які відсутні. Джерела фінансування ідеї відсутні	Потрібні незначні фінансові ресурси. Джерела фінансування відсутні	Потрібні значні фінансові ресурси. Джерела фінансування є	Потрібні незначні фінансові ресурси. Джерела фінансування є	Не потребує додаткового фінансування

Продовження таблиці 4.1

Критерії оцінювання та бали (за 5-ти бальною шкалою)					
	0	1	2	3	4
10	Необхідна розробка нових матеріалів	Потрібні матеріали, що використовуються у військово-промисловому комплексі	Потрібні дорогі матеріали	Потрібні досяжні та дешеві матеріали	Всі матеріали для реалізації ідеї відомі та давно використовуються у виробництві
11	Термін реалізації ідеї більший за 10 років	Термін реалізації ідеї більший за 5 років. Термін окупності інвестицій більше 10-ти років	Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більше 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій від 3-х до 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій менше 3-х років
12	Необхідна розробка регламентних документів та отримання великої кількості дозвільних документів на виробництво та реалізацію продукту	Необхідно отримання великої кількості дозвільних документів на виробництво та реалізацію продукту, що вимагає значних коштів та часу	Процедура отримання дозвільних документів для виробництва та реалізації продукту вимагає незначних коштів та часу	Необхідно тільки повідомлення відповідним органам про виробництво та реалізацію продукту	Відсутні будь-які регламентні обмеження на виробництво та реалізацію продукту

Запрошені експерти оцінили розроблений нами модифікований метод стиснення зображень таким чином (див. таблицю 4.2):

Таблиця 4.2 – Результати технологічного аудиту розробленого нами модифікованого методу стиснення зображень (за шкалою оцінювання 0-1-2-3-4)

Критерії	Прізвище, ініціали експертів		
	Кабачій В.В.	Папінов В.М.	Богач І.В.
	Бали, що їх виставили експерти:		
1	3	3	3
2	3	3	3
3	3	3	3
4	3	2	3
5	2	3	3
6	2	2	3
7	3	3	3
8	2	3	3
9	3	3	3
10	3	3	3
11	2	3	3
12	3	3	3
Сума балів	СБ ₁ = 32	СБ ₂ = 34	СБ ₃ = 36
Середньоарифметична сума балів $\overline{СБ}$	$\overline{СБ} = \frac{\sum_{i=1}^3 СБ_i}{3} = \frac{32 + 34 + 36}{3} = \frac{102}{3} = 34,00$		

Встановлення комерційного потенціалу розробленого нами модифікованого методу стиснення зображень будемо здійснювати на основі рекомендацій, наведених в таблиці 4.3 [26].

Таблиця 4.3 – Рівні комерційного потенціалу будь-якої наукової розробки

Середньоарифметична сума балів $\overline{СБ}$, розрахована на основі висновків Експертів	Рівень комерційного потенціалу розробки
0 – 10	Низький
11 – 20	Нижче середнього
21 – 30	Середній
31 – 40	Вище середнього
41 – 48	Високий

Оскільки середньоарифметична сума балів, що їх виставили експерти, складає 34 бали, то це свідчить про те, що розроблений нами модифікований метод стиснення зображень має рівень комерційного потенціалу, який вважається «вище середнього».

Це пояснюється тим, що розроблений нами модифікований метод стиснення зображень має значно кращі експлуатаційні показники і може бути використаний для пришвидшення завантаження сайтів у мережі Інтернет.

4.2 Розрахунок витрат на розроблення модифікованого методу стиснення зображень

При розробленні модифікованого методу стиснення зображень були зроблені певні витрати. Зокрема:

А) Основна заробітна плата Z_o розробників, яка визначається за формулою:

$$Z_o = \frac{M}{T_p} \cdot t \text{ грн,} \quad (4.1)$$

де M – місячний посадовий оклад розробника, грн; приймемо, що

$M = (8000 \dots 35000)$ грн/місяць;

T_p – число робочих днів в місяці; приймемо $T_p = 20$ днів;

t – число днів роботи розробників.

Зроблені розрахунки зведемо до таблиці 4.4

Таблиця 4.4 – Основна заробітна плата розробників

Найменування посади виконавця	Місячний посадовий оклад, грн	Оплата за робочий день, грн	Число днів роботи	Витрати на оплату праці, грн
1. Науковий керівник магістерської роботи	25000	1250	20 годин	≈ 4167
2. Магістрант-студент-виконавець	Беремо мінімальну оплату 8000 грн	400	82	≈ 32800
3. Консультант з економічної частини	20000	1000	1,5 години	≈ 250 (при 6-годинному робочому дні)
Загалом				$3_o = 37217$ грн

Б) Додаткова заробітна плата $З_d$ розробників розраховується як (10...12)% від величини їх основної заробітної плати, тобто:

$$З_d = \alpha \cdot З_o = (0,1...0,12) \cdot З_o. \quad (4.2)$$

Приймемо, що $\alpha = 0,1$. Тоді для нашого випадку отримаємо:

$$З_d = 0,1 \times 37217 = 3721,7 \approx 3722 \text{ грн.}$$

В) Нарахування на заробітну плату НЗП_{зп} розробників (дослідників) розраховуються за формулою:

$$\text{НЗП}_{\text{зп}} = (З_o + З_d) \cdot \frac{\beta}{100}, \quad (4.3)$$

де β – ставка обов'язкового єдиного внеску на державне соціальне страхування, %. $\beta = 22\%$. Тоді:

$$H3H_{3П} = (37217 + 3722) \times 0,22 = 9006,58 \approx 9007 \text{ грн.}$$

Г) Амортизація основних засобів А, які використовувались під час виконання цієї роботи:

$$A = \frac{Ц \cdot H_a}{100} \cdot \frac{T}{12} \text{ грн,} \quad (4.4)$$

де Ц – загальна балансова вартість основних засобів, грн;

H_a – річна норма амортизаційних відрахувань. Для нашого випадку можна прийняти, що $H_a = (2,5...25)\%$;

T – термін використання основних засобів, місяці.

Зроблені розрахунки зведено в таблицю 5.5.

Таблиця 4.5 – Розрахунок амортизаційних відрахувань

Найменування обладнання, приміщень тощо	Балансова вартість, грн.	Норма амортизації, %	Термін використання, місяці	Величина амортизаційних відрахувань, грн
1. Комп'ютерна техніка, обладнання, принтери тощо	58000	25	3,5 (при 80% використанні i)	3383,33 $\approx$ 3384
2. Приміщення університету, кафедри	24000	2,5	3,5 при 40% використанні i	70
Всього				A = 3454 грн

Д) Витрати на матеріали М розраховуються за формулою:

$$M = \sum_1^n H_i \cdot Ц_i \cdot K_i - \sum_1^n B_i \cdot Ц_b \text{ грн.,} \quad (4.5)$$

де H_i – витрати матеріалу i -го найменування, кг; Π_i – вартість матеріалу i -го найменування; K_i – коефіцієнт транспортних витрат, $K_i = (1,1 \dots 1,15)$; B_i – маса відходів матеріалу i -го найменування; Π_b – ціна відходів матеріалу i -го найменування; n – кількість видів матеріалів.

Е). Витрати на комплектуючі K розраховуються за формулою:

$$K = \sum_1^n H_i \cdot \Pi_i \cdot K_i \text{ грн}, \quad (4.6)$$

де H_i – кількість комплектуючих i -го виду, шт.; Π_i – ціна комплектуючих i -го виду; K_i – коефіцієнт транспортних витрат, $K_i = (1,1 \dots 1,15)$; n – кількість видів комплектуючих.

Під час виконання магістерської кваліфікаційної роботи загальні витрати на матеріали та комплектуючі склали приблизно 1600 грн.

Ж). Витрати на силову електроенергію B_e розраховуються за формулою:

$$B_e = \frac{B \cdot \Pi \cdot \Phi \cdot K_{\Pi}}{K_d}, \quad (4.7)$$

де B – вартість 1 кВт-год. електроенергії, в 2024 р. $B \approx 4,5$ грн/кВт;

Π – установлена потужність обладнання, кВт; $\Pi = 1,0$ кВт;

Φ – фактична кількість годин роботи обладнання, годин.

Прийmemo, що $\Phi = 310$ годин;

K_{Π} – коефіцієнт використання потужності; $K_{\Pi} < 1 = 0,86$.

K_d – коефіцієнт корисної дії, $K_d = 0,79$.

Тоді витрати на силову електроенергію будуть дорівнювати:

$$B_e = \frac{B \cdot \Pi \cdot \Phi \cdot K_{\Pi}}{K_d} = \frac{4,5 \cdot 1,0 \cdot 310 \cdot 0,86}{0,79} = 1518,51 \approx 1519 \text{ грн.}$$

И) Інші витрати $V_{\text{інш}}$ можна прийняти як $(50...300)\%$ від основної заробітної плати розробників, тобто:

$$V_{\text{інш}} = (0,5....3) \times Z_o. \quad (4.8)$$

Для нашого випадку отримаємо:

$$V_{\text{інш}} = 0,9 \times 37217 = 33495,3 \approx 33496 \text{ грн.}$$

К) Сума всіх попередніх статей витрат складає витрати на виконання роботи безпосередньо розробником-магістрантом – V .

$$V = 37217 + 3722 + 9007 + 3454 + 1600 + 1519 + 33496 = 90015 \text{ грн.}$$

Л) Загальні витрати на розробку модифікованого методу стиснення зображень $V_{\text{заг}}$ розраховуються за формулою:

$$V_{\text{заг}} = \frac{V}{\beta}, \quad (4.9)$$

де β – коефіцієнт, який характеризує етап (стадію) виконання цієї роботи.

Можна прийняти, що, $\beta \approx 0,95$ [26].

$$\text{Тоді: } V_{\text{заг}} = \frac{90015}{0,95} = 94752,63 \text{ грн або приблизно 95 тисяч грн.}$$

Тобто прогнозовані загальні витрати на розробку модифікованого методу стиснення зображень становлять приблизно 95 тисяч грн.

4.3 Розрахунок економічного ефекту від можливої комерціалізації розробки

Економічний ефект від впровадження та можливої комерціалізації розробленого нами модифікованого методу стиснення зображень пояснюється його значно кращими функціональними можливостями. Тому нашу розробку можна реалізовувати на ринку дещо дорожче, ніж аналогічні за функціями розробки. Так, якщо подібна за функціями розробка коштує на ринку приблизно 95 тисяч грн, то нашу розробку можна буде реалізовувати на ринку приблизно за 115 тисяч грн або на 20 тисяч грн дорожче.

Аналіз місткості ринку показує, що на сьогодні в Україні кількість потенційних споживачів розробленого нами модифікованого методу стиснення зображень може становити приблизно 100 осіб. Це, насамперед, веб-розробники, дизайнери, виробники цифрових камер і смартфонів, інтернет-провайдери, мобільні оператори, соціальні мережі та платформи для обміну зображеннями, хмарні сервіси зберігання даних; бізнес-користувачі; медіа та розважальні платформи; освітні установи та інші потенційні споживачі. Тому можна очікувати зростання попиту на нашу розробку принаймні протягом 3-х років після її впровадження.

Тобто, якщо наша розробка буде впроваджена з 1 січня 2025 року, то її результати будуть виявлятися протягом 2025-го, 2026-го та 2027-го років.

Прогноз зростання попиту на нашу розробку складає по роках:

- а) 2025 р. – приблизно +10 шт. до базового року;
- б) 2026 р. – +15 шт. до базового року;
- в) 2027 р. – +20 шт. до базового року.

Можливе збільшення чистого прибутку $\Delta\Pi_i$, що його може отримати потенційний інвестор від комерціалізації нашої розробки, тобто виведення нашої розробки на ринок, становитиме:

$$\Delta\Pi_i = \sum_{i=1}^n (\Delta\Pi_o \cdot N + \Pi_o \cdot \Delta N)_i \cdot \lambda \cdot \rho \cdot \left(1 - \frac{\nu}{100}\right), \quad (4.10)$$

де $\Delta \Pi_0$ – покращення основного якісного показника від впровадження результатів нашої розробки у цьому році. Для нашого випадку це є збільшення ціни реалізації нашої розробки $\Delta \Pi_0 = (115 - 95) = + 20$ тисяч грн;

N – основний кількісний показник, який визначає обсяг діяльності у році до впровадження результатів розробки; $N = 100$ шт.;

ΔN – покращення основного кількісного показника від впровадження результатів розробки. Таке покращення становитиме по роках, відповідно: у 2025 році – + 10 шт., у 2026 році +15 шт, та у 2027 році + 20 шт. (відносно базового 2024 року);

Π_0 – основний якісний показник (тобто ціна), який визначає обсяг діяльності у році після впровадження результатів розробки, грн; $\Pi_0 = 115$ тисяч грн;

n – кількість років, протягом яких очікується отримання позитивних результатів від впровадження розробки; для нашого випадку $n = 3$;

λ – коефіцієнт, який враховує сплату податку на додану вартість; $\lambda = 0,8333$;

ρ – коефіцієнт, який враховує рентабельність продукту. Рекомендується приймати $\rho = (0,2 \dots 0,5)$; візьмемо $\rho = 0,45$;

v – ставка податку на прибуток. У 2024 $v = 18\%$. У 2025 році і в наступних роках також очікуємо ставку $v = 18\%$.

Тоді можливе зростання чистого прибутку $\Delta \Pi_1$ для потенційного інвестора протягом першого року від можливого впровадження (комерціалізації) нашої розробки (2025 р.) складе:

$$\Delta \Pi_1 = [20 \cdot 100 + 115 \cdot 10] \cdot 0,8333 \cdot 0,45 \cdot \left(1 - \frac{18}{100}\right) \approx 969 \text{ тис. грн.}$$

Можливе зростання чистого прибутку $\Delta \Pi_2$ для потенційного інвестора від можливого впровадження (комерціалізації) нашої розробки протягом другого (2026 р.) року складе:

$$\Delta\Pi_2 = [20 \cdot 100 + 115 \cdot 15] \cdot 0,8333 \cdot 0,45 \cdot \left(1 - \frac{18}{100}\right) \approx 1146 \text{ тис. грн.}$$

Можливе зростання чистого прибутку $\Delta\Pi_3$ для потенційного інвестора від можливого впровадження (комерціалізації) нашої розробки протягом третього (2027 р.) року становитиме:

$$\Delta\Pi_3 = [20 \cdot 100 + 115 \cdot 20] \cdot 0,8333 \cdot 0,45 \cdot \left(1 - \frac{18}{100}\right) \approx 1322 \text{ тис. грн.}$$

Приведена вартість зростання всіх чистих прибутків потенційного інвестора від можливого впровадження нашої розробки становитиме:

$$\text{ПП} = \sum_{i=1}^t \frac{\Delta\Pi_i}{(1+\tau)^t}, \quad (4.11)$$

де $\Delta\Pi_i$ – збільшення чистого прибутку у кожному із років, протягом яких виявляються результати виконаної та впровадженої роботи, грн;

t – період часу, протягом якого виявляються результати впровадженої роботи, роки. Для нашого випадку $t = 3$ роки;

τ – ставка дисконтування. Прийmemo $\tau = 0,10$ (10%);

t – період часу від моменту початку розроблення модифікованого методу стиснення зображень до моменту отримання можливих чистих прибутків потенційним інвестором.

Тоді приведена вартість зростання всіх можливих чистих прибутків ПП, що їх може отримати потенційний інвестор від комерціалізації нашої розробки, складе:

$$\text{ПП}^{10\%} = \frac{969}{(1+0,1)^2} + \frac{1146}{(1+0,1)^3} + \frac{1322}{(1+0,1)^4} \approx 801 + 861 + 903 = 2565 \text{ тисяч грн.}$$

Теперішня вартість інвестицій PV , що можуть бути вкладені для реалізації нашої розробки: $PV = (1,0...5) \times B_{\text{заг}}$.

Для нашого випадку приймемо:

$$PV = (1,0...5) \times 95 = 5 \times 95 = 475 \text{ тисяч грн.}$$

Розраховуємо абсолютний ефект від можливих вкладених інвестицій $E_{\text{абс}}$.

$$E_{\text{абс}} = \text{ПП} - PV, \quad (4.12)$$

де ПП – приведена вартість збільшення всіх чистих прибутків, що їх може отримати потенційний інвестор від можливого впровадження нашої розробки, грн;

PV – теперішня вартість інвестицій $PV = 475$ тисяч грн.

Абсолютний ефект від можливого впровадження нашої розробки (при прогнозованому ринку збуту) за три роки складе:

$$E_{\text{абс}} = 2565 - 475 = 2090 \text{ тисяч грн.}$$

Оскільки $E_{\text{абс}} > 0$, то комерціалізація нашої розробки може бути доцільною.

Далі розрахуємо внутрішню дохідність E_v вкладених інвестицій:

$$E_v = T_{\text{ж}} \sqrt[1 + \frac{E_{\text{абс}}}{PV}]{} - 1, \quad (4.13)$$

де $E_{\text{абс}}$ – абсолютний ефект вкладених інвестицій; $E_{\text{абс}} = 2090$ тис. грн;

PV – теперішня вартість початкових інвестицій $PV = 475$ тис. грн;

$T_{\text{ж}}$ – життєвий цикл розробки, роки.

$T_{\text{ж}} = 4$ років (2024-й, 2025-й, 2026-й, 2027-й роки)

Для нашого випадку отримаємо:

$$E_b = \sqrt[4]{1 + \frac{2090}{475}} - 1 = \sqrt[4]{1 + 4,4000} - 1 = \sqrt[4]{5,4000} - 1 = 1,524 - 1 = 0,524 = 52,4\%.$$

Далі визначимо ту мінімальну дохідність, нижче за яку потенційному інвестору не вигідно буде займатися комерціалізацією нашої розробки.

Мінімальна дохідність або мінімальна (бар'єрна) ставка дисконтування $\tau_{\min}$ визначається за формулою:

$$\tau_{\min} = d + f, \quad (4.14)$$

де d – середньозважена ставка за депозитними операціями в комерційних банках; в 2024 році в Україні $d = (0,15...0,19)$;

f – показник, що характеризує ризикованість вкладень; $f = (0,05...0,40)$.

Для нашого випадку отримаємо:

$$\tau_{\min} = 0,15 + 0,35 = 0,50 \text{ або } \tau_{\min} = 50\%.$$

Оскільки величина $E_b = 52,4\% > \tau_{\min} = 50\%$, то потенційний інвестор у принципі може бути зацікавлений у фінансуванні та комерціалізації розробленого нами модифікованого методу стиснення зображень.

Далі розраховуємо термін окупності коштів, вкладених у можливу комерціалізацію розробленого нами модифікованого методу стиснення зображень.

Термін окупності $T_{\text{ок}}$ розраховується за формулою:

$$T_{\text{ок}} = \frac{1}{E_b}. \quad (4.15)$$

Для нашого випадку термін окупності $T_{ок}$ коштів (інвестицій) становитиме:

$$T_{ок} = \frac{1}{0,534} = 1,9 \text{ років} < 3 \text{ років},$$

що свідчить про потенційну доцільність комерціалізації розробленого нами модифікованого методу стиснення зображень.

Далі проведено моделювання залежності величини внутрішньої дохідності вкладених потенційних інвестицій від рівня інфляції в країні (який в умовах війни, на жаль, може зростати).

Прийнявши рівень інфляції у 20%, отримаємо:

$$ПП^{20\%} = \frac{969}{(1+0,2)^2} + \frac{1146}{(1+0,2)^3} + \frac{1322}{(1+0,2)^4} \approx 673 + 663 + 638 = 1974 \text{ тисяч грн.}$$

Тоді абсолютний ефект від можливого впровадження нашої розробки за три роки складе:

$$E_{абс} = 1974 - 475 = 1499 \text{ тисяч грн.}$$

Внутрішня дохідність E_v вкладених інвестицій становитиме:

$$E_v = T_{ж} \sqrt[3]{1 + \frac{E_{абс}}{PV}} - 1,$$

де $E_{абс}$ – абсолютний ефект вкладених інвестицій; $E_{абс} = 1499$ тисяч грн;

PV –теперішня вартість початкових інвестицій $PV = 475$ тисяч грн.

Для нашого випадку отримаємо:

$$E_b = \sqrt[4]{1 + \frac{1499}{475}} - 1 = \sqrt[4]{1 + 3,1557} - 1 = \sqrt[4]{4,1557} - 1 = 1,428 - 1 = 0,428 = 42,8\%.$$

Оскільки величина $E_b = 42,8\% < \tau_{\min} = 50\%$, то потенційний інвестор у принципі може бути не зацікавлений у фінансуванні та комерціалізації розробленого нами модифікованого методу стиснення зображень.

Прийнявши рівень інфляції у 30%, отримаємо:

$$\text{ПП}^{30\%} = \frac{969}{(1+0,3)^2} + \frac{1146}{(1+0,3)^3} + \frac{1322}{(1+0,3)^4} \approx 574 + 522 + 463 = 1559 \text{ тисяч грн.}$$

Тоді абсолютний ефект від можливого впровадження нашої розробки за три роки складе:

$$E_{\text{абс}} = 1559 - 475 = 1084 \text{ тисячі грн.}$$

Внутрішня дохідність E_b вкладених інвестицій становитиме:

$$E_b = \sqrt[T_{\text{ж}}]{1 + \frac{E_{\text{абс}}}{PV}} - 1,$$

де $E_{\text{абс}}$ – абсолютний ефект вкладених інвестицій; $E_{\text{абс}} = 1084$ тисяч грн;

PV –теперішня вартість початкових інвестицій $PV = 475$ тисяч грн.

Для нашого випадку отримаємо:

$$E_b = \sqrt[4]{1 + \frac{1084}{475}} - 1 = \sqrt[4]{1 + 2,2821} - 1 = \sqrt[4]{3,2821} - 1 = 1,346 - 1 = 0,346 = 34,6\%.$$

Оскільки величина $E_b = 34,6\% < \tau_{\min} = 50\%$, то потенційний інвестор ще більшою мірою може бути не зацікавлений у фінансуванні та

комерціалізації розробленого нами модифікованого методу стиснення зображень.

Зроблені розрахунки у вигляді графіків наведено на рисунку 4.1.

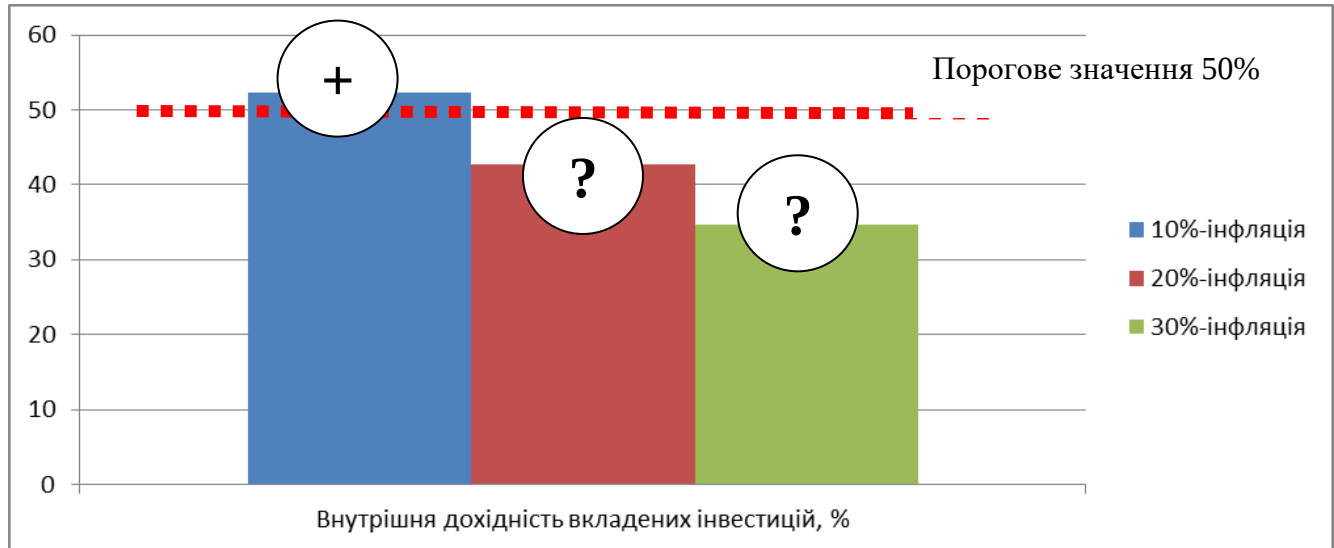


Рисунок 4.1 – Моделювання залежності величини внутрішньої дохідності потенційних інвестицій, вкладених у комерціалізацію розробленого нами модифікованого методу стиснення зображень, від рівня інфляції в країні

Аналіз графіка на рисунку 4.1 показує, що при рівні інфляції в 10% величина внутрішньої дохідності потенційних інвестицій, вкладених у комерціалізацію розробленого нами модифікованого методу стиснення зображень, може становити $E_v = 52,4\%$, що більше порогового значення $\tau_{\min} = 50\%$, і тому комерціалізація нашої розробки може бути доцільною. При рівні інфляції в 20%, 30% і вище величина внутрішньої дохідності інвестицій E_v , вкладених в комерціалізацію нашої розробки, буде меншою за $\tau_{\min} = 50\%$, і тому комерціалізація нашої розробки (при нинішніх зовнішніх умовах) може бути для потенційного інвестора не доцільною. Для прийняття остаточного рішення потрібно провести додаткові обґрунтування і розрахунки (наприклад, знизити рівень прийнятого ризику, підняти ціну реалізації нашої розробки тощо).

4.4 Висновки до економічного розділу

Результати виконаної економічної частини магістерської валіфікаційної роботи зведено у таблицю 4.6

Таблиця 4.6 - Результати економічної частини

Показники	Задані у ТЗ	Досягнуті у магістерській кваліфікаційній роботі	Висновок
1. Витрати на розробку модифікованого методу	Приблизно 100 тис. грн	95 тис. грн.	Практично досягнуто
2. Абсолютний ефект від впровадження розробки, тисяч грн	В межах 2000 тисяч грн (за три роки)	2090 тисяч грн	Практично виконано
3. Внутрішня дохідність потенційних інвестицій, %	не менше 50%	52,4%	Досягнуто
4. Термін окупності інвестицій, роки	до 3-ти років	1,9 років	Виконано

Таким чином, основні техніко-економічні показники розробленого нами модифікованого методу стиснення зображень, визначені у технічному завданні, виконані.

ВИСНОВКИ

Методи стиснення зображення поділяються на дві великі групи – методи стиснення без втрат та методи стиснення з втратами.

Методи стиснення без втрат дозволяють отримати відновлене зображення ідентичне вихідному. Такі методи стиснення використовують тоді, коли ідентичність вхідного та вихідного зображень є критичною або коли самі шуми зображення є об'єктом дослідження – наприклад, в медицині та науковій діяльності. Методи стиснення без втрат дають в результаті невеликі коефіцієнти стиснення.

Методи стиснення з втратами використовують у випадках, коли зміни зображення, які непомітні людському оку є прийнятними фото- та відеокамери, особливо веб. Такі зміни відновленого зображення дають змогу досягти значно більших коефіцієнтів стиснення, ніж при стисненні без втрат.

Проаналізовано сучасні методи стиснення зображень та виконано вибір аналогу.

Розглянуто застосування ортогональних дискретних перетворень для стиснення зображень. Оскільки в базовому кодеку JPEG використовується дискретне косинусне перетворення, саме його і обрано для вирішення задачі модифікації алгоритму стиснення зображень JPEG

Запропоновано модифікацію JPEG-MOD, що дає вигреш у стиску до 20% порівняно з базовим кодеком JPEG залежно від необхідної якості та вихідного розміру зображення. Модифікація дає вигреш у стиску як на зображеннях з дрібними деталями, так і на фонових зображеннях.

Використання запропонованих модифікацій призводить до підвищення ефективності стиснення у всьому діапазоні якості. При високій якості вигреш становить близько 0.5%, при дуже високій якості - до 4.5%, а при середній і низькій якості зростає до 20%, залежно від розміру зображення.

При роботі із зображеннями розміром від 0.2 до 6.3 Мп виграш у стиску змінюється наступним чином: від 2.0 до 4.0% у діапазоні середньої якості, від 5.0 до 20.0% у діапазоні низької якості.

Модифікація не збільшує трудомісткість базового алгоритму JPEG при компресії та декомпресії зображень. Тому цю зміну можна впроваджувати в існуючі програмні та апаратні модулі стиснення зображення на основі JPEG.

Можливе перенесення модифікації JPEG-MOD у відеокодеки на основі JPEG, наприклад Motion-JPEG.

Результати виконаної економічної частини магістерської кваліфікаційної роботи:

- витрати на розробку – 95 тис. грн.;
- абсолютний ефект від впровадження розробки – 2090 тисяч грн.;
- внутрішня дохідність потенційних інвестицій, % - 52,4%;
- термін окупності – 1,9 років.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Acharya T. Image Processing Principles and Application / T. Acharya, A. Ray. – New Jersey: John Wiley & Sons, Inc. Hoboken, 2005. – 425 p. – ISBN 978-0-4717-1998-4.
2. Юдін О. К. Технологія стиснення на базі методу кодування двійкових послідовностей за кількістю бітових переходів / О. К Юдін, М. Б. Гумен, К. О. Курінь // Захист інформації. – 2012. – Т. 14. – №4. – С. 12 – 18.
3. Daubechies L, Sweldens W. Factoring Wavelet Transforms into Lifting Steps // IEEE Trans, on Image Processing. — 2013. — Vol. 9. No. 3. — pp. 480-496.
4. Digital photography with flash and no-flash image pairs / G. Petschnigg, M. Agrawala, H. Hoppe and others // ACM Transactions on Graphics. – 2014. – Vol. 23. – № 3. – P. 664 – 672.
5. Gonzales R.C. Digital Image Processing Using MATLAB. / Gonzales R.C., Woods R.E., Eddins S. /– Prentice Hall, Upper Saddle River, NJ, 2004 – 492 p.
6. Yuhui Sun. Evolution-Operator-Based Single-Step Method for Image Processing / Yuhui Sun, PeiruWu, Wei G.W.// International Journal of Biomedical Imaging Volume. – 2016. – Article ID 83847. – P. 1–27.
7. Dougherty E.R. Random Processes for Image and Signal Processing. / Dougherty E.R. / – NY: IEEE Press, 2020. – 639 p.
8. Damerval C. A fast algorithm for bidimensional EMD / C. Damerval, S. Meignen, V. Perrier : IEEE Signal Processing Letters. 2015. V. 12(10). c.701-704.
9. Tiwari, Deepshikha, and Vipin Tyagi. "Digital Image Compression." Indian Science Cruiser 31, no. 6 (November 1, 2017): 44. <http://dx.doi.org/10.24906/isc/2017/v31/i6/166460>.
10. Brysina, Iryna Victorivna, and Victor Olexandrovysh Makarichev. "Discrete atomic compression of digital images: almost lossless compression."

radioelectronic and computer systems, no. 1 (March 23, 2019): 29–36.
<http://dx.doi.org/10.32620/reks.2019.1.03>.

11. Mohammed, Sajaa G., Safa S. Abdul-Jabbar, and Faisel G. Mohammed. "Art Image Compression Based on Lossless LZW Hashing Ciphering Algorithm." *Journal of Physics: Conference Series* 2114, no. 1 (December 1, 2021): 012080.
<http://dx.doi.org/10.1088/1742-6596/2114/1/012080>.

12. Yang, J., and K. Bhattacharya. "Combining Image Compression with Digital Image Correlation." *Experimental Mechanics* 59, no. 5 (January 18, 2019): 629–42. <http://dx.doi.org/10.1007/s11340-018-00459-y>.

13. Chaturvedi, Soumya. "Different Type of Image Compression using Various techniques, Highlighting Segmentation based image Compression." *International Journal for Research in Applied Science and Engineering Technology* 10, no. 2 (February 28, 2022): 171–77.
<http://dx.doi.org/10.22214/ijraset.2022.40207>.

14. El Banhawey, Magy, Walaa Saber, and Fathy Amer. "A native enhanced elastic extension tables multi-tenant database." *International Journal of Electrical and Computer Engineering (IJECE)* 10, no. 6 (December 1, 2020): 6618.
<http://dx.doi.org/10.11591/ijece.v10i6.pp6618-6628>.

15. Fathahillah, Fathahillah, Satria Gunawan Zain, and Rismawati Rismawati. "Homogeneous Image Compression Techniques with the Shannon-Fano Algorithm." *International Journal of Environment, Engineering and Education* 1, no. 2 (August 28, 2019): 59–66.
<http://dx.doi.org/10.55151/ijeedu.v1i2.17>.

16. Braner, Dana, and Barry Markovitz. "Digital Image Compression: Pixels n' Bits." *Journal of Intensive Care Medicine* 18, no. 4 (July 2003): 229–30.
<http://dx.doi.org/10.1177/0885066603254290>.

17. Гармаш В.В. Метод стиснення цифрових зображень на основі усічення простору вейвлет-коефіцієнтів / В. В. Гармаш, Н. М. Ольшанська: XLVIII Науково-технічна конференція факультету комп'ютерних систем і

автоматики (2019). URL : <https://conferences.vntu.edu.ua/index.php/all-fksa/all-fksa-2019/paper/view/6617/5488>.

18. Гармаш В. Вейвлет-кодер з корекцією вейвлет-коефіцієнтів / В. Гармаш, Ю. Сокирко // Матеріали XLVIII науково-технічної конференції підрозділів ВНТУ, Вінниця, 11-15 травня 2019 р. – Електрон. текст. дані. – 2019. – URL: <https://conferences.vntu.edu.ua/index.php/all-fksa/all-fksa-2019/paper/view/6625/5489>.

19. Wallace G.K. JPEG algoritm for image compression standard: Communications of the ACM. 1991. Vol. 34. №4. с. 30-44.

20. Волошина А.О. Модифікація алгоритму стиснення JPEG /А. О. Волошина, В. В. Гармаш [Електронний ресурс]: LI Науково-технічна конференція факультету комп'ютерних систем і автоматики (2022). – Режим доступу: <https://conferences.vntu.edu.ua/index.php/all-fksa/all-fksa-2022/paper/view/15276/12923>

21. Sheikh H. R., Bovik A. C. Image Information and Visual Quality: IEEE Transactions on Image Processing. 2016. Vol. 15. с. 430-444.

22. Беляєв І.А. СФ-блок кодування Хаффмана для стиснення зображень за стандартом JPEG: Проблеми розробки перспективних мікро- та наноелектронних систем. 2010. с. 332-335.

23. Cahya Dewi, Dewa Ayu Indah, and I. Made Oka Widyantara. "Usage analysis of SVD, DWT and JPEG compression methods for image compression." Jurnal Ilmu Komputer 14, no. 2 (September 30, 2021): 99. <http://dx.doi.org/10.24843/jik.2021.v14.i02.p04>.

24. Гринь Аліна Олександрівна, Гармаш Володимир Володимирович «Метод стиснення зображень на основі модифікованого алгоритму JPEG», «Науково-технічна конференція факультету інтелектуальних інформаційних технологій та автоматизації (2024)», 2024. URL: <http://surl.li/ulyug>

25. The USC-SIPI Image Database. URL: <http://sipi.usc.edu/database/> (дата звернення 12.06.2024).

26. Методичні вказівки до виконання економічної частини магістерських кваліфікаційних робіт. / Укладачі В.О. Козловський, О.Й. Лесько, В.В.Кавецький. – Вінниця : ВНТУ, 2021. – 42 с.

ДОДАТКИ

Додаток А (обов'язковий)**Технічне завдання на магістерську кваліфікаційну роботу**

ЗАТВЕРДЖЕНО
Зав. кафедри АІТ,
д.т.н., проф.
_____ Олег БІСІКАЛО

“ ____ ” _____ 2024 р.

ТЕХНІЧНЕ ЗАВДАННЯ

на виконання магістерської кваліфікаційну роботу

**Розробка модифікованого методу стиснення зображень на основі
алгоритму JPEG в системах передачі інформації**

08-31.МКР.002.02.000 ТЗ

Студентка групи

_____ АКТ-22мз Аліна ГРИНЬ

Керівник

_____ к.т.н., доцент Володимир ГАРМАШ

Вінниця ВНТУ - 2024 рік

1. Назва та галузь застосування

Розробка модифікованого методу стиснення зображень на основі алгоритму JPEG в системах передачі інформації. Розроблене програмне забезпечення застосовуватиметься в системах передачі інформації.

2. Підстава проведення для розробки.

Підставою для виконання роботи є наказ №81 по ВНТУ від «11» березня 2024р.

3. Мета та призначення розробки.

Метою роботи є підвищення коефіцієнту стиснення цифрових зображень шляхом модифікації алгоритму стиснення JPEG, що дозволяє ефективно стискати зображення при мінімальному спотворенні.

Призначенням розробки є створення програмного забезпечення, яке би реалізовувало метод стиснення цифрових зображень.

4. Джерела розробки.

Магістерська кваліфікаційна робота виконується вперше. В ході проведення розробки повинні використовуватись такі документи:

1. Mohammed, Sajaa G., Safa S. Abdul-Jabbar, and Faisel G. Mohammed. "Art Image Compression Based on Lossless LZW Hashing Cipherring Algorithm." *Journal of Physics: Conference Series* 2114, no. 1 (December 1, 2021): 012080. <http://dx.doi.org/10.1088/1742-6596/2114/1/012080>.

2. Yang, J., and K. Bhattacharya. "Combining Image Compression with Digital Image Correlation." *Experimental Mechanics* 59, no. 5 (January 18, 2019): 629–42. <http://dx.doi.org/10.1007/s11340-018-00459-y>.

3. Chaturvedi, Soumya. "Different Type of Image Compression using Various techniques, Highlighting Segmentation based image Compression." *International Journal for Research in Applied Science and Engineering Technology*

10, no. 2 (February 28, 2022): 171–77.
<http://dx.doi.org/10.22214/ijraset.2022.40207>

4. Гармаш В.В. Метод стиснення цифрових зображень на основі усічення простору вейвлет-коефіцієнтів / В. В. Гармаш, Н. М. Ольшанська: XLVIII Науково-технічна конференція факультету комп'ютерних систем і автоматики (2019). URL : <https://conferences.vntu.edu.ua/index.php/all-fksa/all-fksa-2019/paper/view/6617/5488>.

5. Методичні вказівки до виконання економічної частини магістерських кваліфікаційних робіт. / Укладачі В.О. Козловський, О.Й. Лесько, В.В.Кавецький. – Вінниця : ВНТУ, 2021. – 42 с.

5. Вихідні дані:

- тип цифрового зображення – півтонові та кольорові;
- мінімальна роздільна здатність зображення 128 x 128 пікселів,
- максимальна роздільна здатність 2048 x 2048 пікселів;
- коефіцієнт стиснення 2 – 200.
- виграш у стисненні від 0,5% до 20% у порівнянні з класичним алгоритмом JPEG.

6. Економічні показники

До економічних показників входять:

- витрати на розробку – до 100 тис. грн.
- абсолютний ефект від впровадження розробки – в межах 2000 тисяч грн. (за три роки)
- Внутрішня дохідність потенційних інвестицій, % - не менше 50%
- термін окупності – не більше 3 років

7. Стадії та етапи розробки.

а) Аналіз сучасних методів стиснення зображень 12.03.2024 – 26.03.2024

б) Застосування ортогональних перетворень для стиснення зображень 27.03.2024 – 15.04.2024

в) Модифікація алгоритму JPEG 16.04.2024 – 02.05.2024

г) Експериментальні дослідження 03.05.2024 – 17.05.2024

д) Економічний розділ 20.05.2024 – 27.05.2024

е) Оформлення пояснювальної записки 28.05.2024 – 05.06.2024

8. Порядок контролю і приймання.

Рубіжний контроль провести до «20» травня 2024 р.

Попередній захист МКР провести «06» червня 2024 р.

Захист МКР провести до «20» червня 2024 р.

Додаток Б (обов'язковий)

Ілюстративна частина

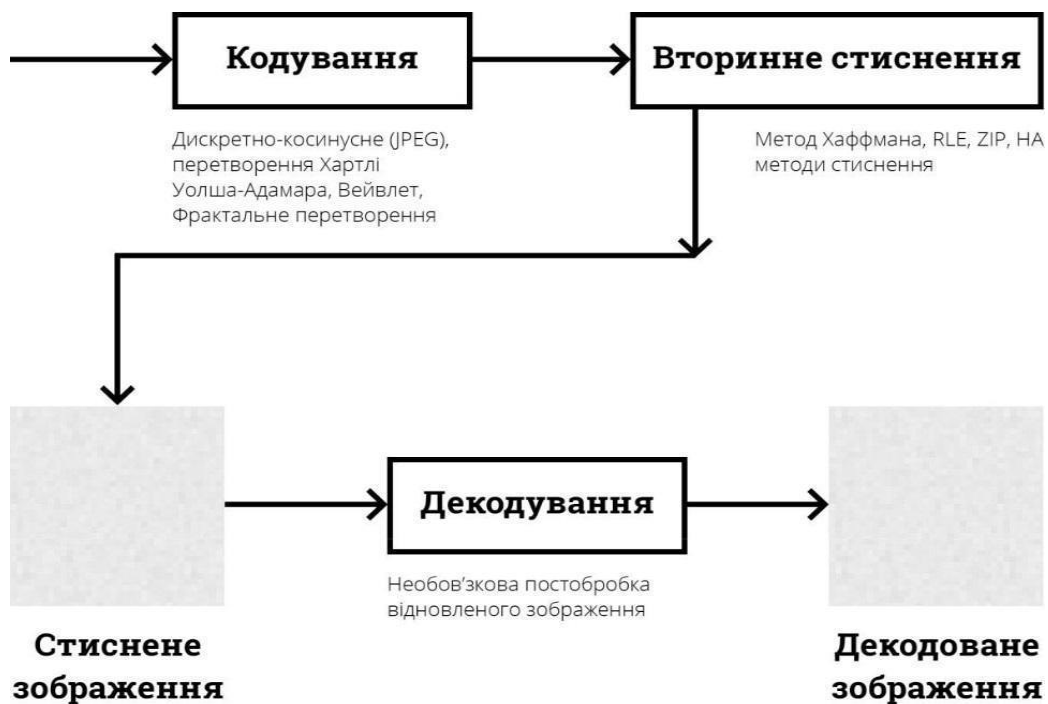


Рисунок Б.1 – Основні етапи стиснення цифрових зображень

Продовження додатка Б



Рисунок Б.2 – Схема стиснення зображення по стандарту JPEG

Продовження додатка Б

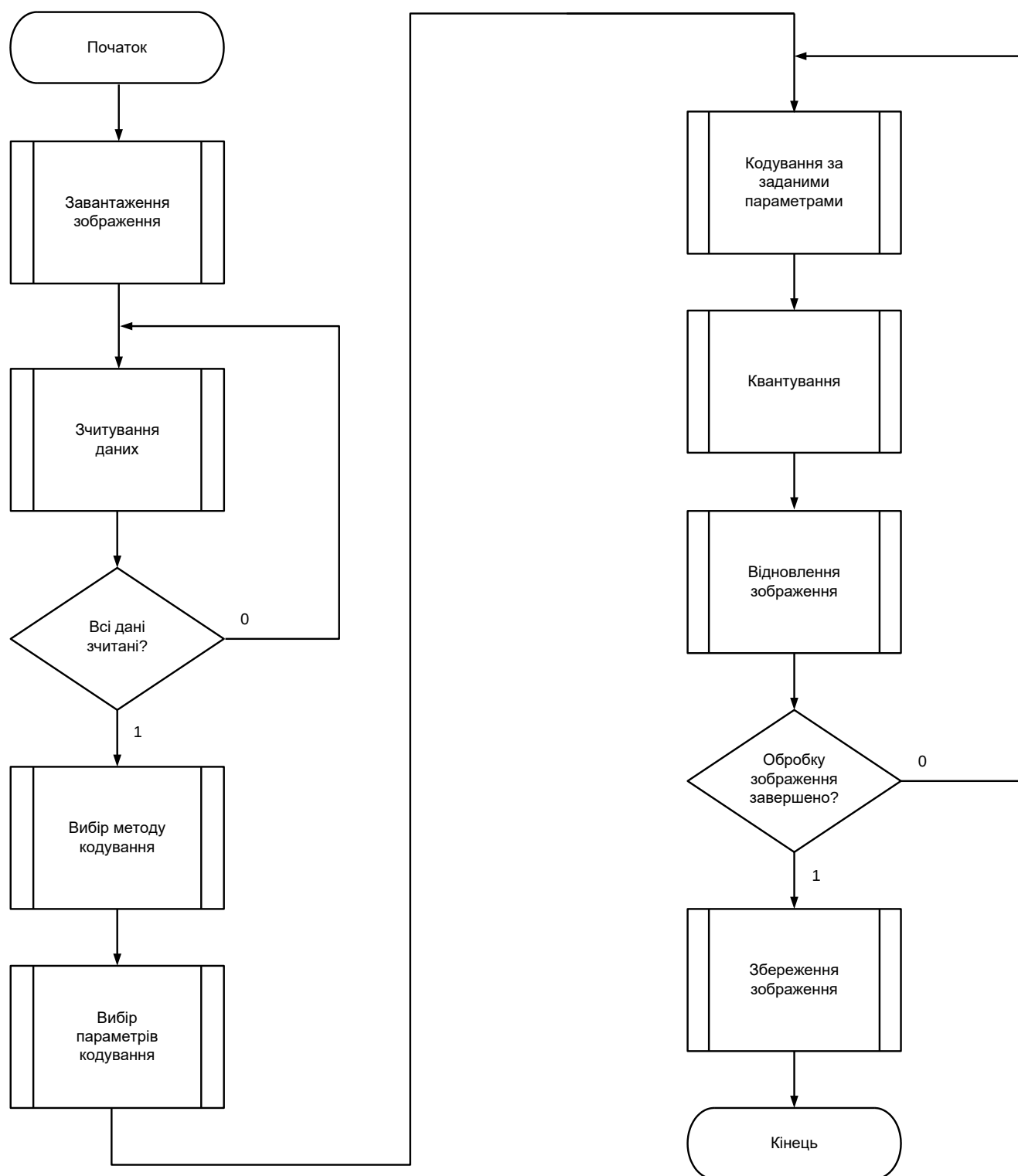


Рисунок Б.3 – Алгоритм стиснення зображень

Продовження додатка Б

Бітові витрати на піксель зображень

Якість VIF	0.11	0.21	0.31	0.46	0.68	0.99
Кодек						
Baseline JPEG	0.159	0.218	0.349	0.541	0.939	3.46
JPEG-MOD	0.132	0.197	0.332	0.529	0.935	3.30
Виграш, %	20.4	10.6	5.12	2.26	0.427	4.85

Порівняння розмірів файлів

	Алгоритм стиснення	
<i>Ступінь стиснення</i>	<i>JPEG</i>	<i>JPEG-MOD</i>
50%	75 KB	66 KB
80%	68 KB	60 KB
90%	57 KB	55 KB

Значення критеріїв якості

		Ступінь стиснення								
		10%	20%	30%	40%	50%	60%	70%	80%	90%
JPEG	Середньокв. відхилення	1.0111	1.0361	1.0949	1.1776	1.3151	1.5413	2.0153	3.5573	33.0291
	PSNR	0.1302	0.3417	0.8212	1.4537	2.4128	3.7918	6.1209	11.0564	30.4119
JPEG-MOD	Середньокв. відхилення	2.5175	7.2188	934.9723	502.9834	1823.3	8434	5525.3	37415	16788000
	PSNR	8.0535	17.2033	59.45	54.0651	65.2509	78.5546	94.8811	91.495	164.5337

Рисунок Б.4 - Експериментальні дослідження

Продовження додатка Б

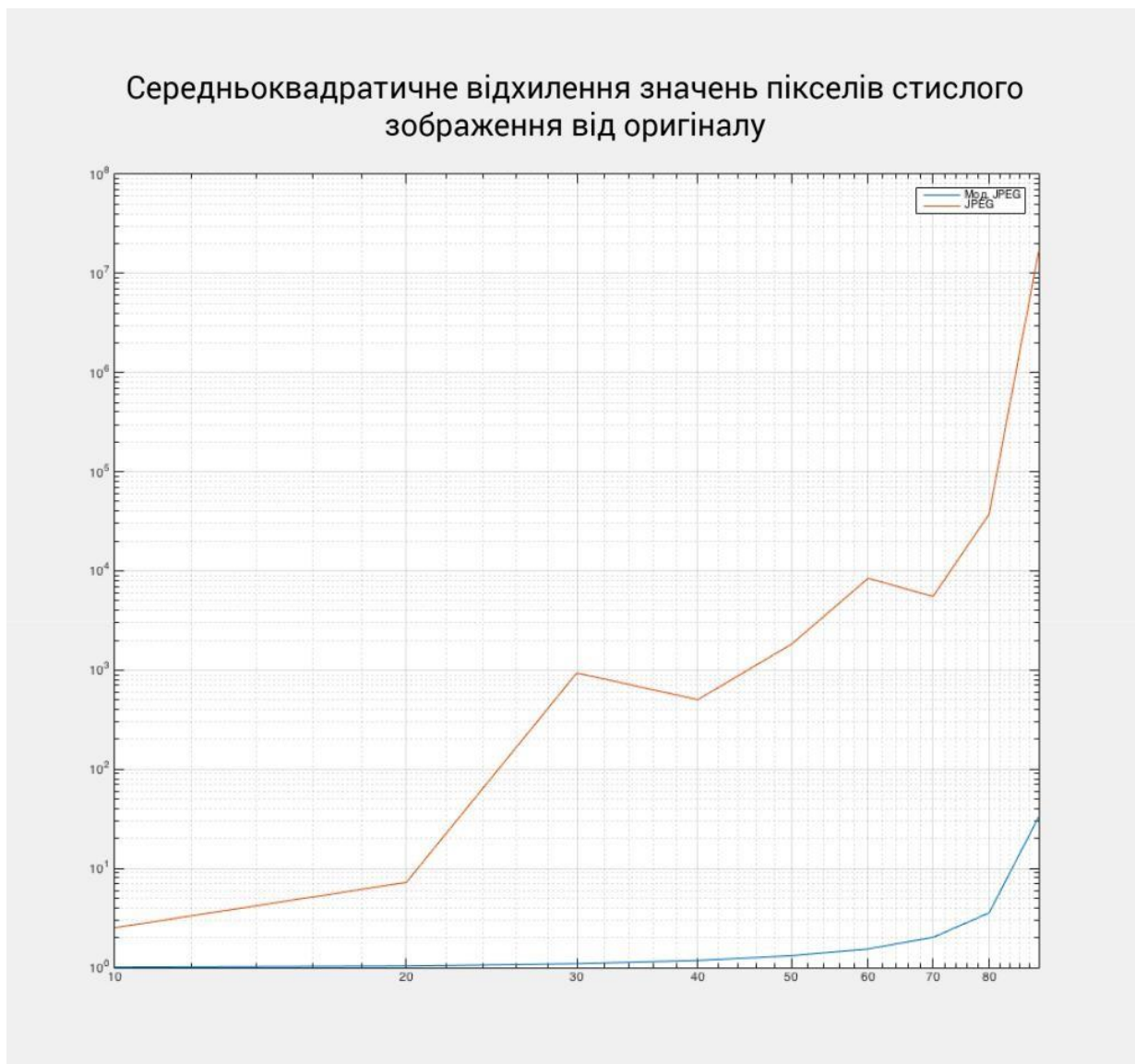


Рисунок Б.5 - Середньоквадратичне відхилення значень пікселів стислого зображення від оригіналу

Продовження додатка Б

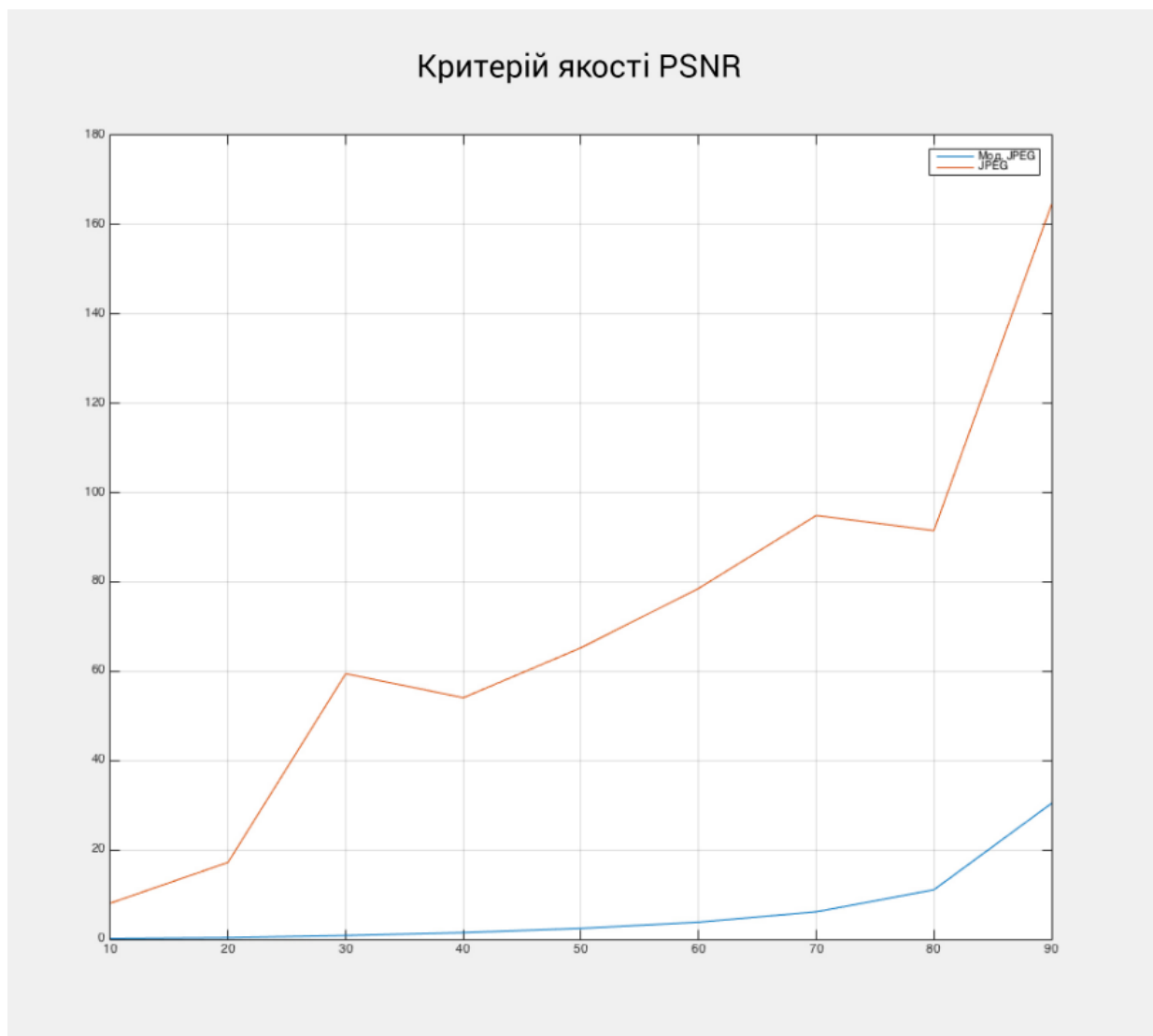


Рисунок Б.6 – Порівняння методів за критерієм якості PSNR

Додаток В (обов'язковий)

Лістинг реалізації модифікованого алгоритму стиснення JPEG

```

clc;
clear all;
I = imread('cameraman.tif');
I1=I;
[row coln P]= size(I);
I= double(I);
%-----
% Subtracting each image pixel value by 128 %--
%-----

I = I - (128*ones(256));
quality = input('What quality of compression you require - ');
%-----
% Quality Matrix Formulation
%-----
Q50 = [ 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];
if quality > 50
    QX = round(Q50.*(ones(8)*((100-quality)/50)));
    QX = uint8(QX);
elseif quality < 50
    QX = round(Q50.*(ones(8)*(50/quality)));
    QX = uint8(QX);
elseif quality == 50
    QX = Q50;
end
%-----
% Jpeg Compression
%-----
restored = zeros(row,coln);
QX = double(QX);
%-----
% Jpeg Encoding
%-----
X=I;
iter = 2; % number of wavelet iterations
cf = 1-(quality/100);
M = row;
N = coln;
% Haar basis constants
avg = 1/sqrt(2)*[1;1];
diff = 1/sqrt(2)*[1;-1];

```

```

%-----
% Compute the wavelet coefficients
%-----
tic
    WX=X;
    for i=1:iter
        %      if strcmp(type,'Haar')
        % Generate the  $M/(2^{(i-1)})$  mother and father Haar wavelets

        mother_m = [avg;zeros(M/(2^(i-1))-2,1)];

        father_m = [diff;zeros(M/(2^(i-1))-2,1)];
        % Generate the  $N/(2^{(i-1)})$  mother and father Haar wavelets

        mother_n = [avg;zeros(N/(2^(i-1))-2,1)];

        father_n = [diff;zeros(N/(2^(i-1))-2,1)];
        % Compute wavelet transform of upper left quadrant
        fft_mother_n_tilde = fft([mother_n(1);mother_n(end:-1:2)]');
        fft_father_n_tilde = fft([father_n(1);father_n(end:-1:2)]');
        fft_mother_m_tilde = fft([mother_m(1);mother_m(end:-1:2)]');
        fft_father_m_tilde = fft([father_m(1);father_m(end:-1:2)]');

        for p = 1:P
            for j = 1:M/(2^(i-1))
                lowfreq = ifft(fft_mother_n_tilde.*fft(WX(j,1:N/(2^(i-1))),p));

                detail = ifft(fft_father_n_tilde.*fft(WX(j,1:N/(2^(i-1))),p));

                WX(j,1:N/(2^(i-1)),p) = [lowfreq(1:2:end) detail(1:2:end)];
            end
            for j = 1:N/(2^(i-1))
                lowfreq = ifft(fft_mother_m_tilde.*fft(WX(1:M/(2^(i-1))),j,p));

                detail = ifft(fft_father_m_tilde.*fft(WX(1:M/(2^(i-1))),j,p));

                WX(1:M/(2^(i-1)),j,p) = [lowfreq(1:2:end);detail(1:2:end)];
            end
        end
    end
    ForwardTransformTime = toc %#ok
%-----
%-----
% Compress image by zeroing out cf% of the wavelet coefficients %-
%-----

WXc = WX;
if cf > 0
    % Zero out smallest cf% wavelet coefficients in absolute value
    coeffs = zeros(1,M*N*P);

    for p =
        1:P

```

```

    for i =
    1:M
        coeffs(M*N*(p-1) + ((i-1)*N+1:N*i)) = abs(WX(i,.,p));
    end
end
sortedcoeffs = sort(coeffs);
thresh = sortedcoeffs(ceil(cf*M*N*P)); indices = find((abs(WX) <=
thresh));
WXc(indices(randperm(length(indices),floor(cf*M*N*P)))) = 0;
elseif cf == -1
    % Keep the low frequency approximation and zero out everything else
    WXc = zeros(M,N,P);
    for p = 1:P
        WXc((M-M/(2^iter)+1):end,(N-N/(2^iter)+1):end,p) = WX((M-
M/(2^iter)+1):end,(N-N/(2^iter)+1):end,p);
    end
end
%-----
in = WXc;
% Z-scanning      initializing the variables
%-----
h = 1;
v = 1;
vmin = 1;
hmin = 1;
vmax = size(in, 1);
hmax = size(in, 2);
i = 1;
output = zeros(1, vmax * hmax);
%-----
while ((v <= vmax) & (h <= hmax))
    if (mod(h + v, 2) == 0) % going up
        if (v == vmin)
            output(i) = in(v, h); % if we got to the first line
            if (h == hmax)
                v = v + 1;
            end
        else
            h = h + 1;
        end;
        i = i + 1;
    elseif ((h == hmax) & (v < vmax)) % if we got to the last column
        output(i) = in(v, h);
        v = v + 1;
        i = i + 1;
    elseif ((v > vmin) & (h < hmax)) % all other cases

```

```

        output(i) = in(v, h);
    v = v - 1;
    h = h + 1;
    i = i + 1;
    end;
    % going down

    if ((v == vmax) & (h <= hmax))          % if we got to the last line
        output(i) = in(v, h);

        h = h + 1;
        i = i + 1;
    elseif (h == hmin)          % if we got to the first column
        output(i) = in(v, h);

        if (v == vmax)
            h = h + 1;
        else
            v = v + 1;
        end;
        i = i + 1;
    elseif ((v < vmax) & (h > hmin))  % all other cases

        output(i) = in(v, h);
        v = v + 1;
        h = h - 1;
        i = i + 1;
    end;

    end;
    if ((v == vmax) & (h == hmax))      % bottom right element
        output(i) = in(v, h);
        break
    end;
end;
output
%-----
% Reconstruct original image from compressed wavelet coefficients %-
-----

tic
Xhat =

WXc; for
i=iter:-
1:1

    % Generate the  $M/(2^{(i-1)})$  mother and father Haar wavelets
    mother_m = [avg;zeros(M/(2^(i-1))-2,1)];
    father_m = [diff;zeros(M/(2^(i-1))-2,1)];

```

```

% Generate the  $N/(2^{(i-1)})$  mother and father Haar wavelets
mother_n = [avg;zeros(N/(2^(i-1))-2,1)];
father_n = [diff;zeros(N/(2^(i-1))-2,1)]
% Compute inverse wavelet transform of upper left quadrant
fft_mother_m = fft(mother_m);
fft_father_m =
fft(father_m); fft_mother_n
= fft(mother_n);
fft_father_n = fft(father_n);
for p = 1:P
    for j = 1:N/(2^(i-1))
        up1 = upsample(Xhat(1:M/(2^i),j,p),2);
        up2 = upsample(Xhat(M/(2^i)+1:M/(2^(i-1)),j,p),2);
        Xhat(1:M/(2^(i-1)),j,p) = ifft(fft_mother_m.*fft(up1(:))) +
ifft(fft_father_m.*fft(up2(:)));
    end
    for j = 1:M/(2^(i-1))
        up1 = upsample(Xhat(j,1:N/(2^i),p),2);
        up2 = upsample(Xhat(j,N/(2^i)+1:N/(2^(i-1)),p),2);
        Xhat(j,1:N/(2^(i-1)),p) = (ifft(fft_mother_n.*fft(up1(:))) +
ifft(fft_father_n.*fft(up2(:))))';
    end
end
end
InverseTransformTime = toc %ok
I2 = Xhat;
% -----
% Conversion of Image Matrix to Intensity image
%-----

K=mat2gray(I2);
%-----
% Formulation of forward DCT Matrix and inverse DCT matrix %-
-----

DCT_matrix8 = dct(eye(8));
iDCT_matrix8 = DCT_matrix8'; %inv(DCT_matrix8);
%-----
% Jpeg Compression
%-----
    dct_restored = zeros(row,coln);
QX = double(QX);
%-----
% Jpeg Encoding

```

```

%-----
%-----
% Forward Discret Cosine Transform
%-----
for i1=[1:8:row]
    for i2=[1:8:coln]
        zBLOCK=I(i1:i1+7,i2:i2+7);
        win1_dct=DCT_matrix8*zBLOCK*iDCT_matrix8;

        dct_domain(i1:i1+7,i2:i2+7)=win1_dct;

    end
end
%-----
% Quantization of the DCT coefficients
%-----
for i1=[1:8:row]
    for i2=[1:8:coln]
        win1_dct = dct_domain(i1:i1+7,i2:i2+7);
        win2_dct=round(win1_dct./QX);
        dct_quantized(i1:i1+7,i2:i2+7)=win2_dct;
    end
end
%-----
% Jpeg Decoding
%-----
% Dequantization of DCT Coefficients
%-----
for i1=[1:8:row]
    for i2=[1:8:coln]
        win2_dct = dct_quantized(i1:i1+7,i2:i2+7);
        win3_dct = win2_dct.*QX;
        dct_dequantized(i1:i1+7,i2:i2+7) = win3_dct;
    end
end
%-----
% Inverse DISCRETE COSINE TRANSFORM %--
%-----

for i1=[1:8:row]
    for i2=[1:8:coln]
        win3_dct = dct_dequantized(i1:i1+7,i2:i2+7);
        win4_dct=iDCT_matrix8*win3_dct*DCT_matrix8;

        dct_restored(i1:i1+7,i2:i2+7)=win4_dct;

    end
end
I2_dct=dct_restored;
% -----
% Conversion of Image Matrix to Intensity image
%-----

```

```
K_dct=mat2gray(I2_dct);
%-----
%Display of Results
%-----
K1 = single(K);
K1_dct = single(K_dct);
saveas(gcf,'jpg.jpg')
figure(2);imshow(I1);
saveas(gcf,'orig.jpg')
figure(3);imshow(K1);
saveas(gcf,['wavelet-',num2str(quality),'.jpg'])
figure(4);imshow(K1_dct);
saveas(gcf,['dct-',num2str(quality),'.jpg'])
```

Додаток Г (обов'язковий)

Протокол перевірки магістерської кваліфікаційної роботи на наявність
текстових запозичень

Назва роботи: «Розробка модифікованого методу стиснення зображень на основі алгоритму JPEG в системах передачі інформації»

Тип роботи: магістерська кваліфікаційна робота

Підрозділ: кафедра АІТ

Показники звіту подібності Unicheck

Оригінальність 84,2% Схожість 15,8%

Аналіз звіту подібності (відмітити потрібне)

- ☐ Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.
- ☐ Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і відсутності самостійності її автора. Роботу направити на розгляд експертної комісії кафедри.
- ☐ Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень.

Особа, відповідальна за перевірку _____ Роман МАСЛІЙ
(підпис) (ім'я, прізвище)

Ознайомлені з повним звітом подібності, який був згенерований системою
Unicheck щодо роботи.

Автор роботи _____ Аліна ГРИНЬ
(підпис) (ім'я, прізвище)

Керівник роботи _____ Володимир ГАРМАШ
(підпис) (ім'я, прізвище)