ДНІПРОВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

ІМЕНІ ОЛЕСЯ ГОНЧАРА

Факультет прикладної математики

Кафедра обчислювальної математики та математичної кібернетики

Дипломна робота

МОДЕЛІ ТА МЕТОДИ ВІДНОВЛЕННЯ СПОТВОРЕНИХ ЦИФРОВИХ ЗОБРАЖЕНЬ

перший (бакалаврський) рівень вищої освіти

спеціальність 113 Прикладна математика

освітня програма Комп’ютерне моделювання та технології програмування

Виконавець

студент групи ПА–17–2

Гурдіш Анастасія Олегівна \_\_\_\_\_\_\_\_\_\_\_\_\_\_

прізвище, ім’я, по-батькові підпис

Керівник

доцент кафедри КТ

наук. ступ., вчене звання

М. Є. Сердюк

ініціали, прізвище підпис

Завідувач кафедри комп’ютерних технологій  
д-р. фіз.-мат. наук, проф.

Н. А. Гук                             \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

2021

**Дніпровський національний університет імені Олеся Гончара**

Факультет прикладної математики

Кафедра обчислювальної математики та математичної кібернетики

Рівень вищої освіти перший (бакалаврський)

Спеціальність 113 Прикладна математика

Освітня програма Комп’ютерне моделювання та технології програмування

**ЗАТВЕРДЖУЮ**

**Завідувач кафедри комп’ютерних**

**технологій**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Н.А. Гук**

(підпис)

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2021 року

**З А В Д А Н Н Я**

**НА ВИПУСКНУДИПЛОМНУ РОБОТУ**

                                              Гурдіш Анастасії Олегівні

(прізвище, ім’я, по батькові)

1. Тема роботи Моделі та методи відновлення спотворених цифрових зображень

керівник роботи Сердюк Марина Євгеніївна, доцент кафедри Комп’ютерних технологій

( прізвище, ім’я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по Університету від «19» березня 2021 року № 332с

2. Строк подання роботи 31 травня 2021 р.

3. Вхідні дані до роботи Спотворені апаратною функцією зображення

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

1. Провести огляд методів та програмних засобів для розв’язання задачі відновлення спотворених зображень.

2. Розробити власне програмне забезпечення для відновлення спотворених зображень за допомогою їх спектра Фур’є.

3. Провести практичну апробацію програмного забезпечення на реальних даних і оцінити якість роботи алгоритму.

5. Перелік графічного матеріалу (з точним зазначенням обов’язкових креслень)

Презентація у Microsoft PowerPoint                                                                             \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

6. Консультанти розділів роботи

|  |  |  |  |
| --- | --- | --- | --- |
| Розділ | Прізвище, ініціали та посада  Консультанта | Підпис, дата | |
| завдання видав | завдання прийняв |
| 1 | Сердюк М. Є. |  |  |
| 2 | Сердюк М. Є. |  |  |
| 3 | Сердюк М. Є. |  |  |
| 4 | Сердюк М. Є. |  |  |

7. Дата видачі завдання        19 жовтня 2020 р.

**КАЛЕНДАРНИЙ ПЛАН**

|  |  |  |  |
| --- | --- | --- | --- |
| №  з/п | Назва етапів дипломної роботи | Строк виконання етапів роботи | Примітка |
| 1 | Знайомство із задачею відновлення зображень, методами її розв’язання, існуючими програмними застосунками. Вивчення способу відновлення зображень на базі методів, що використовують спектральне представлення. Опрацювання літературних джерел. | 17.03.2021 –  29.04.2021 |  |
| 2 | Розроблення і тестування програмного забезпечення для відновлення спотворених апаратною функцією зображень на базі методів, що використовують спектральне представлення. | 29.03.2021 – 26.04.2021 |  |
| 3 | Проведення практичної апробації програмного забезпечення на реальних даних, оцінка якості та аналіз результатів. | 26.04.2021 – 17.05.2021 |  |
| 4 | Оформлення роботи | 17.05.2021 – 31.05.2021 |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Студент А. О. Гурдіш       (*ініціали, прізвище*)

( підпис )

Керівник роботи М. Є. Сердюк      (*ініціали, прізвище*)

( підпис )

РЕФЕРАТ

**Дипломна робота:** 67 с., 62 рис., 0 табл., 14 джерел, 0 додатків.

**Об'єкт дослідження:**процес відновлення спотворених цифрових зображень

**Мета роботи:** розробити програмне забезпечення, яке б дозволяло швидко та якісно відновлювати спотворені цифрові зображення.

**Одержані висновки та їх новизна:** під час виконання роботи був проведений детальний аналіз проблеми відновлення спотворених зображень, методів та алгоритмів, які вирішують цю задачу, методів обробки зображень у частотній області та представлення зображення за допомогою спектра Фур’є, було розроблено програмний застосунок та проведено порівняльну характеристику роботи алгоритму з різними зображеннями.

**Результати досліджень можуть бути застосовані у** сферах обробки та відновлення зображень, у аерокосмічній області, а також у сферах обробки медичних чи супутникових зображень.

**Перелік ключових слів:** ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ, ВІДНОВЛЕННЯ ЗОБРАЖЕНЬ, СПЕКТР ФУР’Є, ЧАСТОТНА ОБЛАСТЬ, СПЕКТР, СПОТВОРЕНЕ ЗОБРАЖЕННЯ, ВІДНОВЛЕНЕ ЗОБРАЖЕННЯ, МОДЕЛЬ, WINDOWN FORMS.

ANNOTATION

The graduation research of the 4-year student A. Gurdish (Oles Honchar Dnipro National University, Faculty of Applied Mathematics, Department of Computer Technologies) deals with desktop software development for Windows operation system.

Purpose of the work to develop a desktop application for the Windows Operating System aimed at solving big problems of reconstruction of the image. This system will have a number of advantages over its counterparts and will be relevant in use.

In the process of the work at software «Recoveric» were thorough analyzed the problem of recovering of distorted images, methods and algorithms that solve this problem, methods of image processing in the frequency domain and image representation using the Fourier spectrum. The software was developed and a comparative characteristic of the work of the algorithm was carried out with different images.

This desktop application was implemented using C# programming language, .Net Framework and Windows Forms. Developed software can be used on any Windows version.

Bibliography 9, pictures 62, tables 0.

# ЗМІСТ

[ВСТУП 7](#_Toc73380617)

[ПОСТАНОВКА ЗАДАЧІ 10](#_Toc73380618)

[1. АНАЛІТИЧНИЙ ОГЛЯД ЛІТЕРАТУРНИХ ДЖЕРЕЛ 11](#_Toc73380619)

[1.1. Основні поняття теорії сигналів 11](#_Toc73380620)

[1.2. Цифрове зображення та його обробка 13](#_Toc73380621)

[1.3. Моделі зображень та їх спотворень 16](#_Toc73380622)

[1.3.1. Трансляційно-інваріантні спотворення 17](#_Toc73380623)

[1.3.2. Моделі шуму 21](#_Toc73380624)

[1.3.3. Спотворення зображення в наслідок змазу 25](#_Toc73380625)

[1.3.4. Спотворення зображень апаратною функцією 29](#_Toc73380626)

[2. МОДЕЛІ ТА МЕТОДИ ВІДНОВЛЕННЯ ЗОБРАЖЕНЬ 31](#_Toc73380627)

[2.1. Методи обробки зображень у частотній області 32](#_Toc73380628)

[2.2. Алгоритм відновлення зображення у частотній області 36](#_Toc73380629)

[2.3. Повний алгоритм вирішення задачі про відновлення зображення за допомогою спектру Фур’є 41](#_Toc73380630)

[3. ОПИС РОЗРОБЛЕНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ 44](#_Toc73380631)

[3.1. Розбір використаних технологій та графічних інтерфейсів 44](#_Toc73380632)

[3.2. Розбір основних моментів 45](#_Toc73380633)

[АНАЛІЗ ОТРИМАНИХ РЕЗУЛЬТАТІВ РОЗВ’ЯЗАННЯ ЗАДАЧІ 58](#_Toc73380634)

[ВИСНОВКИ 65](#_Toc73380635)

[СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ 66](#_Toc73380636)

# ВСТУП

У сьогодення широкого розвитку набули галузі науки, в яких застосовують системи обробки інформації з використанням даних у цифровому вигляді. Базовими складовими елементами таких систем є системи цифрового перетворення інформації, призначені для переведення в цифровий вигляд природніх або штучних об'єктів, які є носіями інформації в аналоговому вигляді. Дані в цифровому вигляді використовують для збереження, передачі, аналізу та обробки в системах цифрової обробки інформації [1].

Багато галузей техніки, що мають відношення до отримання, обробки, зберігання та передачі інформації, в значній мірі орієнтуються в даний час на розвиток систем, в яких інформація має характер зображень. Зображення, яке можна розглядати як двовимірний сигнал, є значно більш ємним носієм інформації, ніж звичайний одновимірний (часовий) сигнал. Але дуже часто під час перетворення або передачі сигналу ми втрачаємо частину інформації.

У зв'язку з цим, масово стали затребуваними задачі автоматичної обробки і аналізу зображень, що спрямовані на вирішення проблем обробки зображень, причому, пріоритетна увага приділяється цифровим методам, які приваблюють своєю гнучкістю.

Прикладом такої задачі є задача відновлення зображень в загублених або спотворених областях. Рішення такої задачі виявляється необхідним у різноманітних сферах нашого життя. Прикладами можуть слугувати:

* Редагування фотографій з метою видалення з них небажаних об'єктів (із заповненням їх областей тим фоном, який міг би бути на фотографії в їх відсутність), що псують композицію знімка;
* Видалення силуетів людей на публічно доступних знімках;
* Реставрація старих фільмів;
* Та багато іншого.

Наявність таких додатків викликає потребу у вирішенні задачі відновлення зображень і підтверджує актуальність теми не тільки в прикладному, але і науковому плані, так як дана задача не має в даний час задовільного рішення і викликає ряд питань. Існує велика кількість методів її часткового вирішення, розроблених в рамках різних підходів, що використовують різні типи представлень зображень. Прикладами таких методів можуть виступати алгоритми реконструкції на основі диференціальних рівнянь у частинних похідних, алгоритми реконструкції на основі пошуку за зразком, алгоритми напівавтоматичної реконструкції і тому подібне. Але при цьому відсутня узагальнююча теорія та система використання всіх цих методів та немає єдиних критеріїв їх оцінювання в зв'язку з тим, що задача «вгадування» відсутнього змісту виглядає як математично некоректна.

Можливість застосування аналізу та дослідження до задачі відновлення зображень в загублених або спотворених областях за допомогою спектру Фур’є може дозволити створити ємнісну теоретичну основу і розробити алгоритм, який відображатиме роботу цього методу, що й обумовлює актуальність цієї роботи.

У даній переддипломній роботі розглядаються методи обробки зображень у частотній області і алгоритм відновлення зображень за допомогою спектру Фур’є. Метою даної роботи є аналіз цього методу відновлення інформації в загублених або пошкоджених областях зображень під впливом апаратної функції та порівняння результатів роботи алгоритму в залежності від характеристик зображення.

Об'єктом дослідження є процес відновлення спотворених цифрових зображень.

Предмет дослідження – цифрові зображення, які зазнали локальних динамічних спотворень.

Дипломна робота складається з таких частин:

* вступ, який обґрунтовує актуальність роботи; визначає цілі проведення наукового дослідження; галузь дослідження; методи дослідження або розрахунків;
* постановка задачі;
* аналітичний огляду джерел;
* розділ про моделі та методи відновлення зображень;
* опис розробленого програмного забезпечення;
* аналіз результатів;
* висновки;
* список використаної літератури;
* додатки

# ПОСТАНОВКА ЗАДАЧІ

Провести дослідження методу та алгоритму відновлення зображень в загублених або спотворених областях, яке надалі можна використовувати для розробки програмного забезпечення для відновлення зображень. Студент має виконати такі задачі:

* дослідити та провести аналіз предметної області;
* провести огляд різноманітних моделей спотворення зображень;
* провести огляд існуючих алгоритмів відновлення зображень;
* провести аналіз методу відновлення зображення за допомогою спектру Фур’є;
* визначити функціональні вимоги до програмного забезпечення;
* вивчити відповідну технічну літературу по мові програмування С#;
* спроектувати та розробити додаток;
* зробити висновки, знайти можливі шляхи для подальшого поліпшення реалізації.

# 1. АНАЛІТИЧНИЙ ОГЛЯД ЛІТЕРАТУРНИХ ДЖЕРЕЛ

## 1.1. Основні поняття теорії сигналів

Перш за все, слід дати визначення такого поняття, як сигнал. Залежно від контексту воно може набувати різних значень. У загальному випадку сигнал - це зміна деякої фізичної величини. Залежно від області визначення говорять про тимчасову, частотну або просторову форми подання сигналу.

Сигнали найчастіше розглядають як функцію, задану в деяких фізичних координатах. За цим критерієм можна виділити одновимірні сигнали (залежні, наприклад, від часу), двовимірні сигнали, задані на площині (наприклад, зображення) і тривимірні сигнали (описують просторові об'єкти) [6].

Якщо область визначення сигналу неперервна, то він називається ***неперервним*** або аналоговим. Такий сигнал і його аргументи приймають будь-які значення. Сигнал, аргументи якого приймають злічену множину значень, називається ***дискретним***. Якщо ж сам сигнал приймає злічену множину значень, то він називається ***квантованим***. Цифровими називаються дискретні квантовані сигнали.

Сигнали в оптичних системах зазнають різноманітних перетворень. При цьому перетворення сигналів представляються як сукупність деяких елементарних перетворень. Оптичний прилад при цьому розглядається як каскад перетворювачів інформації, а оптична система є лінійним фільтром сигналу [6].

Кожен зображуючий прилад приймає інформацію від попереднього елемента каскаду і передає наступному. Вхідний сигнал називають предметом, а вихідний - зображенням.

Завданням зображуючого приладу є перетворення вхідного сигналу - функції предмета  в вихідний сигнал – функцію зображення . Модель оптичного приладу, що описує загальні закономірності формування зображення в оптичних системах, не пов'язані з фізичними принципами їх роботи (зовнішня функціональна модель), є оператор перетворення:



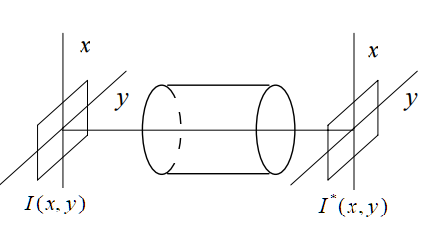


Рисунок 1.1 – Схема формування зображення

Дуже рідко зображення, одержувані в інформаційних системах, мають цифрову форму. Тому їх перетворення до цього виду є обов'язковим, якщо передбачається використання цифрової обробки, передача або зберігання. Дане перетворення включає в себе дві процедури [6].

Перша полягає в заміні безперервного кадру дискретним і зазвичай називається ***дискретизацією***, а друга виконує заміну безперервної множини значень яскравості множиною квантованих значень і носить назву ***квантування***. При цифровому поданні кожному з квантованих значень яскравості ставиться у відповідність двійкове число, чим і досягається можливість введення зображення в ЕОМ.

***Дискретизація*** - це перетворення неперервного сигналу у послідовність чисел, тобто представлення цього сигналу за будь-яким кінцевовимірним базисом [6].

Найбільш зручним з точки зору організації обробки і природнім способом дискретизації є представлення сигналів у вигляді вибірки їх значень в окремі, регулярно розташовані точки. Такий спосіб називають ***раструванням***, а послідовність вузлів, в яких беруться значення - ***растром***. Інтервал, через який беруться значення безперервного сигналу називається ***кроком дискретизації***. Зворотна до кроку величина називається ***частотою дискретизації***.

Під час цифрової обробки зображень безперервний динамічний діапазон значень яскравості ділиться на ряд дискретних рівнів. Ця процедура називається ***квантуванням***. При цьому відбувається перетворення безперервної змінної  в дискретну змінну , приймаючу кінцеву множину значень . Ці значення називаються ***рівнями квантування*** [6].

## 1.2. Цифрове зображення та його обробка

При розгляді питань, пов'язаних з моделюванням і обробкою зображень, необхідно перш за все сформулювати визначення самого поняття «зображення». Найпростіше визначення цього терміну: зображення - це те, що ми бачимо. Інше визначення: зображення - це інформація, придатна для візуального сприйняття. Залежно від походження, умовно можна виділити наступні типи зображень:

1. Намальоване або друковане;
2. Оптичне
3. Фотографічне (оптичне зображення, зареєстроване на фотоматеріалі в результаті хімічного процесу);
4. Електронне або цифрове (оптичне зображення, зареєстроване за допомогою електронного приймача, наприклад, сканера). Електронним зазвичай називають зображення, що відображається на екрані монітора.

Вочевидь, що даний поділ умовний: зображення з одного типу відразу переходить в інший. Необхідно дати зображенню формалізоване визначення, яке дозволить описати цей об'єкт математично і маніпулювати ним для досягнення певних цілей. Ці маніпуляції прийнято називати «обробка зображення» [6].

Зображення можна визначити як двовимірну функцію , де  і  - координати в просторі (конкретно на площині) і , що задається парою координат  - *інтенсивність* або рівень сірого зображення в цій точці. Якщо величини ,  і  приймають кінцеве число дискретних значень, то мова йде про ***цифрове зображення***[3].

Як було описано вище, оптичне зображення з точки зору теорії сигналів є двовимірним безперервним сигналом. У такому вигляді воно не придатне для обробки в комп'ютерних системах, а, отже, має бути перетворено. Для цього виконуються операції дискретизації (по просторовим координатам) і квантування (за інтенсивністю). Отримане цифрове зображення являє собою масив дискретних значень. Пристрій, що виконує операції дискретизації і квантування, в теорії сигналів називають АЦП (аналого-цифровий перетворювач), а виконує зворотне перетворення - ЦАП (цифро-аналоговий перетворювач) [6].

Серед характеристик цифрових зображень слід виділити:

***Розмір:*** цей параметр може бути будь-яким, але часто вибирається виходячи з особливостей реєстрації зображення, особливостей подальшої обробки і т.п.

***Кількість кольорів (глибина кольору):*** точніше кількість біт, що відводиться для зберігання кольору, кратна ступеня 2. Зображення для зберігання інформації про кольори якого необхідно 1 біт називаються бінарним. Для зберігання напівтонових зображень використовується зазвичай 8 біт. Кольорові зображення зберігаються зазвичай з використанням 24 біт по 8 на кожен з трьох колірних каналів.

***Роздільна здатність:*** вимірюється зазвичай в dpi (кількість точок на дюйм). В процесі обробки роздільну здатність можна змінювати: на саме зображення це не вплине, але зміниться його відображення пристроєм візуалізації [6].

Цифровою обробкою зображень називається обробка цифрових зображень за допомогою цифрових обчислювальних машин (комп'ютерів).

Зір є найбільш досконалим з наших органів почуттів, тому не дивно, що зорові образи грають найважливішу роль в людському сприйнятті. Однак, на відміну від людей, здатних сприймати електромагнітне випромінювання лише у видимому діапазоні, машинна обробка зображень охоплює практично весь електромагнітний спектр від гамма-випромінювань до радіохвиль. Оброблювані зображення можуть породжуватися такими джерелами, які для людини незвично пов'язувати з зображеннями. Такі, наприклад, ультразвукові зображення, зображення, одержані в електронній мікроскопії або генеруються комп'ютером. Таким чином, цифрова обробка зображень охоплює широкі і різноманітні області використання [3].

Як область цифрової обробки сигналів, цифрова обробка зображень має багато переваг перед аналогової обробкою. Вона дозволяє застосовувати набагато більш широкий ряд алгоритмів до вхідних даних та уникати проблем, таких як шуми і спотворення в процесі обробки. Оскільки зображення визначаються як двомірні (або вище), цифрова обробка зображення може бути промодельована у вигляді багатовимірних систем.

Історія цифрової обробки зображень тісно пов'язана з розвитком цифрової обчислювальної техніки. Справді, для цифрових зображень потрібна велика пам'ять і обчислювальна потужність, що прогрес в області цифрової обробки зображень в значній мірі визначається розвитком комп'ютерів і допоміжних технологій для зберігання, відображення і передачі даних [3].

Перші комп'ютери з потужністю, достатньою для виконання завдань цифрової обробки зображень, з'явилися на початку 1960-х рр. Народження того, що ми сьогодні називаємо цифровою обробкою зображень, простежується з моменту виникнення таких машин і появи програм вивчення космосу. Паралельний прогрес в цих двох областях привів в дію потужний потенціал ідей цифрової обробки зображень.

З 1960-х рр. до теперішнього часу область застосування обробки зображень значно розширилася. Крім медичних і космічних програм, методи цифрової обробки зображень сьогодні використовуються в широкому колі областей. Комп'ютеризовані процедури застосовуються для полегшення сприйняття рентгенівських і інших зображень в промисловості, медицині та біології шляхом підвищення контрасту або колірного кодування різних рівнів інтенсивності (представлення зображень в псевдокольорах). Аналогічні методи застосовуються в географії для вивчення картини забруднень навколишнього середовища за даними аерофотозйомки і космічними знімками. Методи поліпшення і відновлення зображень застосовуються при обробці неякісних зображень втрачених об'єктів або тяжко відтворених експериментальних результатів. У фізиці та суміжних областях комп'ютерна обробка є звичайним способом поліпшення якості зображень, одержуваних в ході експериментів, як, наприклад, в електронній мікроскопії або фізики високотемпературної плазми. Аналогічні приклади успішного застосування технологій обробки зображень можна знайти в астрономії, біології, медичної радіології, промисловості, в оборонній та правоохоронній сфері [3].

## 1.3. Моделі зображень та їх спотворень

Найбільш загальна схема формування спотвореного зображення представлена на малюнку.

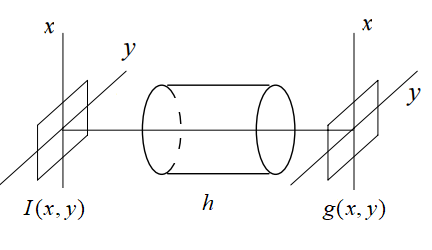


Рисунок 1.2 – Схема формування зображення

де  - вихідне зображення, що описується двома змінними ;  - спотворене зображення, сформоване з  за допомогою деякого відомого оператора спотворень (формуюча система) :

 (1.1)

Більшість формуючих систем за першого наближення можна розглядати як лінійні і інваріантні до зсуву. Зображення, сформовані такими системами, зазнають лінійні трансляційно-інваріантні спотворення, які характеризуються тим, що механізм їх виникнення один і той же для всіх точок . Лінійні спотворення виявляються в ослабленні верхніх частот вихідного зображення. Візуально це призводить до погіршення його різкості. В процесі запису зображення спотворюються також шумами, присутніми в будь-якому реальному фізичному пристрої. У ряді практично важливих випадків шум можна вважати адитивним і незалежних від вихідного зображення [2].

Модель процесу спотворення передбачає дію деякого спотворюючого оператора  на вихідне зображення , який після додавання адитивного шуму дає спотворене зображення . Задача відновлення полягає в побудові деякого наближення  вихідного зображення до заданого (спотвореного) зображення , деякої інформації щодо спотворюючого оператора  і деякої інформації щодо адитивного шуму . Щоб наближення  було якомога ближче до вихідного зображення , ми повинні володіти якомога більшою інформацією про оператор h та про функцію  [3].

### 1.3.1. Трансляційно-інваріантні спотворення

Якщо  - лінійний трансляційно-інваріантний оператор, то спотворене зображення може бути представлене у просторовій області у вигляді :

 (1.2)

 (1.3)

Де  – функція, що представляє собою спотворюючий оператор у просторовій області, а \* – використовується для позначення просторової згортки.

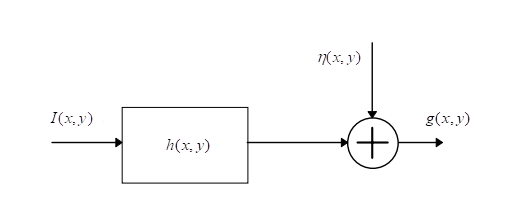


Рисунок 1.3 – Лінійна модель трансляційно-інваріантного спотворення

Таким чином, значення функції яскравості  вихідного зображення в точці з координатами  «розмазується» відповідно до виду функції розсіювання точки  і спотворюється адитивним шумом [3].

Відомо, що згортка в просторовій області аналогічна добутку в частотній області, тому рівність, що задає модель може бути еквівалентним чином записана в частотної області:

 (1.4)

Надалі будемо використовувати позначення .

Тимчасово покладемо, що , тому . Оператор Н називається ***лінійним***, якщо:

 (1.5)

де a і b – будь-які скаляри, а  і  – дві будь-які функції (два будь-яких зображення) [3].

Оператор, що діє за правилом , називається ***трансляційно-інваріантним*** (або просторово-інваріантним), якщо для будь-якої функції  і для будь-яких чисел α і β виконується рівність:

 (1.6)

Відповідно до даного означення дія оператора в точці залежить лише від значення аргументу і не залежить від місця розташування точки у просторі [3].

Запишемо функцію  в термінах безперервної імпульсної функції (δ-функції):

 (1.7)

Знову тимчасово припустимо, що . Тоді підстановка дає

  (1.8)

Нехай  – лінійний оператор. Оскільки властивість адитивності поширюється на інтеграли, то

 (1.9)

Використовуючи властивість однорідності і враховуючи, що  не залежить від x і y, одержуємо

 (1.10)

Функція під знаком інтеграла в правій частині останньої рівності

 (1.11)

називається ***імпульсним відгуком*** (імпульсною характеристикою) або ядром оператора H. Таким чином, функція  являє собою результат дії (відгук) оператора  на δ-функцію, локалізовану в точці з координатами . В оптиці, коли імпульс відповідає точці, що сяє, функцію  зазвичай називають ***функцією розсіювання точки*** (ФРТ). Походження терміну пов'язане з тією обставиною, що будь-яка реальна оптична система до деякої міри розмиває (розсіює) сяючу точку, причому величина розсіювання визначається якістю оптичної системи [3].

Маємо рівняння:

 (1.12)

яке називається ***інтегралом Фредгольма*** першого роду. В останньому виразі закладено фундаментальний результат, що лежить в основі теорії лінійних систем. Цей результат встановлює, що якщо відомий відгук системи на імпульсну функцію, то відгук системи на будь-яку функцію  може бути обчислений на основі цієї формули. Іншими словами, будь-яка лінійна система  повністю характеризується своїм імпульсним відгуком (ядром відповідного оператора).

Якщо оператор  є трансляційно-інваріантним, то з випливає, що

 (1.13)

Вираз (1.12) в цьому випадку приймає вид:

  (1.14)

Цей вираз є ***інтегралом згортки***. Вираз показує, що, знаючи ядро лінійного оператора, можна обчислити результат  його дії на будь-яку функцію . Цей результат просто являє собою згортку ядра з відповідною функцією [3].

За наявності адитивного шуму вираз, що визначає лінійну модель спотворень, набуває вигляду:

 (1.15)

Якщо оператор  трансляційно-інваріантний, то записується у вигляді

 (1.16)

Значення, що описують шум доданка , є випадковими величинами, які передбачаються незалежними від точки простору.

Отже, проведений розгляд демонструє, що вплив лінійної трансляційно-інваріантної спотворюючої системи з адитивним шумом може бути змодельований у просторовій області як згортка спотворюючої функції (ядра спотворюючого оператора) із зображенням і подальше додавання адитивного шуму. На основі теореми про згортку той самий вплив може бути виражено у частотної області як добуток Фур'є - перетворень зображення і спотворюючої функції з подальшим додаванням Фур'є перетворень шуму [3].

Лінійні трансляційно-інваріантні моделі можуть бути використані для наближеного опису багатьох типів спотворень. Перевага такого підходу полягає в тому, що величезна кількість використовуваних в лінійній теорії методів і засобів стають придатними для вирішення задач відновлення зображень. Хоча нелінійні і трансляційно-неінваріантні методи є більш загальними (і зазвичай більш точними), але їх використання часто призводить до непереборним або дуже важко вирішуваним чисельними методами проблем. Оскільки спотворення є результат згортки, то для відновлення необхідно знайти такий фільтр, застосування якого призводило б до зворотного процесу. Тому для позначення лінійного процесу відновлення часто використовується термін ***реконструкція*** (деконволюція) зображень. Аналогічно фільтри, які використовуються для відновлення, часто називаються реконструюють фільтрами [3].

### 1.3.2. Моделі шуму

Основні джерела шуму на цифровому зображенні - це сам процес його отримання, а також процес передачі. Робота сенсорів залежить від різних факторів, таких як зовнішні умови в процесі відеозйомки і якість сенсорів. Наприклад, в процесі отримання зображення за допомогою фотокамери основними факторами, що впливають на величину шуму, є рівень освітленості і температура сенсорів. В процесі передачі зображення можуть спотворюватися перешкодами, що виникають в каналах зв'язку. Наприклад, під час передачі зображення з використанням бездротового зв'язку воно може бути спотворено в результаті розряду блискавки або інших проявів в атмосфері [3].

Для подальшого розгляду важливими є параметри, що визначають просторові характеристики шуму, а також питання, чи корелює шум із зображенням. Під частотними характеристиками розуміються властивості спектра шуму в сенсі перетворення Фур'є. Наприклад, шум, спектр якого є постійною величиною, називається зазвичай ***білим шумом***. Походження цього терміну пов'язано з фізичними властивостями білого світла, який містить практично всі частоти видимого спектру в рівних пропорціях.

За винятком періодичного в просторі шуму, шум не залежить від просторових координат і не корелює із самим зображенням (тобто. між значеннями елементів зображення і значеннями шумової складової немає кореляції). Хоча в ряді випадків такі припущення щонайменше не цілком справедливі, труднощі, що виникають при розгляді просторово-залежного і корельованого шуму, лежать за межами нашого обговорення.

В рамках зроблених раніше припущень ми маємо справу з описом поведінки шуму в просторовій області, яка заснована на статистичних властивостях значень інтенсивності компоненти шуму в моделі. Ці значення яскравості можуть розглядатися як випадкові величини, які характеризуються функцією щільності розподілу ймовірностей [3].

**Гаусів шум**

Математична простота, характерна для роботи з моделями Гаусового шуму (також званого ***нормальним шумом***) як в просторовій, так і в частотній області, зумовила широке поширення цих моделей на практиці. Насправді ця простота виявляється настільки привабливою, що найчастіше гаусові моделі використовуються навіть в тих ситуаціях, коли їх застосування виправдано в кращому випадку лише частково.

Функція щільності розподілу ймовірностей гаусової випадкової величини z задається виразом:

 (1.17)

де z є значення яскравості,  – середнє значення випадкової величини z, σ - її середньоквадратичне відхилення. Квадрат середньоквадратичного відхилення  називається ***дисперсією величини*** z [3].

**Шум Релея**

Функція щільності розподілу ймовірностей шуму Релея задається виразом:

 (1.18)

Середнє і дисперсія для цього розподілу мають вигляд

 (1.19)

**Шум Ерланга (гамма-шум)**

Функція щільності розподілу ймовірностей шуму Ерланга задається виразом:

 (1.20)

де a>0, b - ціле число більше нуля і символ «!» позначає факторіал. Середнє і дисперсія для цього розподілу мають вигляд: 

Вираз (1.20) часто називають гамма-розподілом, хоча, строго кажучи, ця назва відноситься до розподілу більш загального вигляду, коли b не є цілим, а в знаменнику стоїть гамма-функція Γ(b). Розглянутий окремий випадок правильніше називати розподілом Ерланга [3].

**Експоненційний шум**

Функція щільності розподілу ймовірностей експоненціального шуму задається виразом:

 (1.21)

де a>0. Середнє і дисперсія для цього розподілу мають вигляд: 

Зауважимо, що цей розподіл є окремим випадком розподілу Ерланга коли b = 1.

**Рівномірний шум**

Функція щільності розподілу ймовірностей рівномірного шуму задається виразом:

 (1.22)

Середнє значення для цього розподілу дорівнює , а дисперсія 

**Імпульсний шум**

Функція щільності розподілу ймовірностей (біполярного) імпульсного шуму задається виразом:

 (1.23)

Якщо b > a, то піксель з яскравістю b виглядає як світла точка на зображенні. Піксель з яскравістю a виглядає, навпаки, як темна крапка. Якщо одне із значень ймовірності ( або ) дорівнює нулю, то імпульсний шум називається ***уніполярним***. Якщо жодна з ймовірностей не дорівнює нулю і особливо якщо вони приблизно рівні за величиною, імпульсний шум схожий на крупиці солі і перцю, випадково розсипані по зображенню. З цієї причини імпульсний шум називають також шумом типу «сіль і перець». Також для позначення цього типу шумів використовуються терміни шум випадіння і піковий шум [3].

Значення імпульсів шуму можуть бути як позитивні, так і негативні. При відцифруванні зображення зазвичай відбувається масштабування (і обмеження) значень яскравості. Оскільки величина пов'язаних з імпульсним шумом спотворень, як правило, велика в порівнянні з величиною корисного сигналу, імпульсний шум після відцифрування приймає екстремальні значення, що відповідає появі абсолютно чорних і білих точок на зображенні. Тому зазвичай передбачається, що значення a і b є «Інтенсивними» в тому сенсі, що вони рівні мінімальному і максимальному значенням, які в принципі можуть бути присутніми в відцифрованному зображенні. В результаті негативні імпульси виглядають як чорні точки на зображенні (перець). З тих же причин позитивні імпульси виглядають як білі крапки (сіль). Для 8-бітових зображень це зазвичай означає, що a = 0 (чорне) і b = 255 (біле).

Розглянуті розподіли в сукупності представляють собою набір засобів, які дозволяють моделювати спотворення, пов'язані з широким діапазоном шумів, що зустрічаються на практиці. Так наприклад, гаусів шум виникає на зображенні в результаті впливу таких факторів, як шум в електронних ланцюгах, а шум сенсорів - через нестачу освітлення і / або високої температури. Розподіл Рейлі корисний при моделюванні шуму, який виникає на знімках, знятих з великої відстані. Експоненціальний і гамма-розподіл відповідають шуму на зображеннях, отриманих з використанням лазерів. З імпульсним шумом ми стикаємося в ситуаціях, коли під час відцифрування зображення через перешкоди в мережі живлення виникають перехідні процеси, що призводять до появи екстремальних значень. Рівномірний розподіл, мабуть, в найменшій мірі підходить для опису явищ, що зустрічаються на практиці. Однак цей розподіл дуже корисний як основа для створення різних генераторів випадкових чисел [3].

### 1.3.3. Спотворення зображення в наслідок змазу

Змазами називатимемо такий тип спотворення, який виникає внаслідок динамічних змін об’єктів уваги чи фону під час експонування кадру.

Відомі методи усунення змазів ґрунтуються на математичній моделі змазу, що основана на природі й особливостях спотворення та описує зміни, що відбуваються із кольорами точок зображення при змазі. Розглядаючи тут поняття змазу, маємо на увазі глобальне спотворення, а не локальне, яке є його складнішою формою [7].

Хоч існує декілька достатньо ефективних алгоритмів реконструкції зображень, спотворених змазом, не існує способу відтворити таке зображення з нульовими втратами інформативності. Величина цих втрат є мірою ефективності алгоритму реконструкції.

Для того, щоб зрозуміти причину цих втрат, насамперед потрібно перевірити правильність вибраної моделі змазу зображення. Типово ця перевірка повинна здійснюватися процедурою моделювання змазу на неспотвореному зображенні. Для цього зображення штучно спотворюється деяким рухом (переміщенням) з наперед відомими параметрами (траєкторією та рівномірністю). Надалі за наперед визначеною метрикою проводиться порівняння із зображенням, спотвореним природно – під час експонування рухомого об’єкта пристроєм зі світлочутливою матрицею [7].

Сутність змазу, подібно до спотворення від розфокусування оптичної системи, полягає в тому, що інформація про колір кожної точки перерозподіляється по усьому зображенню за деяким законом. Різниця між різними видами спотворень полягає саме у законі, за яким відбувається цей перерозподіл. Цей закон визначається функцією розсіювання точки.

У загальному випадку значення функції інтенсивності у точці зони змазу отримує деякий приріст кольору. Зокрема, у разі горизонтального змазу на один піксель, внаслідок спотворення, значення кожного пікселя додається зі значенням попереднього за координатою (лівого) пікселя та ділиться на два:

 (1.24)

Ця формула випливає з таких міркувань: оскільки лівий піксель насувається під час руху на заданий, то за час експонування обидва значення рівномірно встигли відтворитись у цій позиції [7].

В результаті отримуємо нове спотворене зображення:

 (1.25)

Це є моделлю утворення ідеального спотворення.

Наведений вище аналіз змазу стосувався загального випадку, тобто без огляду на його тип. Класифікація передбачає існування глобального (повного) та локального (часткового) змазів. Останній є складнішим випадком за перший тип змазу. Тому зупинимось детальніше на моделі утворення саме локального змазу, тобто на такому змазі, який охоплює не усе зображення, а лише його частину. При цьому решта зображення лишається неспотвореним.

На відміну від повного змазу, який утворюється під час руху камери і за нерухомого фону, частковий має різні механізми утворення. Відповідно до цього треба було б розглядати різні моделі спотворень [7].

Існує декілька типів локальних змазів. Зокрема такий тип, який утворюється, коли нерухома камера, що експонує кадр, в якому, своєю чергою, наявний об’єкт, що в цей момент швидко рухається відносно власного фону.

Схема утворення такої розмитої ділянки є такою. У момент  затвор закривається. За цей проміжок часу рухомий об’єкт перемістився на деяку відстань, яку, враховуючи дискретну природу цифрової фотографії, можна оцінити скінченною кількістю пікселів m. Це дає можливість проміжок часу  розділити на m рівних відрізків. Протягом кожного з цих відрізків часу кожен піксель рухомого об’єкта залишав відбиток власного значення функції інтенсивності в іншій точці зображення, накладаючи власне значення функції інтенсивності на той, який існував у цій точці позиції раніше. У результаті значення функції інтенсивності у внутрішніх пікселях об’єкта накладаються на значення інших пікселів цього ж об’єкта і виникає класична задача деконволюції.

На краях зони розмиття ситуація є дещо іншою. Краї можна визначити як область вздовж периметра об’єкта по обидві сторони в напрямку руху шириною m точок (буферна зона). У цій зоні відбувається змішування значень функції інтенсивності рухомого об’єкта зі значеннями функції інтенсивності пікселів нерухомого фону. Оскільки час експонування дискретизовано на m періодів, то можна вважати, що за одиницю часу  значення функції інтенсивності кожної точки буферної зони формується за рахунок -ї частини значення кольору рухомого об’єкта та (1-)-ї частини решти значення, яке, своєю чергою, утворюється в результаті адитивного накладання у тій самій пропорції значень функції інтенсивності, що належать об’єкту та фону, над яким цей об’єкт рухався.

Для граничного пікселя об’єкта значення функції інтенсивності визначатиметься за описаним співвідношенням. Для наступного пікселя об’єкта його значення функції інтенсивності *с*(*xi*, *yi*) визначатиметься як результат скалярного добутку векторів  і 

 (1.26)

де  – значення функції інтенсивності відповідно фону, у заданому та у граничному пікселях. За подібною схемою визначатимуться значення функції інтенсивності у кожній точці буферної зони. Останній піксель буферної зони матиме значення функції інтенсивності, яке лише на величину  складатиметься зі значення кольору фону [7].

Описаний підхід формування значень функції інтенсивності у пікселях буферної зони ***називатимемо операцією зважування значень функції інтенсивності***, а сам вектор **v** – ***оператором зважування***. Треба зазначити, що вказані вище параметри оператора **v** можливі лише за умови рівномірного руху. У випадку нерівномірного руху ці параметри будуть іншими і формуватимуться за правилом: що менший час піксель об’єкта перебував в заданій позиції, то меншою буде його частка значення функції інтенсивності в результуючому значенні.

Проте закономірність, отримана для випадку рівномірного руху, зберігатиметься і для *j*-го пікселя буферної зони і може бути записана так:

  (1.27)

де *j* є [0; *m*]; *hi* – *i*-те ненульове значення дискретної функції розсіяння точки *h*; *bj* – значення функції інтенсивності фону в цій точці; *fj* – інтегральне значення функції інтенсивності у пікселях рухомого об’єкта, які перебували в заданій позиції під час руху об’єкта.

Інакше кажучи, конфігурація та розмірність матриці ФРТ залежать від швидкості, рівномірності, траєкторії руху та від часу експонування кадру. Розміщення ненульових елементів матриці ФРТ повторює траєкторію руху об’єкта за той час, поки світлочутлива матриця експонувала заданий кадр. Значення елементів матриці пропорційні до швидкості руху об’єкта у проміжок часу від *tn* до *tn+1*, який дорівнює 1/*m* часу експонування кадру, де *m* – кількість ненульових елементів матриці. Відповідно, за рівномірного руху ненульові елементи матриці будуть приблизно рівними між собою, а при прямолінійному вибудуваними в лінію. Строгий горизонтальний рух породжує вектор рядок, а строго вертикальний – вектор-стовпець.

Звідси можна зробити висновок, що досліджувана буферна зона повторює конфігурацію ФРТ: якщо ФРТ є вектор-стовпцем розмірністю 5 елементів, то буферна зона матиме ширину 5 точок строго зверху та знизу зображення рухомого предмета.

### 1.3.4. Спотворення зображень апаратною функцією

Суть цього методу полягає в тому, що на наше зображення накладається якесь спотворення у вигляді лакуни.

***Лакуна*** - це загублена або вирізана спеціально частина зображення, яка призводить до втрати інформації про вихідне зображення.

Одним із завдань цифрової обробки є заповнення або відновлення відсутніх частин зображення (лакун). Поява лакун може бути обумовленою свідомим вилученням частини зображення. У цьому випадку метою заповнення лакуни є приховування факту вилучення. В іншому випаду лакуни на зображенні можуть утворитися при затіненні розфокусованих або змазаних об'єктів, що нас цікавлять.

Якщо відомо, який об'єкт повинен бути на місці лакуни, то метою заповнення лакуни може бути з'ясування орієнтації, розміру і кількості цих об'єктів. При такій постановці задачі після реконструкції вмісту лакун до всього зображення застосовують відомі методи деконволюції. В цьому і деяких інших випадках при заповненні лакун ефективна інтерполяція - лінійна, вагова або сплайнова. Як правило, інтерполяції зображення доцільно застосовувати при малому поперечному розмірі лакун [9].

У випадку, коли немає інформації про вміст втраченої частини зображення, крім решти зображення. Одним з варіантів вирішення такої задачі може бути використання спектра зображення. Тому пропонується застосувати метод інтерполяції послідовно обчислюваного спектру Фур'є (МІПОС), що дозволяє ретушувати і відновлювати відсутні (затінені) частини зображення. На відміну від деяких традиційних методів інтерполяцій МІПОС можна використовувати при будь-якій формі відсутньої частини зображення. Ретушування та відновлення МІПОС зображення може дати хороші результати навіть при значній частці відсутнього зображення. Запропонована методика не вимагає ніякої апріорної інформації про те, яким було реконструйоване зображення до пошкодження. Інформація береться безпосередньо з вихідного спектра пошкодженого об'єкта. Заповнення відсутньої частини зображення робиться ітераційно (від нульової просторової частоти до найвищої просторової частоти) на основі спектра присутньої частини зображення [9].

Ідеологія МІПОС коротко може бути представлена в такий спосіб. Повне зображення  можна записати як суму зображень поза лакуни  і всередині лакуни :

  (1.28)

Нехай маска, за допомогою якої можна з повного зображення отримати зображення з лакуною, позначена як L, і ця маска всередині лакуни дорівнює одиниці, а поза лакуни дорівнює нулю.

Тоді (1.3.4.1) можна переписати так:

  (1.29)

Під спектром зображення розуміється Фур'є-образ цього зображення, отриманий в результаті виконання дискретного перетворення Фур'є. Знайдемо спектр (1.3.4.2) і отримаємо:

 (1.30)

де  - дельта-функція,  - спектр маски L для отримання лакуни;  - спектр повного зображення;  - спектр зображення з лакуною, знак (\*) позначає операцію згортки [9].

# 2. МОДЕЛІ ТА МЕТОДИ ВІДНОВЛЕННЯ ЗОБРАЖЕНЬ

***Відновлення зображень*** - це процес реконструкції або відновлення зображення, яке було погіршено внаслідок якогось явища деградації. Методи реставрації в основному моделюють деградацію та застосовують зворотний процес для відновлення вихідного зображення.

З появою такої задачі, як спотворення зображень, виникла проблема їх відновлення. Задача відновлення спотвореного зображення не є простою і включає у себе два основні підходи для її вирішення:

– методи обробки в просторовій області (просторові методи), засновані на прямому маніпулюванні пікселями зображення;

– методи обробки в частотній області (частотні методи), засновані на модифікації (фільтрації) сигналу, який формується шляхом застосування до зображення перетворення Фур’є.

Просторова обробка застосовується, коли єдиним джерелом викривлень є адитивний шум. Частотна фільтрація може використовуватися для нечітких зображень з дефектами освітлення, вона також враховує і шум, тому частотна обробка є найбільш універсальним і поширеним методом поліпшення якості цифрового зображення.

З розробкою алгоритмів реконструкції зображень в загублених областях пов'язано ряд складнощів. По-перше, такі алгоритми повинні виробляти якісне текстурне і структурне заповнення відсутніх областей. По-друге, бажано, щоб час роботи таких алгоритмів було відносно невеликим. По-третє, такі алгоритми повинні бути стійкими до зміни різних функціональних умов. І, по-четверте, нетривіальним завданням подається кількісна оцінка якості результатів відновлення з точки зору візуального сприйняття [5].

Далі ми трохи докладніше розглянемо суть завдання реконструкції зображень в загублених областях, а також пов'язані з нею завдання і труднощі, що виникають при їх вирішенні.

## 2.1. Методи обробки зображень у частотній області

Суть цього методу полягає у поданні зображення, як двовимірної функції , де х і y - координати у просторі (конкретно, на площині). Значення  в будь-якій точці, заданої парою координат (х, у), називається ***інтенсивністю***, або рівнем сірого в цій точці.

Загальновідомим є твердження, що будь-яка функція, яка періодично повторює свої значення, може бути представлена у вигляді суми синусів і косинусів різних частот, помножених на деякі коефіцієнти. Таке уявлення функції називається представленням у вигляді ряду Фур'є. Коли функція не є періодичною, а площа під її графіком є кінцевою, то це - перетворення Фур'є.

Функція, задана як рядом, так і перетворенням Фур'є, може бути повністю без втрати інформації відновлена за допомогою алгоритму перетворення. Ця властивість є надзвичайно важливою, оскільки дозволяє працювати в «Фур'є-просторі», а потім повернутися в початкову область визначення функції без втрати будь-якої інформації [1].

На рис. 2.1.а) зображена складна функція, яка є сумою чотирьох синусоїд і косинусоїд рис. 2.1.б).

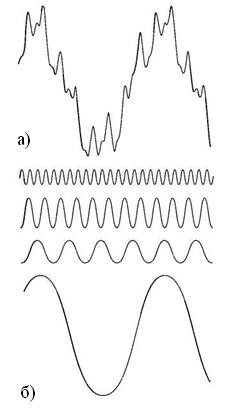


Рисунок 2.1 – Розпад функції на складові: а) функція; б) її складові

Оскільки цифрові зображення описуються двомірними дискретними функціями, то розглянемо дискретне перетворення Фур'є (ДПФ) саме для таких функцій [1].

Нехай , при  і , позначає зображення М×N. Двомірне дискретне перетворення Фур'є зображення , яке відбивається у , задається рівнянням:

  (2.1)

де  і ; M і N - парні числа.

Координатна система, що задає аргументи  частотними змінними u і ν, називається ***частотною областю***. В даному випадку можна виявити аналогію із завданням аргументів  просторовими змінними х і y. Прямокутну область розміру М×N при  і , прийнято називати ***частотним прямокутником***. Він має ті ж розміри, що і початкове зображення [1].

Навіть якщо зображення  дійсне, його перетворення Фур'є є, як правило, комплексним. Основний метод візуального аналізу цього перетворення полягає в обчисленні його спектру (тобто абсолютної величини ) і його відображення на дисплеї. Нехай  і  позначають дійсну і уявну компоненти (u, ν), тоді спектр Фур'є задається виразом:

 (2.2)

Фур'є-перетворення дійсної функції є симетрично зв'язаних; це означає, що її спектр має центральну симетрію:

 (2.3)

Кожен елемент Фур'є-образу  містить всі відліки функції , помножені на значення експоненційних членів, тому зазвичай неможливо встановити пряму відповідність між характерними деталями зображення і його образом. Однак, деякі загальні твердження щодо взаємозв'язку частотних складових Фур'є-образу і просторових характеристик зображення можуть бути зроблені. Наприклад, оскільки частота прямо пов'язана зі швидкістю зміни сигналу, то зрозуміло, що частоти в перетвореннях Фур'є пов'язані з варіацією яскравості на зображенні. Найбільш повільно змінна (постійна) частотна складова (u = ν = 0) збігається з середньою яскравістю зображення. Низькі частоти, відповідні точкам поблизу початку координат Фур'є перетворення, відповідають повільно змінним компонентам зображення. На зображенні кімнати, наприклад, вони можуть відповідати плавним змінам яскравості стін і підлоги. У міру віддалення від початку координат вищі частоти починають відповідати все більшим змінам яскравості деталей зображення і їх граней [1].

Процедура алгоритму фільтрації в частотній області проста і складається з таких кроків [1]:

*Крок 1.* Початкове зображення множиться на . Це робиться для того, щоб Фур’є-перетворення виявилося центрованим, тобто початок координат для образу функції буде знаходитися у центрі частотного прямокутника у точці :

 (2.4)

*Крок 2.* Обчислюється пряме дискретне перетворення Фур’є  зображення, отримане після кроку 1.

*Крок 3.* Функція  множиться на деяку функцію фільтру .

*Крок 4.* Обчислюється зворотне дискретне перетворення Фур’є від результату кроку 3.

*Крок 5.* Виділяється потрібна частка результату кроку 4.

*Крок 6.* Результат кроку 5 множиться на 

Причина, за якою множник  називається фільтром (часто вживається також термін передавальна функція фільтра), полягає в тому, що він пригнічує деякі «зайві» частоти перетворення, залишаючи при цьому інші майже без зміни. Питання знаходження передавальної функції фільтра і є ключовим, оскільки воно визначає метод фільтрації і вказує, які саме частоти будуть фільтруватися [4].

Нехай  позначає початкове зображення після кроку 1, а  – його Фур’є-образ. Тоді Фур’є-образ вихідного зображення визначається виразом:

 (2.5)

Множення функцій двох змінних Н і F здійснюється поелементно. Фільтроване зображення отримують шляхом обчислення зворотного Фур’є-перетворення від Фур’є-образу , обчислюючи за формулою:

Покращене зображення =  (2.6)

Шукане зображення отримаємо виділенням дійсної частини з останнього результату і множенням на , щоб компенсувати ефект від множення вхідного зображення на ту саму величину.

Описана процедура алгоритму фільтрації, схематично зображена на рис. 2.2 в більш загальному вигляді, включаючи стадії попередньої і завершальної обробки.

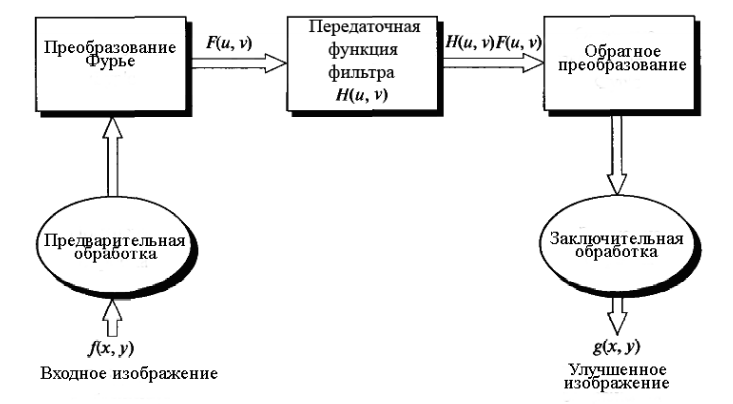


Рисунок 2.2 – Обробка зображення у частотній області

## 2.2. Алгоритм відновлення зображення у частотній області

Розглянемо алгоритм відновлення зображень, спотворених оператором Н, просторовий образ якого (спотворююча функція h) заданий або визначений за допомогою апаратної функції [12].

Існує багато алгоритмів відновлення зображень у частотній області. Але всі вони так чи інакше пов’язані з обчисленням спектру Фур’є зображення. Методи, засновані на перетворенні Фур'є, є найбільш природними і потужними способами для вирішення поставленої задачі. Перевагою даного методу є можливість аналітично оцінити детальність одержуваного зображення, а також порівняно невеликий обсяг обчислень.

Перетворення Фур'є - це рівняння, яке перетворює нормальні значення пікселів в комплексні числа. Але щоб знати, що означають ці комплексні числа, ми повинні дати їм трохи більше контексту, пояснивши, що насправді перетворення Фур'є робить з даними [12].

Коли ми перетворюємо деякі дані за допомогою перетворення Фур'є, ми перетворюємо їх у частотну область. І так сталося, що ми можемо розділити хвилі складної частоти на ряд простих синусоїд.

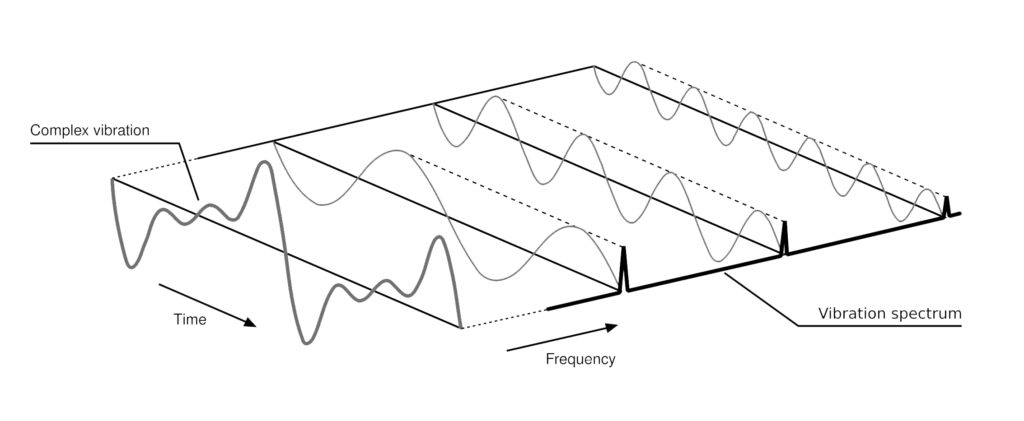


Рисунок 2.3 – Розпад хвилі складної частоти на ряд простих синусоїд

Перетворення Фур'є масиву даних буде виглядати як одна велика хвиля посередині з менше і менше хвилями, що поширюються в обох напрямках.

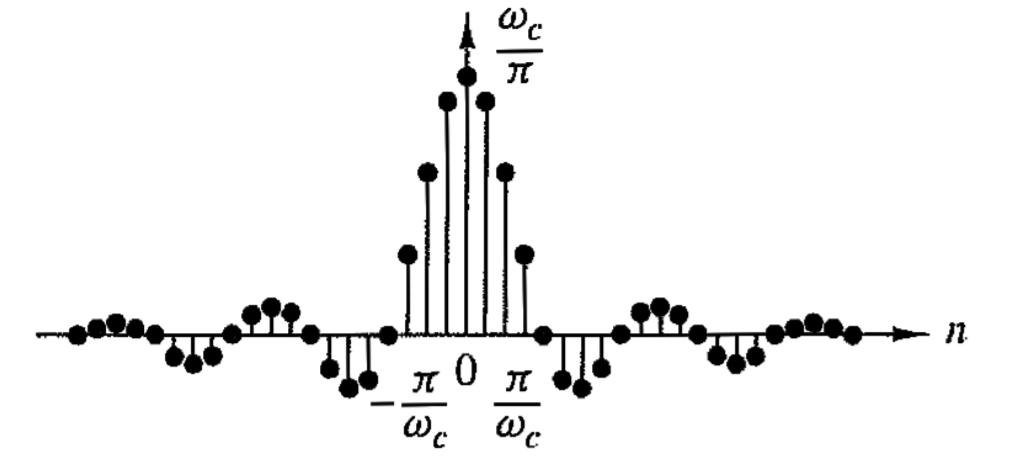


Рисунок 2.4 – Уявлення перетворення Фур’є

Структура та значення частот важливі, оскільки ми можемо використовувати зворотне перетворення Фур'є, щоб повернутися назад у вихідну форму. Звичайним (вихідним) зображенням ми називаємо просторову область.

Мета перетворення наших даних у частотну область полягає в тому, щоб ми могли застосувати зміни до набору частот для обробки наших даних в просторової області [12].

Якщо ми хочемо використовувати перетворення для зображень, нам потрібно використовувати розширене рівняння, яке дозволяє нам обчислювати дані у двох вимірах.

Давайте спочатку розглянемо формули, які дозволяють нам обчислювати тільки в одновимірному просторі.

 (2.7)

Той, що вище, прямий метод, який перетворює наші дані в частотну область, а нижче - зворотний, який повертає їх в просторову область.

Тепер перейдемо до прямого перетворення Фур’є у двовимірному просторі:

 (2.8)

де M, N - розміри, ширина і висота зображення, m, n - положення пікселя на зображенні, положення по горизонталі і вертикалі в «сітці» пікселів, x, y - частотне положення на карті комплексних чисел, e - постійна Ейлера, j - уявна константа комплексних чисел, квадратний корінь з -1 [12].

Видно, що значення M і N однакові для обох формул, що означає, що частотна карта буде того ж розміру, що і зображення.

Використання наведених вище формул в теорії буде працювати чудово. Але ми зіткнемося з проблемами, коли спробуємо реалізувати їх на практиці. Адже, чим більше зображення, тим довше і складніше буде обробити наше фото комп’ютеру. На допомогу у цій ситуації приходить **Швидке Перетворення Фур’є** (ШПФ) [12].

ШПФ розділяє наш набір даних на парні та непарні позиції індексу та обчислює їх окремо, а потім складає. Більш того, ми можемо створити в нашому коді рекурсивну структуру для повторного використання вже обчислених значень, що значно скорочує час обробки.

Швидке перетворення Фур’є відрізняється від класичного перетворення тим, що обчислення координат кожного виміру відбувається окремо. Надалі у коді буде наведено реалізацію, де видно, що ми спочатку обчислюємо стовпці, потім транспонуємо дані і обчислюємо рядки.

 (2.9)

Першим кроком, який потрібно зробити, є перетворення значень пікселів нашого зображення в комплексну карту чисел, беручи значення пікселя з каналу і перетворюючи його як справжній компонент з уявним компонентом, що прямує до 0.

Далі, отримавши нашу карту комплексних чисел, ми можемо переходити безпосередньо до перетворення Фур’є. Спочатку обчислюємо один масив значень. Рекурсивні кроки, які робить функція, обчислюють значення, як показано на зображенні нижче.

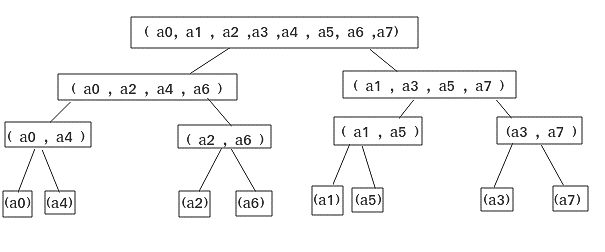


Рисунок 2.5 – Рекурсивний прохід функції по значеннях

Тепер, коли ми знаємо, як обчислити один масив чисел, переходимо за алгоритмом до обчислення цілого зображення. Як результат, ми отримаємо перетворення нашого зображення у Фур’є образ [12].

За допомогою зворотного перетворення Фур’є, будемо повертатися до нашого вихідного зображення. Але ми не можемо використовувати саме зображення, щоб перетворити його назад у просторове. Для візуалізації перетворення нам потрібно отримати на карті величину або модуль комплексних чисел. Нам також потрібно використовувати логарифмічне перетворення, щоб зробити його, тобто зображення, видимим.

Якщо ми будемо використовувати як результат спряжене комплексних чисел з функцією прямого перетворення, то отримаємо значення пікселів, отримавши їх величину.

Хоча обчислення виконується значно швидше, ніж версія DFT (дискретна трансформація Фур'є), воно також має деякі обмеження. По-перше, зображення має бути напівтоновим, адже всі обчислення виконуються з одного каналу. По-друге, зображення, яке ми намагаємось обробити, має бути обов’язково квадратним. Більш того, кожен вимір повинен мати розмір 2n , причому n є натуральним цілим числом.

Щоб обійти це обмеження, будемо використовувати нульові відступи навколо зображення, щоб зробити його квадратним. Надалі розробимо таку функцію.

Як модель спотворення оберемо апаратну функцію, яка «вирізьблює» частину зображення, тобто створює лакуну. Фактично алгоритм полягає у спотворенні пікселів певної області зображення. Ця область стає прозорою, а уся інформація про «вирізьблену» частину забувається [11].

У даному випадку перетворення Фур’є та обчислення його спектру легко допоможе відновити початкове зображення. Для успішного відновлення спотвореного зображення нам необхідно так ретушувати відсутню частину, щоб спектр отриманого зображення не відрізнявся від початкового зображення. Тобто заповнення відсутньої частини зображення потрібно робити на основі спектра присутньої частини зображення. Зробимо це наступним чином:

Для зручності позначимо спотворене зображення А, а відновлене на кожній ітерації . Обчислимо спектр Фур'є, але на першій ітерації будемо використовувати тільки 0 складову для заповнення відсутньої частини зображення. Отримаємо перше наближення. Далі обчислюємо спектр отриманого зображення  і заповнюємо відсутню частину вихідного зображення А зображенням, обчисленим з 0 і 1 складової спектра зображення . Продовжимо це робити аж до максимальних частот, присутніх на зображенні А. Порівняння вихідного і відновленого зображень показує, що поступово ретушована область буде заповнюватися зображенням, все більш відповідним до навколишньому зображенню.

## 2.3. Повний алгоритм вирішення задачі про відновлення зображення за допомогою спектру Фур’є

Для наочності продемонструємо послідовність кроків вирішення задачі про відновлення спотвореного апаратною функцією зображення за допомогою спектру Фур’є.

1. Отримаємо на вхід наше зображення у форматі JPG/PNG/GIF.
2. Перетворюємо його у напівтонове. Це потрібно для того, щоб швидше та більш простим способом обчислити спектр Фур’є зображення.
3. Адаптуємо зображення до квадратного розміру. Алгоритм працює наступним чином:
4. Знаходить і встановлює правильну розмірність за першим числом , що більше, ніж розмір нашого зображення;
5. Копіює значення пікселів нашого зображення в масив;
6. Копіює масив зображення у масив з усіма нульовими значеннями, за винятком альфа-каналу, де ми встановлюємо для нього максимальне значення.
7. Обчислюємо спектр Фур’є напівтонового квадратного зображення за формулою (2.9).
8. Застосовуємо до нашого вихідного зображення спотворюючу функцію. Як результат маємо на зображенні спотворену прямокутну область. Алгоритм спотворення зображення працює наступним чином:
9. На вхід отримаємо наше вихідне зображення;
10. Копіює значення пікселів нашого зображення у масив байтів.
11. За допомогою функції Random() генеруємо довільну прямокутну область, що спотворює зображення.
12. Проходимо циклом по всій ширині та висоті зображення і перевіряємо чи співпадають координати точки з заздалегідь згенерованими.
13. Якщо так, то спотворюємо цей піксель тим, що робимо його прозорим, фактично стираючи всю інформацію про колір. Інакше пропускаємо його. Наприкінці ми маємо отримати прозору область на зображенні, інформація про вміст якої загублена.
14. Обчислюємо спектр Фур’є нашого пошкодженого зображення за формулою (2.9).
15. Застосовуємо алгоритм порівняння двох зображень для визначення «розбитих» пікселів. Беремо спектр Фур’є вихідного та пошкодженого зображень і проходимо по всім складовим, виокремлюючи ті сигнали, що відрізняються.
16. На першій ітерації заповнюємо відсутню частину пошкодженого зображення спектром вихідного зображення з нульовою складовою. Відкидаємо спектр повного зображення, тепер він нам не потрібен. Тепер ми маємо спотворене зображення (позначимо його, як А) та зображення, отримане після першої ітерації, з нульовим заповненням (позначимо його, як ). Обчислюємо спектр цього зображення  для 0 та 1 складової. Заповнюємо спотворене зображення А цим спектром, перетворюємо у зображення (отримаємо ). Обчислюємо спектр  для 0, 1 і 2 складових, заповнюємо ним попереднє зображення . Виконуємо такий алгоритм до максимальних частот, присутніх на спотвореному зображенні. Як результат, ми повинні отримати деяке зображення , яке буде максимально наближене до нашого вихідного зображення.
17. За допомогою зворотного перетворення Фур’є повертаємось до просторової площини та отримаємо наше відновлене зображення.

Алгоритм можна також зобразити у вигляді блок – схеми:

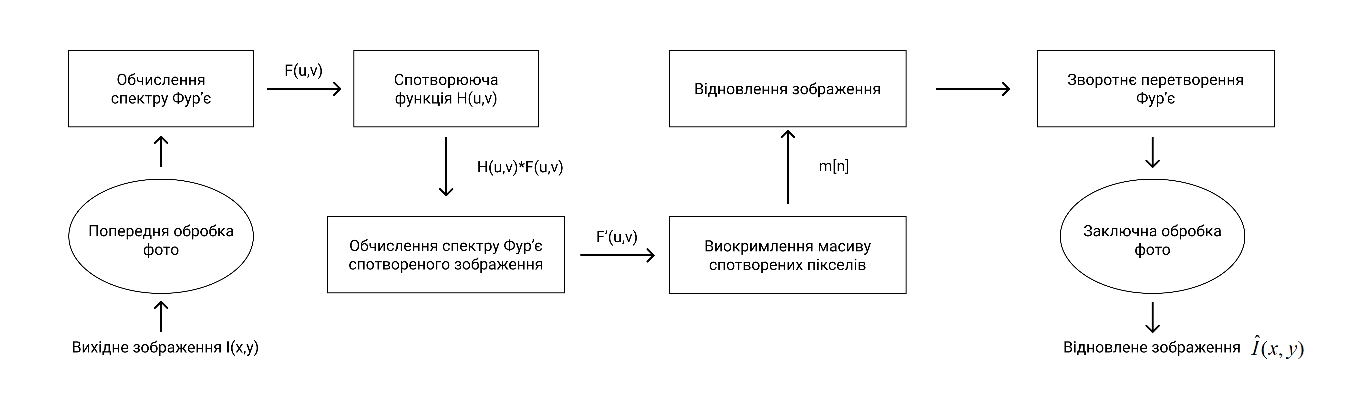


Рисунок 2.6 – Алгоритм спотворення та відновлення зображення

# 3. ОПИС РОЗРОБЛЕНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

## 3.1. Розбір використаних технологій та графічних інтерфейсів

Для виконання дипломної роботи було використано мову С# на базі фреймворку .Net. Графічний інтерфейс виконано за допомогою Windows Forms.

На сьогоднішній день мова програмування C# одна з найпотужніших, швидко розвивається і затребувана мова в ІТ-галузі. На даний момент на ній пишуться найрізноманітніші програми: від невеликих десктопних програм до великих веб-порталів і веб-сервісів, які обслуговують щодня мільйони користувачів [13].

У порівнянні з іншими мовами C# досить молодий, але в той же час він вже пройшов великий шлях. Перша версія мови вийшла разом з релізом Microsoft Visual Studio .NET в лютому 2002 року. Поточною версією мови є версія C# 9.0, яка вийшла в 10 листопада 2020 року.

C# є мовою з Сі-подібним синтаксисом і близький в цьому відношенні до C ++ і Java.

C# є об'єктно-орієнтованим і в цьому плані багато перейняв у Java і С ++. Об'єктно-орієнтований підхід дозволяє вирішити завдання з побудови великих, але в той же час гнучких, масштабованих і розширюваних додатків. І C# продовжує активно розвиватися, і з кожною новою версією з'являється все більше цікавого функціоналу, як, наприклад, лямбда, динамічне зв'язування, асинхронні методи і т.п. [13].

Windows Forms – інтерфейс програмування додатків (API), відповідальний за графічний інтерфейс користувача і є частиною Microsoft .NET Framework. Даний інтерфейс спрощує доступ до елементів інтерфейсу Microsoft Windows за допомогою створення обгортки для Win32 API в керованому коді. Всередині .NET Framework, Windows Forms реалізується в межах простору імен System.Windows.Forms [14].

Для обробки зображень було обрано ***Bitmap***. Простір ***System.Drawing.Bitmap*** дуже корисний в інфраструктурі .NET, тому що дозволяє зчитувати і зберігати файли різних графічних форматів.

Також цей простір імен забезпечує доступ до функціональних можливостей графічного інтерфейсу GDI+, використовуючи близько 50 класів, в тому числі до класів ***Graphics*** та ***Image***. Клас ***Graphics*** надає методи малювання на пристрої відображення. Клас **Image** - абстрактний базовий клас, який надає функціональні можливості для похідних класів ***Bitmap***.

## 3.2. Розбір основних моментів

Графічний інтерфейс програми надає користувачеві такі можливості:

1. Завантаження фото у форматі JPG/PNG/GIF;
2. Спостерігання вихідного зображення;
3. Спостерігання спотвореного зображення;
4. Спостерігання спектрів Фур’є вихідного та спотвореного зображень;
5. Накладання спотворення на зображення;
6. Відновлення зображення;
7. Зберігання готового зображення.

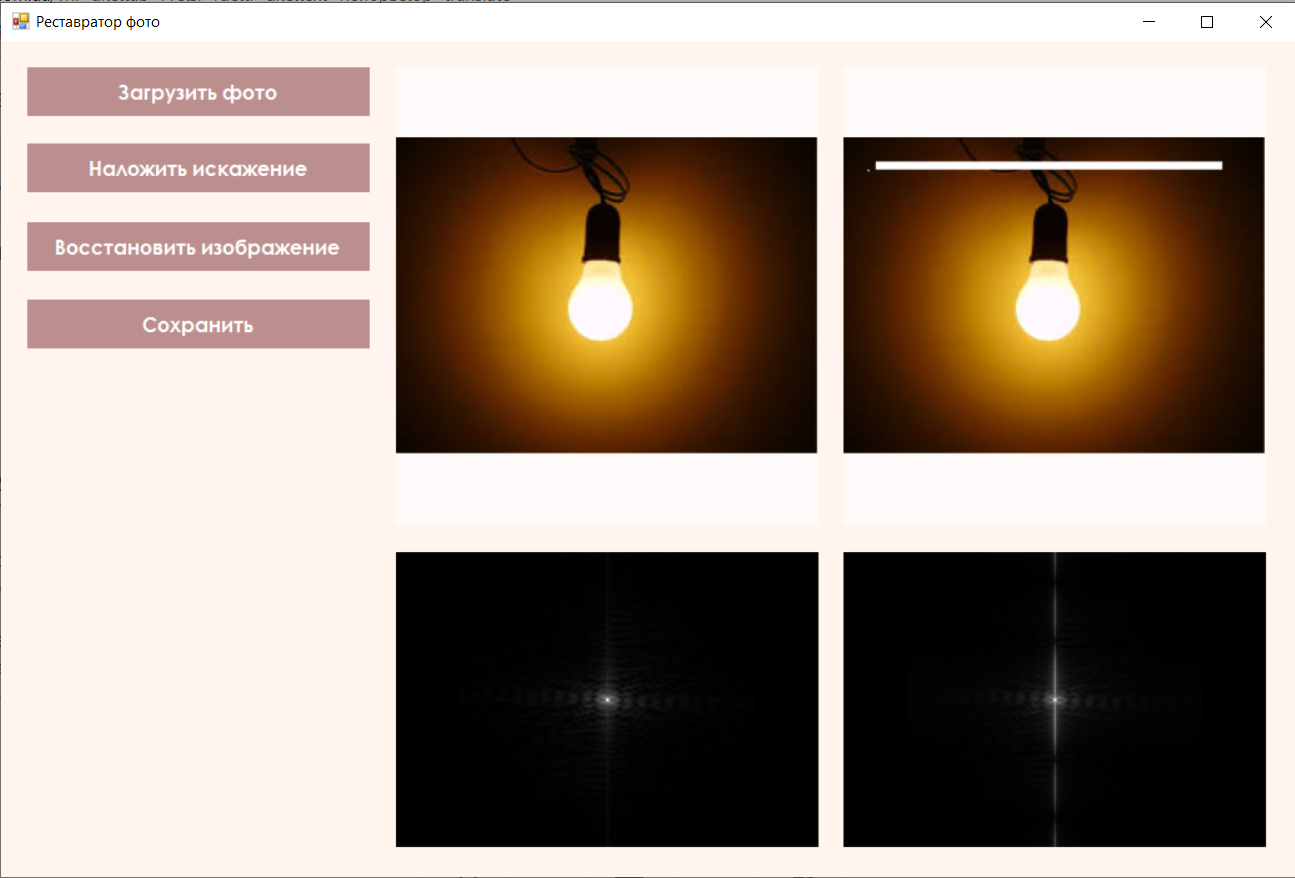


Рисунок 3.1 – Інтерфейс програми

Розглянемо процес завантаження зображення:

Завантаження відбувається за допомогою елементу ***OpenFileDialog***. Перевіряємо, чи дійсно обрано файл, якщо так – запам’ятовуємо шлях до файлу, створюємо Bitmap об’єкт, зберігаємо його розмір, адже він згодом нам знадобиться і завантажуємо наше зображення у елемент відображення PictureBox, інакше – виводимо сповіщення про помилку. Далі викликаємо функцію перетворення зображення на квадратне та напівтонове. Обчислюємо спектр Фур’є зображення та виводимо його також на екран.

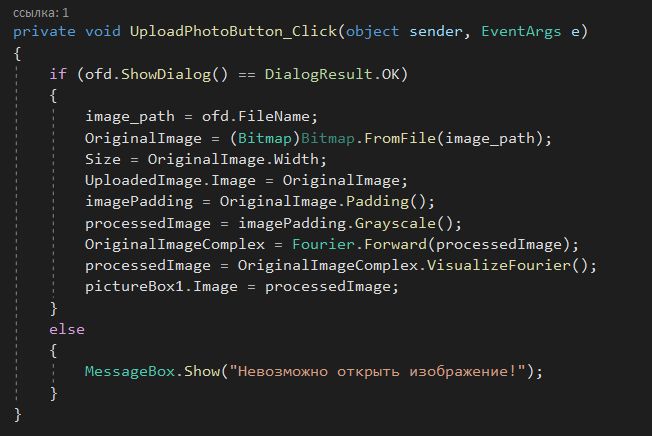


Рисунок 3.2 – Завантаження файлу

Програмна реалізація алгоритму перетворення зображення на квадратне має наступний вигляд:



Рисунок 3.3 – Функція для створення бокових відступів ч.1



Рисунок 3.4 – Функція для створення бокових відступів ч.2

Програмна реалізація алгоритму перетворення зображення на напівтонове має наступний вигляд:

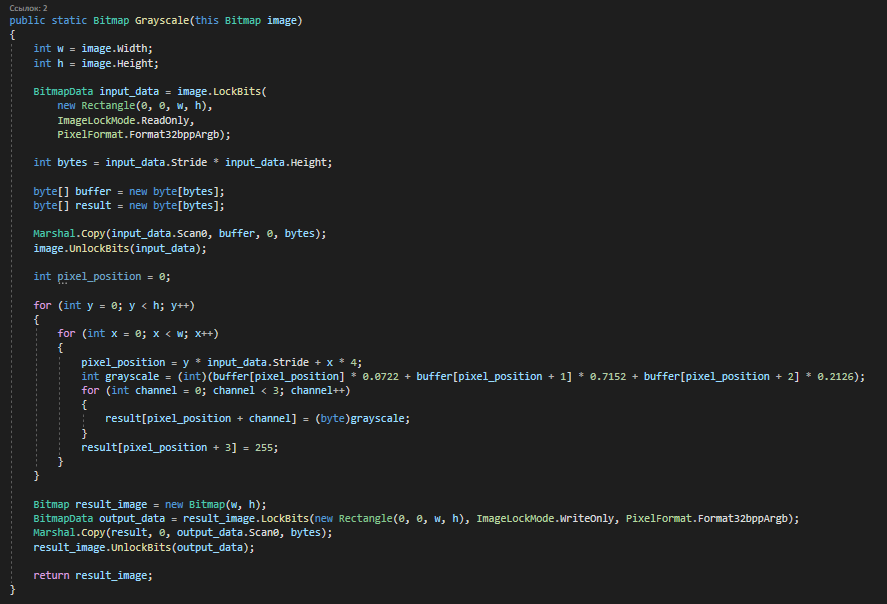


Рисунок 3.5 – Функція перетворення у напівтонове зображення

На даному етапі розглянемо детально алгоритм перетворення Фур’є, який був описаний формулами (2.9).

Перший крок, який нам потрібно зробити, це перетворити значення пікселів нашого зображення на комплексну карту чисел. Взявши значення пікселя в каналі і поставивши його як реальний компонент з уявним компонентом, що прямує до 0.

На вході функція приймає Bitmap об’єкт, тобто наше зображення та копіює інформацію про розмір зображення, задає йому кодування та створює масив байтів у котрий потім запише результат перетворення наших пікселів у комплексну карту чисел.

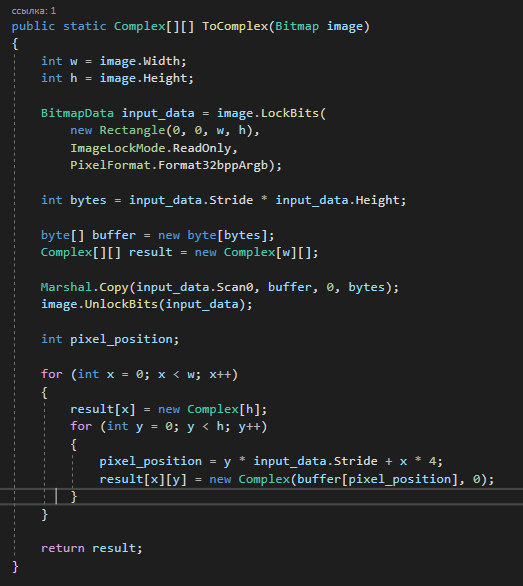


Рисунок 3.6 - Перетворення зображення у комплексну карту чисел

Як тільки ми отримаємо нашу карту комплексних чисел, готову до того, щоб бути переданою у функцію перетворення Фур'є, ми можемо поглянути на те, як функція спочатку обчислює один масив значень.

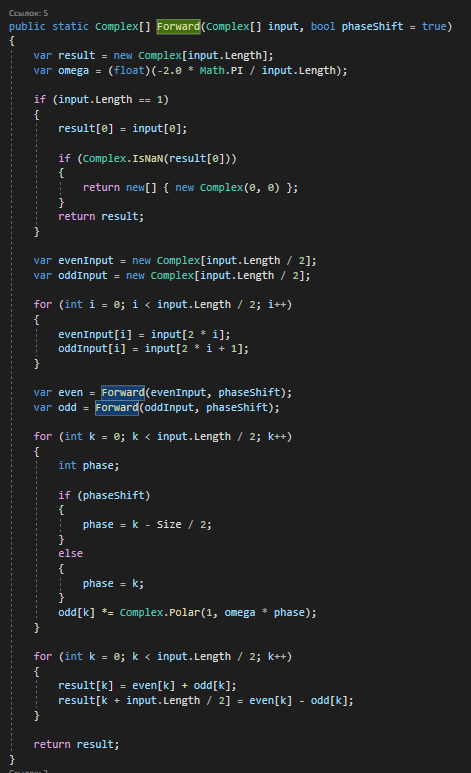


Рисунок 3.7 – Функція перетворення Фур’є ч.1

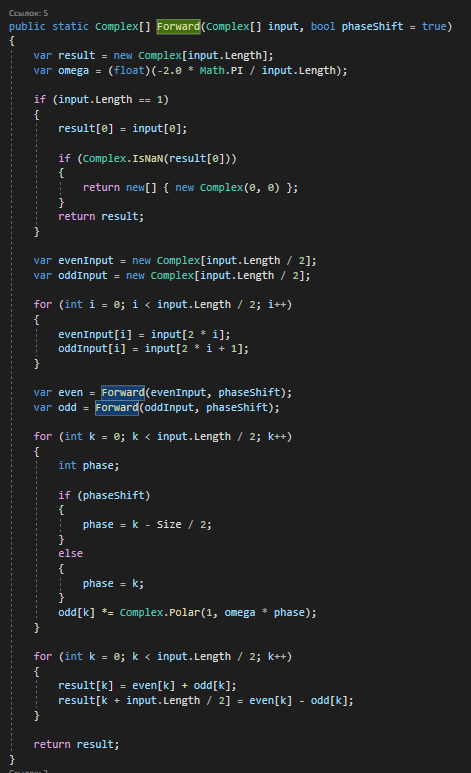


Рисунок 3.8 – Функція перетворення Фур’є ч.2

Тепер, коли ми знаємо, як обчислити один масив чисел, перейдемо до всього зображення.



Рисунок 3.9 – Функція перетворення Фур’є для всього зображення

Так як наша функція повертає масив комплексних чисел, нам потрібно розробити функцію, яка буде приймати цей масив комплексних чисел, розподіляти на частоти та повертати наше зображення у вигляді набору частот.

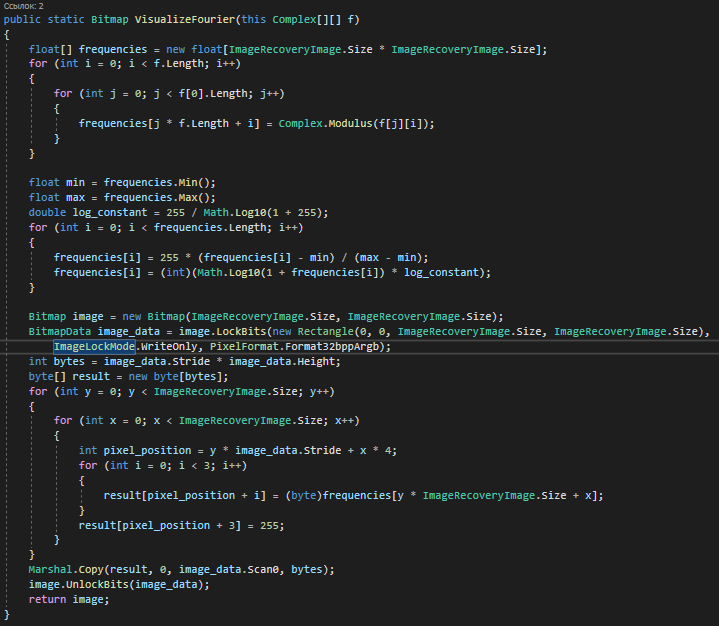


Рисунок 3.10 – Функція представлення Фур’є

Тепер перейдемо до алгоритму спотворення зображення:

Перевіряємо, чи дійсно завантажилось наше зображення. Якщо так – застосовуємо нашу функцію спотворення та виводимо спотворене зображення на екран. Інакше - виводимо сповіщення про помилку. Далі виконуємо ті ж самі дії, що і з вихідним зображенням, тільки спектр виводимо у сусідній елемент.

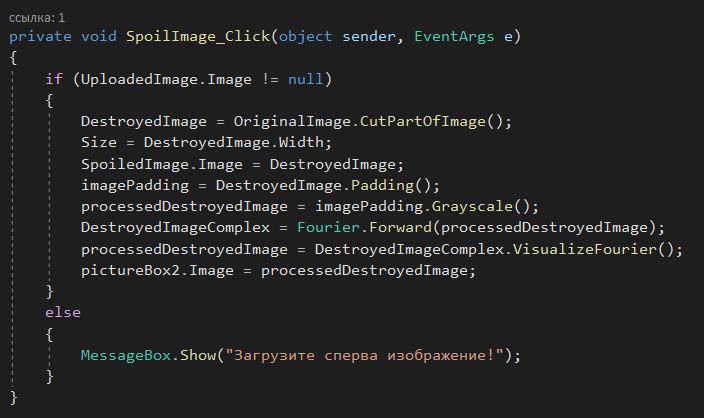
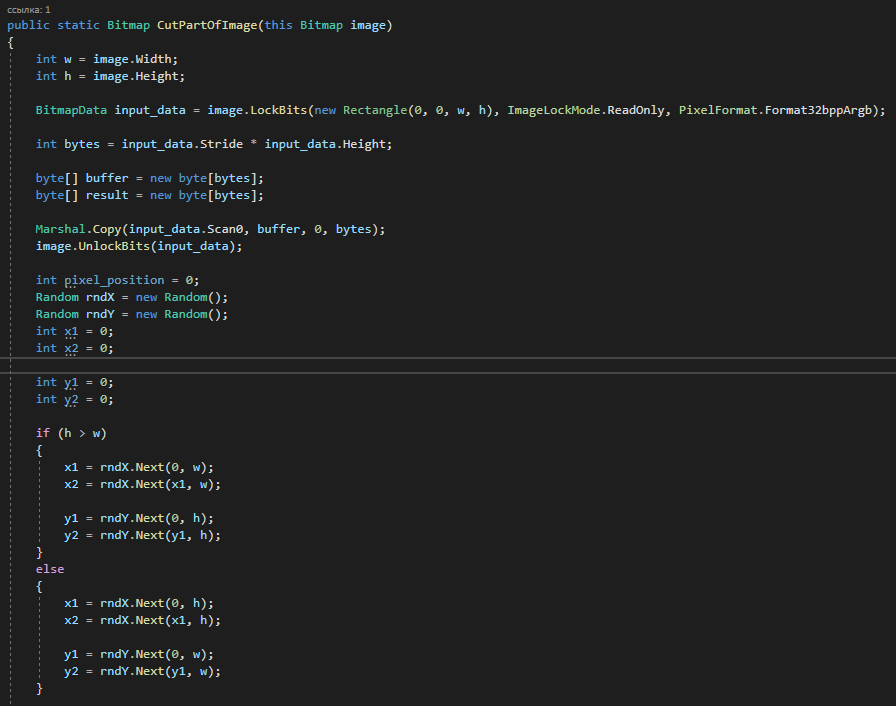


Рисунок 3.11 – Спотворення вихідного файлу

Перейдемо безпосередньо до функції CutPartOfImage(). Вона виконує в нас роль спотворювача зображення. На вході ми маємо наше початкове зображення, зберігаємо його ширину та висоту. Копіюємо наші дані про зображення у допоміжний масив байтів. Далі генеруємо розмір нашої рандомної області, що залежить від розмірів нашого зображення. Маємо за ціль зробити так, щоб пошкоджена область чітко потрапляла у межі нашого зображення. Потім рухаємось поелементно по нашому зображенню і, якщо індекс нашого елемента співпадає з індексом елемента спотворюючої фігури, видаляємо його шляхом перетворення альфа каналів.



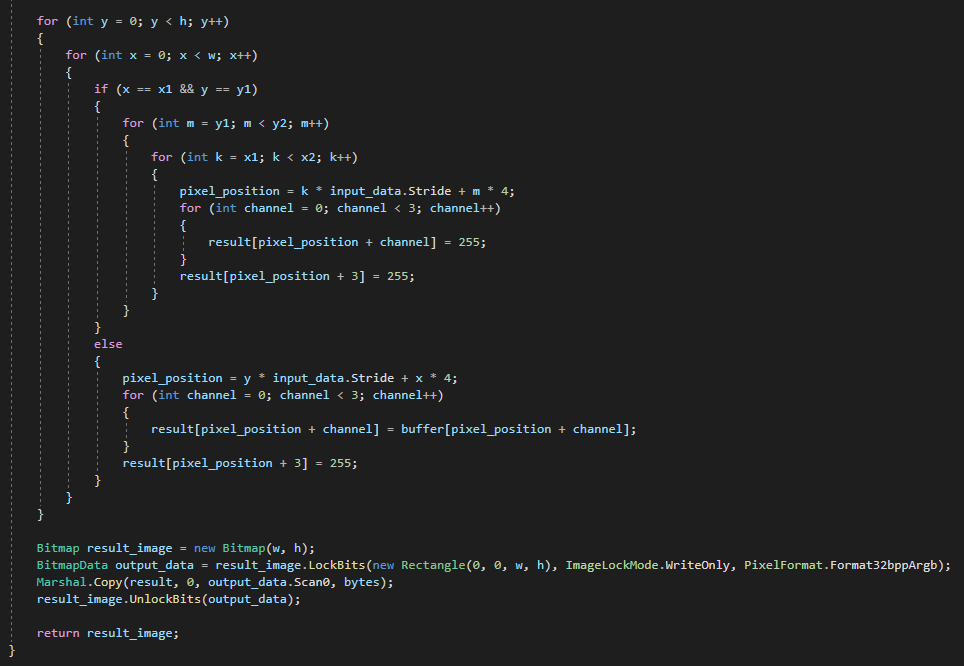


Рисунок 3.12 - Функція спотворення зображення

Наступним кроком розглянемо алгоритм відновлення нашого зображення. На виході після попереднього кроку ми маємо спотворене зображення та його частотне представлення (див. 3.13).

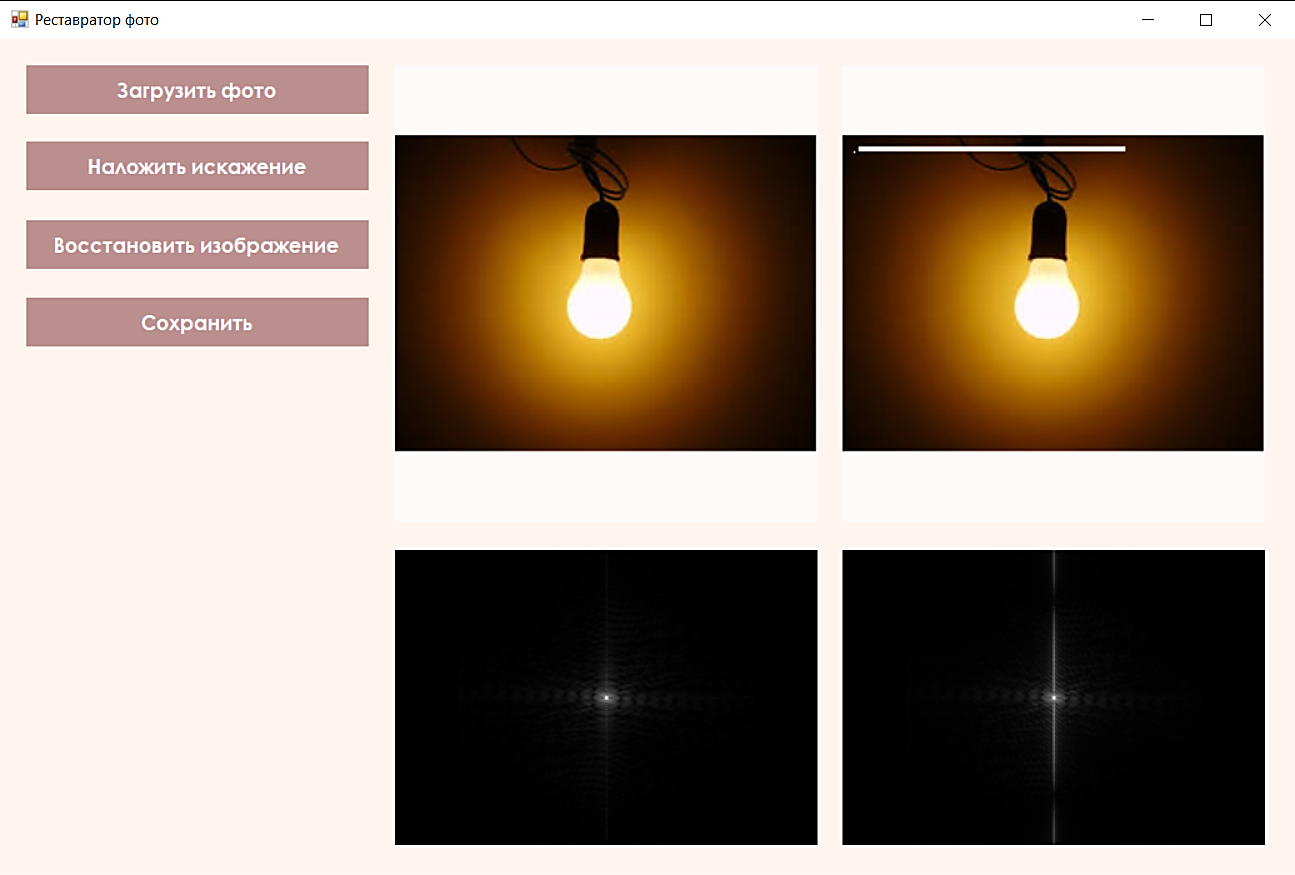


Рисунок 3.13 – Графічний інтерфейс ПЗ після спотворення зображення

Тепер перед нами постає задача відновити спотворене зображення таким чином, щоб воно було максимально наближено до вихідного. Представимо наступну функцію:

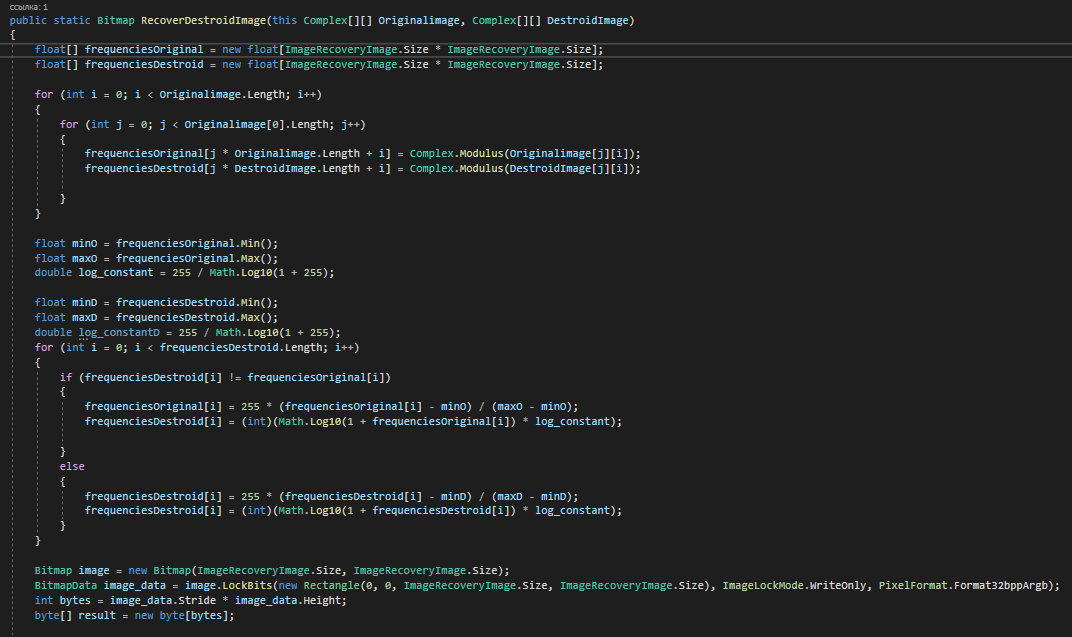


Рисунок 3.14 – Функція відновлення зображення ч.1

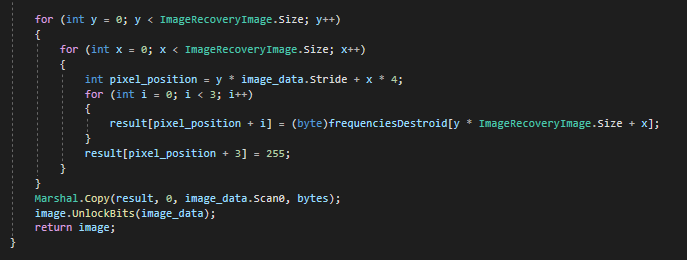


Рисунок 3.15 – Функція відновлення зображення ч.2

На початку ми створюємо два масиви наших частот, один для початкового зображення, другий – для спотвореного. Так як ми передаємо у функцію наші зображення у вигляді масивів комплексних чисел, нам потрібна попередня обробка даних для переходу у частотну область.

Виконавши це, ми переходимо до знаходження максимальних та мінімальних частот зображень.

Тепер безпосередньо проходимо по всім частотам нашого спотвореного зображення. Зустрівши першу частоту, яка не співпадає з початковим зображенням заповнюємо її нульовою частотою початкового зображення. Якщо все задовольняє нашій умові, то залишаємо її, як є. Так рухаємось до кінця циклу. Як результат, ми отримаємо новий набір частот, що є максимально наближеним до набору частот вихідного зображення. Переглянемо результат роботи алгоритму:

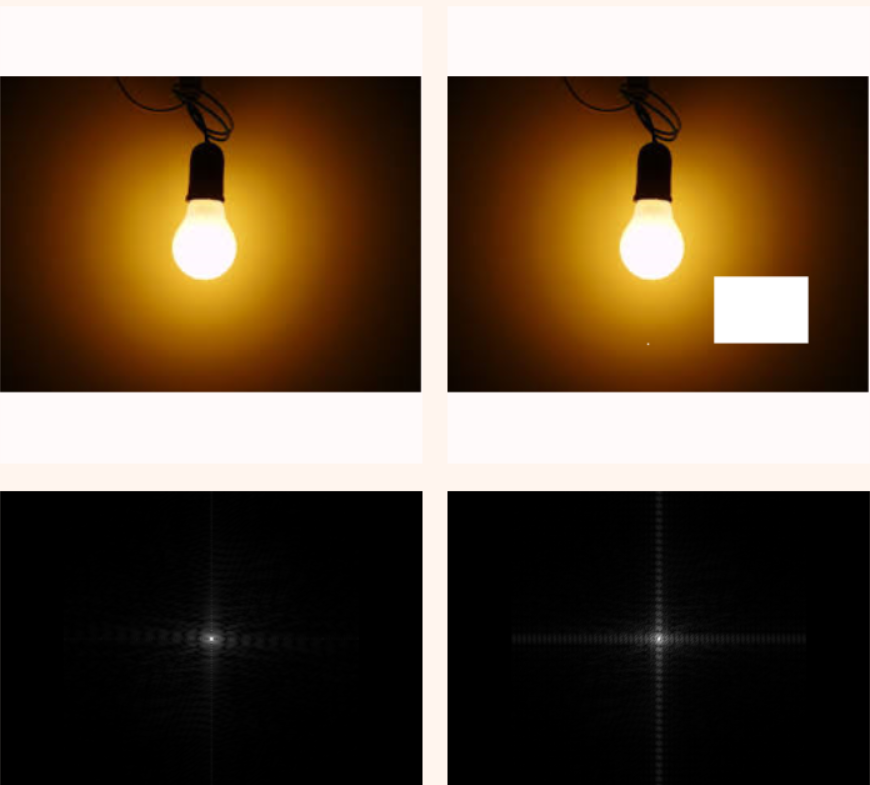
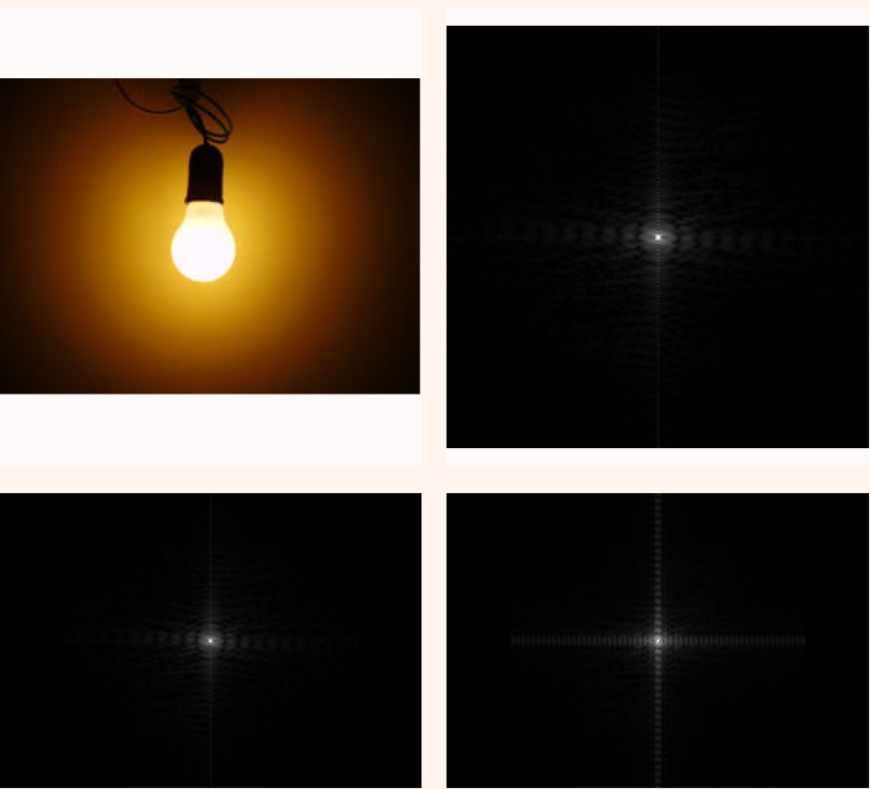
 

Рисунок 3.16 – Представлення Фур’є відновленого зображення.

Як ми можемо спостерігати, спектр Фур’є відновленого зображення майже співпадає зі спектром вихідного зображення. А, отже, ми досягли бажаної цілі. Звісно, є деякі недоліки алгоритму і він не дає 100% відновлення зображення, але результати доволі близькі.

Зараз ми маємо наше відновлене зображення, представлене у частотній області і тепер треба повернутися у просторову область і побачити звичне представлення зображення у вигляді пікселів. Для цього нам знадобиться зворотне перетворення Фур’є.

Розглянемо функцію зворотного перетворення Фур’є, яке допомагає нам повернутися до просторових координат та відобразити наше зображення. Видно, що формули для прямого і зворотного перетворення досить схожі. Таким чином, ми можемо використовувати повторно функцію прямого перетворення. Якщо ми будемо використовувати зв'язку комплексних чисел з функцією прямого перетворення, і цей результат, то ми зможемо отримати просторові пікселі, отримуючи їх величину.

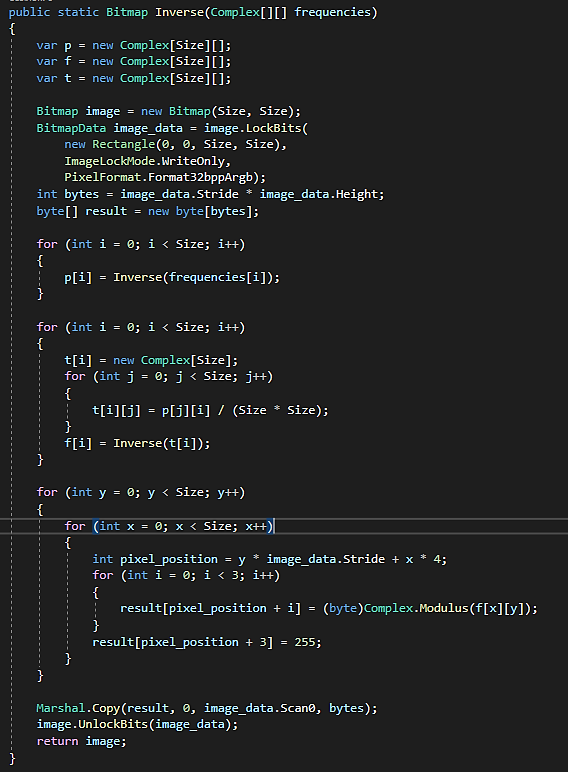


Рисунок 3.17 – Функція зворотного перетворення Фур’є

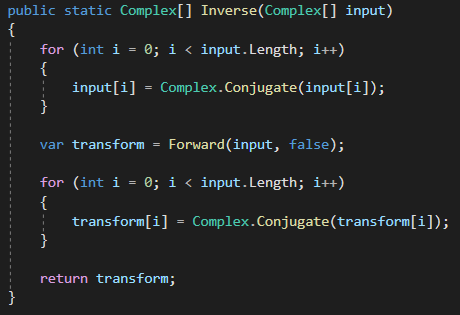


Рисунок 3.18 – Зворотне перетворення Фур’є для всього зображення

Результат роботи функції можемо спостерігати у сусідньому віконечку:

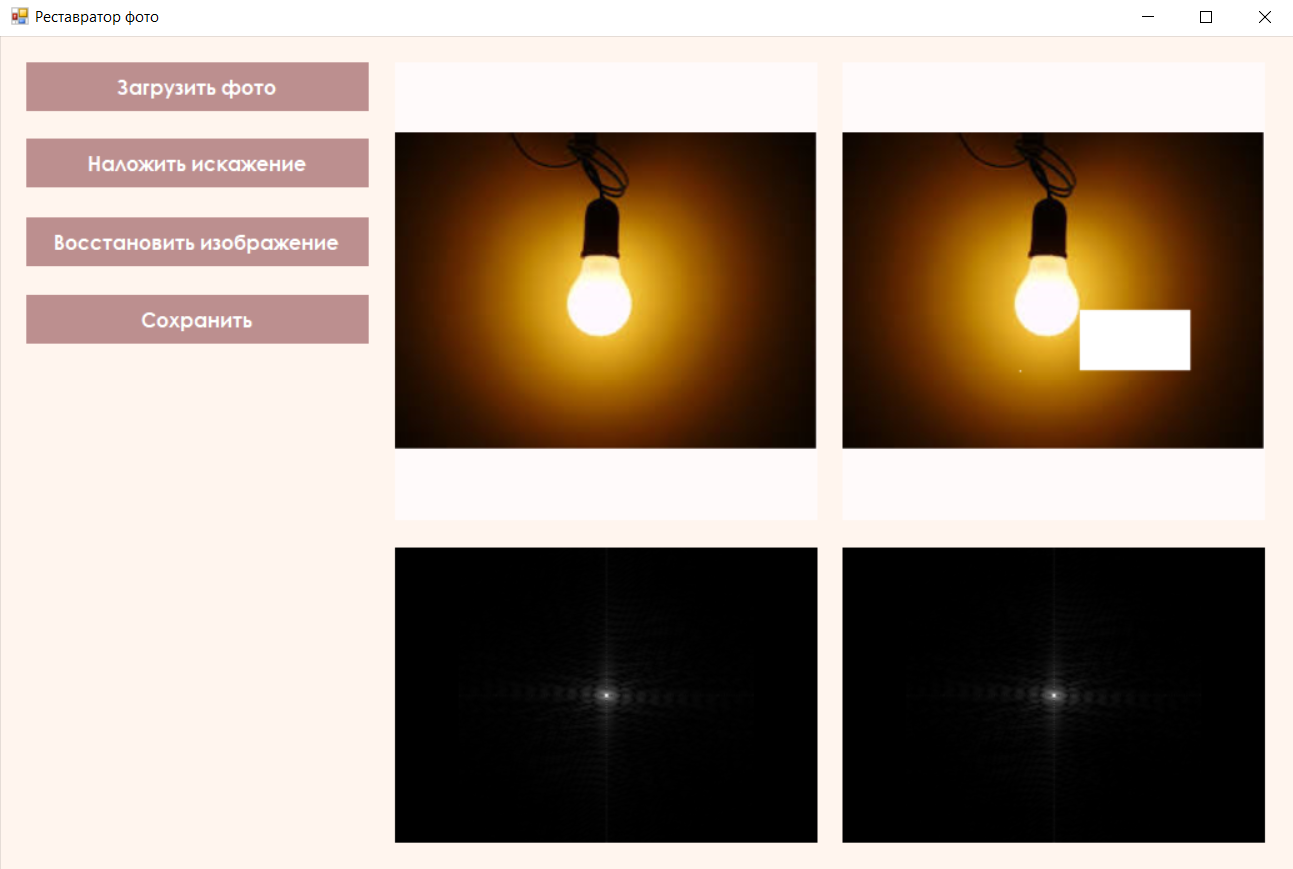


Рисунок 3.19 – Початкове та спотворене зображення

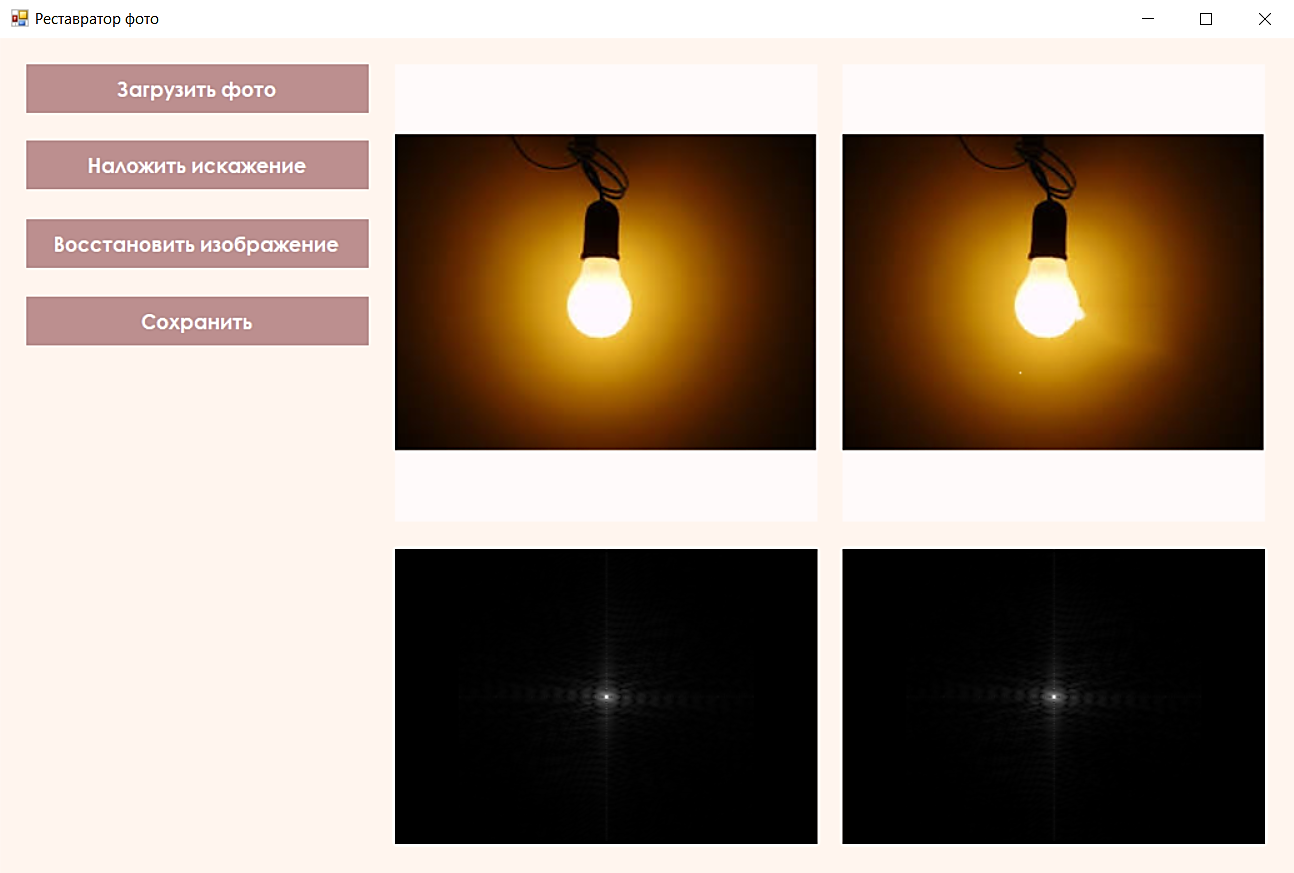


Рисунок 3.20 – Результат роботи алгоритму

Видно, що у алгоритму є деякі недоліки щодо відновлення зображення, адже він чудово впорався з градієнтним фоном зображення, але трохи зіпсував форму лампочки.

Отримавши наше зображення, ми повинні мати можливість його зберегти. Алгоритм збереження доволі простий та передбачає використання заздалегідь збереженої директорії зображення. Збереження відбувається аналогічно за допомогою елементу ***OpenFileDialog***. Ми перевіряємо, чи відкрився наш діалог, обираємо директорію для зберігання файлу, якщо бажаємо її змінити. Результуючий файл зберігається.

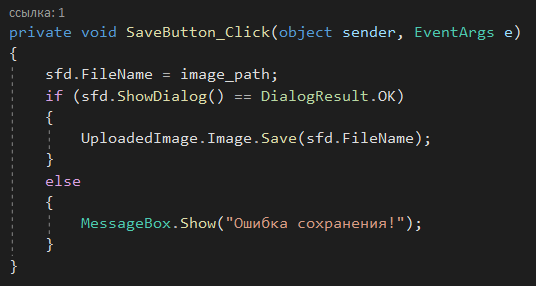


Рисунок 3.21 – Збереження зображення

# АНАЛІЗ ОТРИМАНИХ РЕЗУЛЬТАТІВ РОЗВ’ЯЗАННЯ ЗАДАЧІ

Для того, щоб дійсно розуміти, наскільки якісно виконує свою роботу алгоритм, використаєм декілька різноманітних зображень та подивимось як з ними впорається алгоритм та який буде результат. Візьмемо перше вихідне зображення:

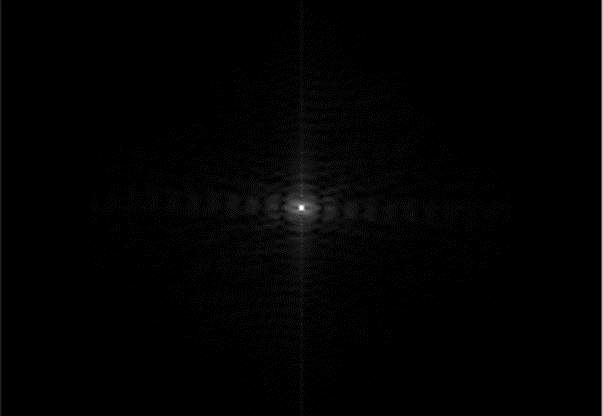
 

Рисунок 4.1 – вихідне зображення №1: а) у просторовій області; б) у частотній області

Тепер застосуємо до зображення функцію спотворення та подивимось на результат.

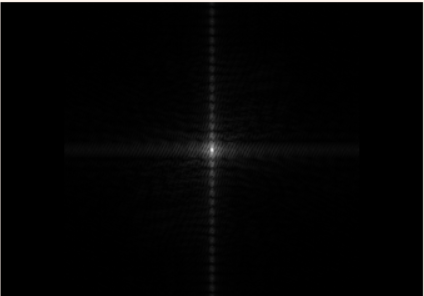
 

Рисунок 4.2 – спотворене зображення №1: а) у просторовій області; б) у частотній області

Відновимо зображення. Порівняння рисунку 1 та 3 показує, що з кожною ітерацією наш ретушований прямокутник заповнювався зображенням, все більш подібним до оточення. Але коли ми дійшли до зображення, що містить частину лампочки – алгоритм спрацював некоректно. Перш за все, це могло статись, тому що колір нашої лакуни білий і лампочки також відповідно. Тому при перевірці, частоти скоріш за все наклались одна на одну та, як результат ми отримали такий результат. З іншого боку якість нашого зображення відносно невелика, спробуємо наступне фото обрати з більш високою чіткістю.

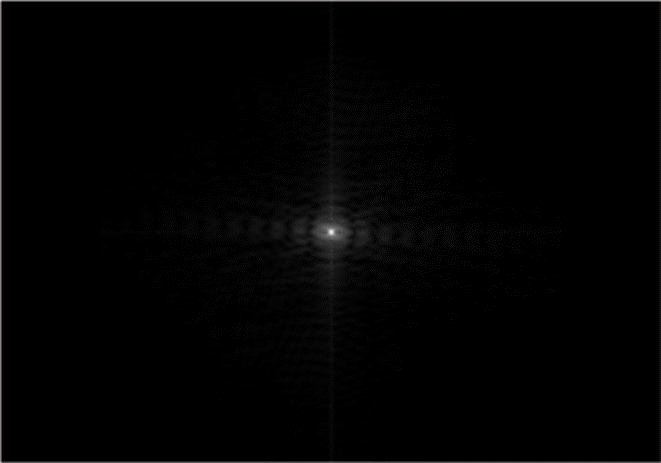
 

Рисунок 4.3 – відновлене зображення №1: а) у просторовій області; б) у частотній області

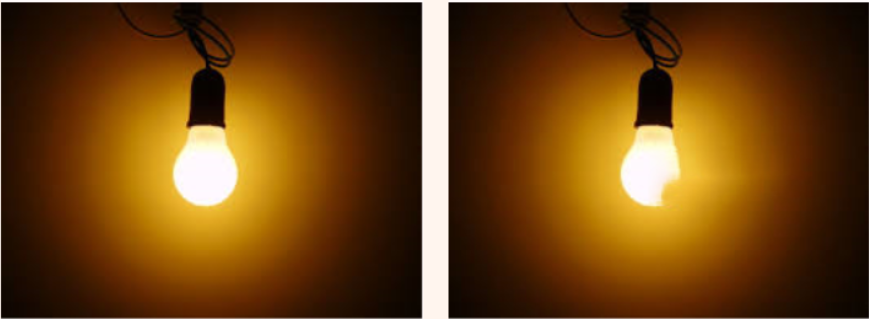


Рисунок 4.4 – зображення №1: а) оригінал; б) відновлене

Цікаво розглянути деформацію амплітудного спектру Фур’є, яка відбувалась протягом застосування функцій до нашого зображення. Порівняння рисунків 4.5.а)-в) дозволяє зробити висновок, що під час ретушування амплітудний спектр було відновлено максимально наближено до вихідного.

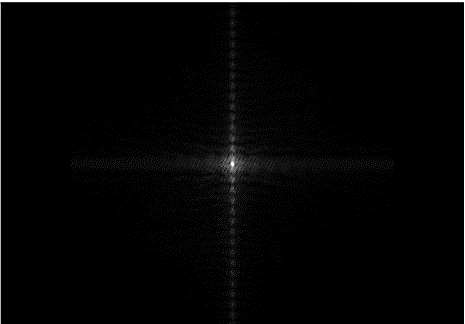
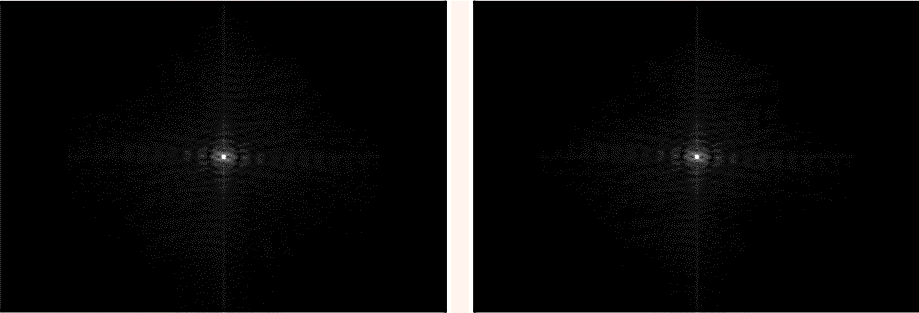
  

Рисунок 4.5 – зображення №1 у частотній області: а) оригінал; б) спотворене; в) відновлене

Розглянемо інше зображення. На цей раз візьмемо зображення близьке до патерну, тобто елементи якого повторюються через певний крок. Маємо чудове зображення з зірочками та ще й в різних кольорах.

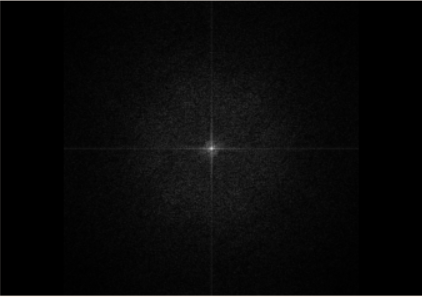
 

Рисунок 4.6 – вихідне зображення №2: а) у просторовій області; б) у частотній області

Аналогічно застосуємо до зображення спотворюючу функцію.

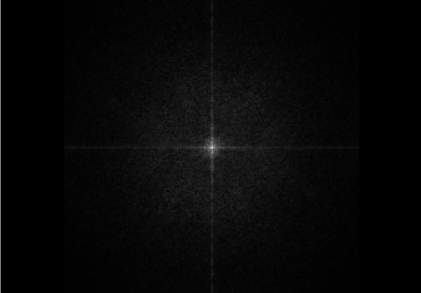
 

Рисунок 4.7 – спотворене зображення №2: а) у просторовій області; б) у частотній області

Застосувавши функцію відновлення, ми можемо спостерігати наступний результат. Наш алгоритм майже чудово впорався з поставленим завданням. Якщо уважно розглядати зображення, можна помітити лише невелике розмиття у місцях, де було спотворення та відсутність однієї блакитної зірки, але загалом ми майже на 100% наблизились до вихідного зображення.

Рисунок 4.8 – відновлене зображення №2: а) у просторовій області; б) у частотній області

Рисунок 4.9 – зображення №2: а) оригінал; б) відновлене

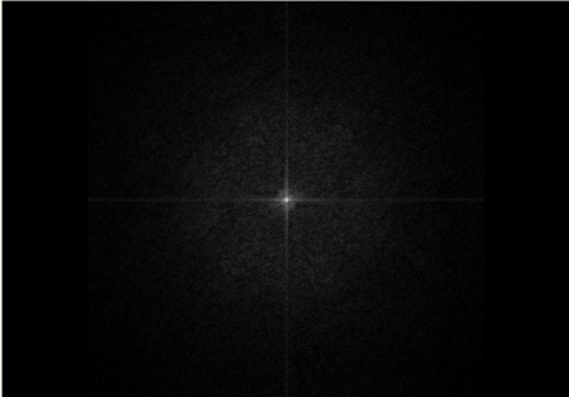
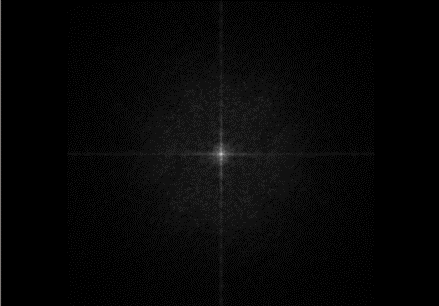
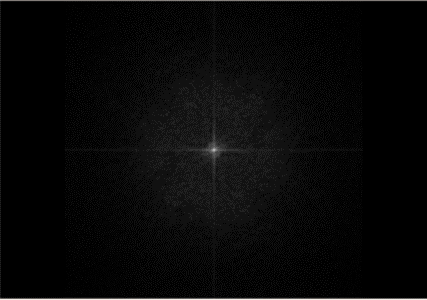


Рисунок 4.10 – зображення №2 у частотній області: а) оригінал; б) спотворене; в) відновлене

Третє зображення візьмемо наближене до першого, але в більш темних кольорах.

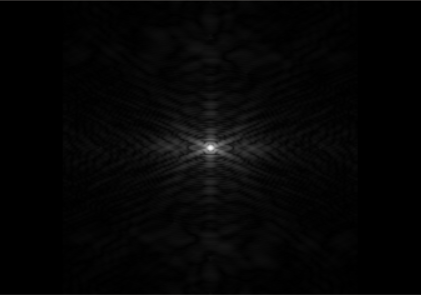
 

Рисунок 4.11 – вихідне зображення №3: а) у просторовій області; б) у частотній області

Аналогічно застосуємо до зображення спотворюючу функцію.

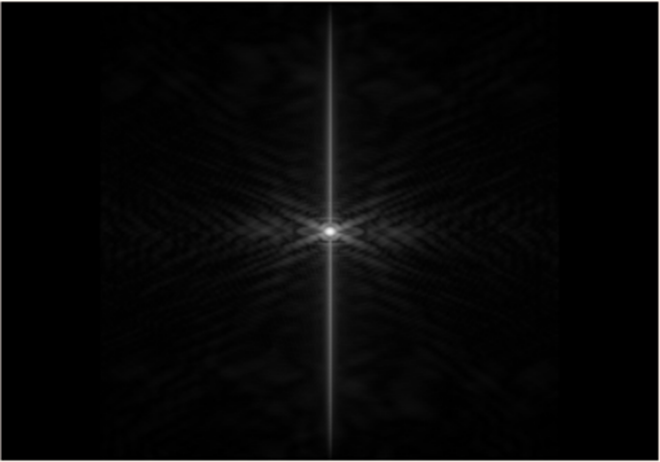
 

Рисунок 4.12 – спотворене зображення №3: а) у просторовій області; б) у частотній області

Звернемо увагу на те, як змінився спектр Фур’є зображення.

Застосуємо функцію відновлення до зображення. Наше спотворення було розміщене таким чином, що повністю потрапляє на однотонний фон, тому алгоритм впорався дуже добре і важко помітити хоч якісь недоліки.

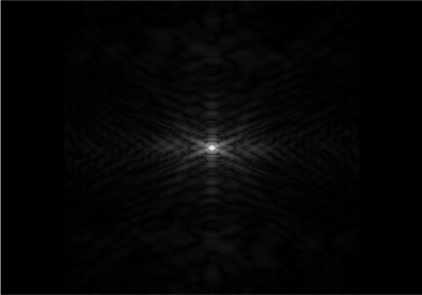
 

Рисунок 4.13 – відновлене зображення №3: а) у просторовій області; б) у частотній області

Наведемо контр приклад, коли спотворення потрапить на зображення лампочки і подивимось, чи впорається алгоритм з таким зображенням.

Рисунок 4.14 – зображення №3: а) спотворене; б) відновлене

Як бачимо, все ж таки з предметами алгоритм впорався на дуже та потребує вдосконалення для більш точного розпізнавання контурів та форм предметів на зображенні. Надалі потрібно використовувати алгоритм розпізнавання контурів, тоді під час відновлення зображення простіше буде відслідкувати, коли саме ми наштовхнулися на певний предмет на зображенні.

Отже, ми розглянули декілька реальних прикладів, щоб перевірити, як добре і коректно відпрацьовує алгоритм. Аналізуючи результати, можна прийти до висновків, що він добре працює з однотонним фоном, або градієнтним, добре відновлює текстурні фото, але якщо на фото з’являються предметні елементи, типу лампочки, або дещо інше, то алгоритм потребує вдосконалення для кращої роботи. Попередньо можна припустити, що для вдосконалення алгоритму потрібно використовувати частотне представлення не повних спотвореного та оригінального зображень, а навпаки, локалізувати область відновлення якнайбільш ближче до спотвореної. Надалі алгоритм буде вдосконалено.

# ВИСНОВКИ

Розглядаючи різноманітні теми, які ставляться моєї переддипломної роботи, я оволоділа багатьма новими знаннями та навичками. Виконала аналітичний огляд предметної області відновлення зображень в загублених або пошкоджених областях, різноманітних моделей спотворення зображень та існуючих алгоритмів відновлення. Також виявила актуальність моєї роботи, адже задача відновлення зображень є до сих пір до кінця не вирішеною, бо існує обмеження у розвитку методів відновлення – відсутність об'єктивного критерію оцінки якості реконструкції, що заважає коректному одночасному урахуванню різних аспектів змісту зображень (зокрема, текстурного і структурного).

У роботі були розглянуті основні існуючі підходи для вирішення задачі поліпшення цифрового зображення та відновлення його структури. Проаналізовано метод обробки зображення в частотній області і його математична модель. Розглянуто алгоритм фільтрації в частотній області і представлена покрокова схема його роботи для поліпшення якості зображення. Наведено один із видів методів обробки зображень в частотної області – за допомогою швидкого перетворення Фур’є.

Як результат, було розроблено програмне забезпечення для відновлення та ретушування зображень за допомогою їх спектру Фур’є.

# СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Рейда А.Н. Методы улучшения цифрового изображения и восстановления его структуры. / А.Н. Рейда, Ю.В. Олейник, А.А. Панчук, М.Л. Синенький. – Изд-во ВНТУ, 2000. – 7 с.
2. Грузман И.С. Цифровая обработка изображений в информационных системах: Учебное пособие. / И.С. Грузман, В.С. Киричук, В.П. Косых, Г.И. Перетягин, А.А. Спектор. – Новосибирск: Изд-во НГТУ, 2000. – 352 с.
3. Гонсалес Р.С. Цифровая обработка изображений. / Р.С. Гонсалес, Р. Вурдс, С. Єддинс. – Изд-во Техносфера, 2005. – 1071 с.
4. Творошенко І.С. Цифрова обробка зображень: конспект лекцій. / м. Харків – ХНУМГ ім. О. М. Бекетова, 2016. – 55с.
5. Щербаков О.В. Реконструкция изображений в отсутствующих или повреждённых областях на основе критерия алгоритмической вероятности. / Санкт-Петербург, 2015. – 128 с.
6. Ежова К.В. Моделирование и обработка изображений. / СПб: НИУ ИТМО, 2011. – 93 с.
7. Пелешко Д.Д. Модель утворення локальних спотворень зображення. / Д.Д. Пелешко, О.В. Клювак, А.М. Ковальчук, І.В. Ізонін, М.І. Голубінська. – м. Львів – НУЛП, 2013. – 6 с.
8. Гаевская Е.А. Методы обработки изображений, полученных методом фазового контраста. / Е.А. Гавская, С.Г. Стиренко. – Вісник НТУУ «КПІ». Інформатика, управління та обчислювальна техніка : збірник наукових праць. – 2011. – 6 с.
9. Кокошкин А.В. Ретуширование и восстановление отсутствующих частей изображений с помощью итерационного вычисления их спектров. / А.В. Кокошкин, В.А. Коротков, К.В. Коротков, Е.П. Новичихин // Компьютерная оптика. – 2019. – 11 с.
10. T. Xavier, “Development of image restoration techniques” / Rourkela city – NIT, 2007. – 52 p.
11. Кокошкин А.В. Использование спектра Фурье изображения для ретуширования и восстановления отсутствующих частей искажённого аппаратной функцией изображения / А.В. Кокошкин, В.А. Коротков, К.В. Коротков // Журнал радиоэлектроники. – 2016. – 13 с.
12. How To Use Fourier Transform On Images – C# Guide [Електронний ресурс] / A. Krzisnik, 2020. – 1 с. – Режим доступу до статті: https://epochabuse.com/fourier-transform/
13. Введение в C#. Язык C# и платформа .NET Core [Електронний ресурс] / Режим доступу до статті: https://metanit.com/sharp/tutorial/1.1.php
14. Windows Forms [Електронний ресурс] / Режим доступу до статті: https://ru.wikipedia.org/wiki/Windows\_Form