

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Інформаційно-аналітичних технологій та менеджменту
(повна назва)

Кафедра Інформатики
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти перший (бакалаврський)

РОЗРОБКА ЗАСТОСУНКУ ДЛЯ АНАЛІЗУ
СЛАБОСТРУКТУРОВАНИХ ЗОБРАЖЕНЬ
(тема)

Виконав:
студент 4 курсу, групи ІТІНФ-19-1
Халькевич О.Д.
(прізвище, ініціали)

Спеціальності 122 Комп'ютерні науки
(код і повна назва спеціальності)

Тип програми освітньо-професійна

Освітня програма Інформатика
(повна назва освітньої програми)

Керівник асист. Кобилін І.О.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри _____
(підпис)

Кобилін О.А.
(прізвище, ініціали)

2023 р.

Харківський національний університет радіоелектроніки

Факультет Інформаційно-аналітичних технологій та менеджменту
(повна назва)Кафедра Інформатики
(повна назва)Рівень вищої освіти перший (бакалаврський)Спеціальність 122 Комп'ютерні науки
(код і повна назва)Тип програми освітньо-професійнаОсвітня програма Інформатика
(повна назва освітньої програми)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

« ____ » _____ 2023 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУстудентові Халькевич Ользі Дмитрівні
(прізвище, ім'я, по батькові)1. Тема роботи Розробка застосунку для аналізу слабоструктурованих зображень

затверджена наказом університету від 15 травня 2023 року № 474 Ст

2. Термін подання студентом роботи до екзаменаційної комісії 29 травня 2023 р.

3. Вихідні дані до роботи набір слабоструктурованих зображень, науково-технічна література, методи та алгоритми обробки зображень, бібліотеки та інструменти.

4. Перелік питань, що потрібно опрацювати в роботі _____

1. Огляд методів аналізу слабоструктурованих зображень: необхідно дослідити різні підходи до аналізу та обробки зображень з низьким контрастом, нерівномірним освітленням або шумом.2. Вибір підходу до покращення якості зображень: розглянути різні методи та алгоритми для підвищення різкості, зменшення шуму та видалення артефактів на слабоструктурованих зображеннях.3. Розробка алгоритмів виявлення об'єктів: вивчити техніки для виявлення та виділення об'єктів на слабоструктурованих зображеннях, такі як сегментація, детекція контурів або класифікація.4. Валідація та оцінка результатів: розробити метрики та методики для оцінки ефективності розробленого застосунку, провести експерименти на вхідних наборах слабоструктурованих зображень та порівняти результати з існуючими методами.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри) Актуальність проблеми обробки зображень, постановка задачі, тестові зображення.

6. Консультанти розділів роботи (п.6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата
Консультант з дотримання діючих стандартів та норм	Доцент Творошенко І.С.		

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Отримання завдання на кваліфікаційну роботу	10.04.2023	
2	Аналіз завдання, підбір літератури	11.04.23-17.04.23	
3	Аналіз літератури з досліджуваної проблеми	18.04.23-20.04.23	
4	Аналіз технічних засобів	21.04.23-30.04.23	
5	Розробка методу	01.05.23-14.05.23	
6	Програмна реалізація	15.05.23-23.05.23	
7	Оформлення пояснювальної записки	24.05.23-26.05.23	
8	Перевірка на плагіат	27.05.23	
9	Рецензування	28.05.23	
10	Підготовка презентації та доповіді	29.05.23-30.05.23	
11	Занесення роботи в електронний архів	31.05.23	
12	Попередній захист кваліфікаційної роботи	06.06.23	

Дата видачі завдання 10 квітня 2023 р.

Студент _____
(підпис)

Керівник роботи _____ асист. Кобилін І.О.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ/ABSTRACT

Пояснювальна записка до кваліфікаційної роботи: 57 с., 38 рис., 32 джерела.

МЕДІАННИЙ ФІЛЬТР, ГАУСІВСЬКИЙ ФІЛЬТР, ФІЛЬТР СОБЕЛЯ, ФІЛЬТР ЛАПЛАСА, ФІЛЬТР МЕДІАННОЇ РІЗНИЦІ, ФІЛЬТР БІЛАТЕРАЛЬНОЇ ФІЛЬТРАЦІЇ, MONGODB, NEXT-AUTH, NEXT.JS, NODE.JS, REACT.

Об'єктом роботи є застосунок для аналізу слабоструктурованих зображень.

Метою даної кваліфікаційної роботи є створення зручного та ефективного інструменту, який дозволяє користувачам проводити аналіз та обробку зображень.

Під час розробки застосунку були використані сучасні технології та фреймворки, такі як React, Next.js, Node.js, MongoDB та інші. Застосунок надає різноманітні методи та фільтри для аналізу та покращення слабоструктурованих зображень. Користувачі можуть застосовувати фільтри, такі як медіанний фільтр, гаусівський фільтр, фільтр Собеля та багато інших, для поліпшення візуальних якостей зображень.

Розроблений застосунок дозволяє ефективно аналізувати та обробляти слабоструктуровані зображення, що дозволяє користувачам досягати більшої якості та деталізації в їх зображеннях.

MEDIAN FILTER, GAUSSIAN FILTER, SOBEL FILTER, LAPLACIAN FILTER, MEDIAN DIFFERENCE FILTER, BILATERAL FILTER, MONGODB, NEXT-AUTH, NEXT.JS, NODE.JS, REACT.

The objective of this work is an application for analyzing weakly structured images.

The aim of this qualification work is to create a convenient and efficient tool that allows users to analyze and process images.

During the development of the application, modern technologies and frameworks such as React, Next.js, Node.js, MongoDB, and others were used. The application provides various methods and filters for the analysis and enhancement of weakly structured images. Users can apply filters such as the median filter, Gaussian filter, Sobel filter, and many others to improve the visual quality of the images.

The developed application enables efficient analysis and processing of weakly structured images, allowing users to achieve higher quality and detail in their images.

ЗМІСТ

Перелік умовних позначень, символів, одиниць, скорочень і термінів	4
Вступ	5
1 Слабоструктуровані зображення	6
1.1 Обробка зображень на основі контурів	6
1.2 Обробка слабоструктурованих зображень на основі ліній	9
1.3 Обробка зображень на основі структур	10
1.4 Фрактальна обробка зображень. Класифікація фракталів. Алгебраїчні, геометричні та стохастичні фрактали.	10
1.5 Постановка задачі	13
2 Математичне обґрунтування обраних методів	15
2.1 Усунення шумів їх класифікація і їх методи усунення	15
2.2 Робастність. Дослідження викидів і стійкість до перешкод	18
2.3 Гаусовський «білий» шум	19
2.4 Чергування чорних та білих частинок (шум солі та перцю)	20
2.5 Методи фільтрації зображень	21
3 Реалізація прототипу застосунка	29
3.1 Figma. Основна інформація. Figma для мобільного дизайну	29
3.2 Опис програмної реалізації	32
Висновки	53
Перелік джерел посилання	54

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

- C3 – слабоструктуровані зображення
- UI – User Interface (інтерфейс користувача)
- API – Application Programming Interface (інтерфейс програмування застосунків)
- ML – Machine Learning (машинне навчання)
- CV – Computer Vision (комп'ютерне зорове сприйняття)
- ROI – Region Of Interest (область інтересу)
- GUI – Graphical User Interface (графічний інтерфейс користувача)
- CNN – Convolutional Neural Network (згорткова нейронна мережа)
- GPU – Graphics Processing Unit (графічний процесор)
- JPEG – Joint Photographic Experts Group (об'єднана експертна група з фотографій)
- APIK – Application Programming Interface Key (ключ інтерфейса програмування застосунків)
- SDK – Software Development Kit (набір розробничих інструментів)
- FPS – Frames Per Second (кадри в секунду)
- GUI – Graphics Processing Unit (графічний процесор)
- ROI – Region of Interest (область інтересу)
- RGB – Red Green Blue (червоний зелений синій)
- SVM – Support Vector Machine метод опорних векторів
- JSON – JavaScript Object Notation (нотація об'єктів JavaScript)
- APIR – Application Programming Interface Resource (ресурс інтерфейса програмування застосунків)
- OCR – Optical Character Recognition (оптичне розпізнавання символів)

ВСТУП

У світі інформаційних технологій значний прошарок займає обробка великої кількості даних, що збирається та аналізується щодня. Достатня кількість таких даних представлена у вигляді або статичних зображень або відеопотоку чи динамічно змінюваних зображень. За допомогою аналізу даних можна здійснити багато корисних висновків та розробити ефективні рішення. Одним з видів даних, що потрібно обробляти є зображення, що зазвичай забруднені або є неструктурованими або слабоструктурованими. Тому для ефективної обробки таких зображень важливо мати інструменти, що дозволяють обробляти дані такого типу : методи передобробки даних, що дозволяють позбутися шумів різного типу та безпосередньо алгоритми обробки зображень. Одним з видів зображень, що потребують уваги та аналізу є слабоструктурні зображення.

Слабоструктурні зображення можуть бути структурованими, неструктурованими або напівструктурованими. Структуровані зображення мають чіткі лінії та регулярні структури, такі як геометричні фігури, решітки, сітки тощо. Неструктуровані зображення, навпаки, не мають чіткої структури та можуть містити велику кількість деталей, таких як текстури, форми, кольори тощо. Напівструктуровані зображення мають часткову структуру та можуть містити як регулярні, так і нерегулярні елементи.

У даній кваліфікаційній роботі було запропоновано застосунок для аналізу слабоструктурованих зображень, який буде здатний обробляти різні типи слабоструктурних зображень. Застосунок буде містити алгоритми для обробки зображень, для передобробки зображень та їх подальшого розпізнавання, у зручному для користувача форматі. Це дозволить проводити аналіз таких зображень достатньо ефективно, що буде забезпечувати зручний інтерфейс для взаємодії з користувачем.

1 СЛАБОСТРУКТУРОВАНІ ЗОБРАЖЕННЯ

Існують декілька розповсюджених методів для обробки слабоструктурованих зображень, одним з таких методів є обробка зображень на основі контурів. На рисунку 1.1 наведено приклади різних видів зображень, що потребують подальшого аналізу. Зображення, що потребують подальшої обробки, зазвичай мають ряд проблем, які перешкоджають їх використанню в деяких конкретних задачах. До таких проблем можуть належати: шум, різкість, деформація, зміна освітлення, перешкоди на зображенні.

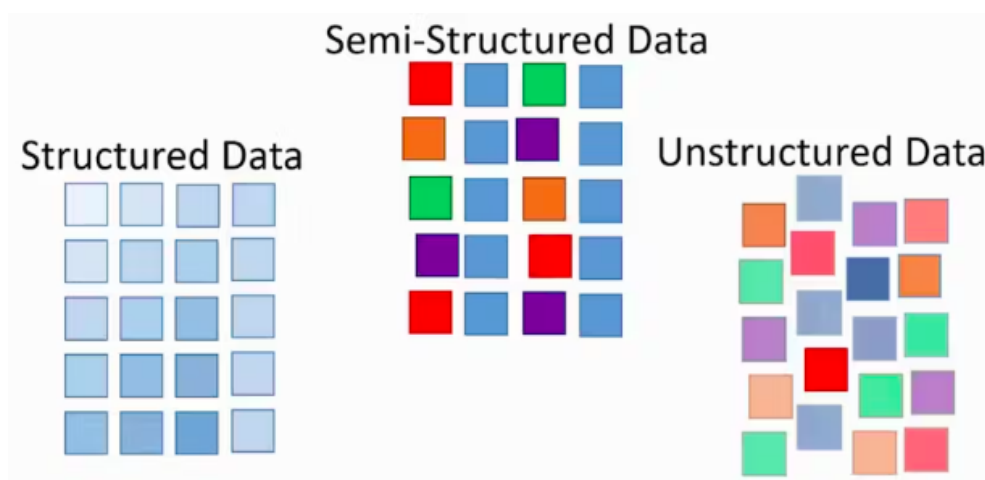


Рисунок 1.1 – Типи зображень

Усі ці проблеми можуть бути вирішені через обробку зображень, що може включати в себе фільтрацію шуму, покращення різкості, корекцію деформації та інше.

1.1 Обробка зображень на основі контурів

Слабоструктуровані зображення – це тип зображень, які містять складну, неоднорідну та неструктуровану інформацію. Обробка таких зображень є складною задачею, оскільки вони можуть містити багато деталей, які не мають чіткої структури [1].

Один з підходів до обробки слабоструктурованих зображень – це обробка на основі контурів. Контур – це замкнутий шлях, який обмежує регіон зображення та відокремлює його від інших регіонів. Обробка зображень на основі контурів полягає в пошуку та відокремленні контурів на зображенні. Для пошуку контурів на зображенні можна використовувати різні методи.

Один з найпоширеніших методів – це метод Кенні. Він полягає в пошуку точок, які знаходяться на границі об'єкту, та з'єднанні їх у вигляді контуру. Метод Кенні дозволяє досить точно визначати контури на зображенні, але він може бути чутливим до шуму та може викликати появу ложних контурів [2].

Інший підхід до обробки слабоструктурних зображень на основі контурів – це використання алгоритмів активних контурів.

Активний контур – це гнучка лінія, яка може змінювати свою форму та розмір, щоб відповідати границі об'єкту на зображенні. Алгоритм активного контуру дозволяє виявляти контури на зображенні з високою точністю та зменшує чутливість до шуму.

Обробка зображень на основі контурів – це підхід до обробки слабоструктурних зображень, який полягає в пошуку та виділенні контурів на зображенні. Контур – це замкнутий шлях, який обмежує регіон зображення та відокремлює його від інших регіонів.

Обробка зображень на основі контурів дозволяє виконувати різноманітні завдання, такі як розпізнавання об'єктів на зображенні, визначення їх розміру та форми, відокремлення фону від об'єкту та багато іншого.

Для пошуку контурів на зображенні можна використовувати різні методи. Один з найпоширеніших методів – це метод Кенні. Він базується на пошуку точок, які знаходяться на границі об'єкту, та з'єднанні їх у вигляді контуру. Метод Кенні дозволяє досить точно визначати контури на зображенні, але може бути чутливим до шуму та може викликати появу ложних контурів.

Функція `edge` використовується для визначення контурів об'єктів. Метод Канні є найбільш ефективним способом, який використовується для досягнення цього (рис. 1.2). В основі цього методу лежить пошук локальних ділянок з перепадами яскравості, які визначаються за допомогою фільтрації по кожній з осей координат одномірним фільтром лапласіан – гаусіана. Для класифікації перепадів на «слабкі» і «сильні» використовуються два пороги – нижній і верхній. Результуюче зображення містить лише «слабкі» границі, якщо вони з'єднані з «сильними». Хоча метод Канні забезпечує найкраще виявлення границь порівняно з іншими методами функції `edge` та методом сегментації для зображень, що містять шум, він вимагає значно більше часу.



Рисунок 1.2 – Розпізнавання об'єктів за допомогою функції `edge`

Інший підхід до обробки слабоструктурних зображень на основі контурів – це використання алгоритмів активних контурів [3].

Активний контур – це гнучка лінія, яка може змінювати свою форму та розмір, щоб відповідати границі об'єкту на зображенні. Алгоритм активного контуру дозволяє виявляти контури на зображенні з високою точністю та зменшує чутливість до шуму [4].

Обробка зображень на основі контурів дозволяє виконувати різноманітні завдання, такі як розпізнавання об'єктів на зображенні, визначення їх розміру та форми, відокремлення фону від об'єкту та інше. Крім того, цей підхід може бути застосований у багатьох галузях, таких як медицина, промисловість, транспорт, розвідка та інші.

1.2 Обробка слабоструктурованих зображень на основі ліній

Обробка слабоструктурованих зображень на основі ліній – це підхід до обробки зображень, який полягає в виділенні та аналізі ліній на зображенні. Лінії можуть бути різних форм та розмірів і відображати різноманітні особливості на зображенні, такі як границі об'єктів, деталі та інші. Для виділення ліній на зображенні можна використовувати різні методи, такі як метод крайніх точок та метод прямих Хафа. Метод крайніх точок полягає в пошуку точок на зображенні, які мають найвищу яскравість в порівнянні з оточенням. Після цього лінії можуть бути відновлені з використанням цих точок. Метод прямих Хафа використовується для знаходження ліній будь-якої форми та розміру на зображенні. Цей метод перетворює кожен точку на зображенні в голосуючу точку в параметричному просторі, який відповідає всім можливим прямим на зображенні. Потім, групуючи голосуючі точки в класи, можна знайти найбільш вірогідні лінії на зображенні.

Після виділення ліній на зображенні можна проводити різні операції, такі як визначення довжини ліній, кутів між ними та інших характеристик. Це може бути корисно при аналізі деталей на зображенні, наприклад, при визначенні форми та розміру об'єктів на зображенні.

Усі ці методи можуть бути використані для розробки застосунку для аналізу даних кластеризації, який може допомогти у визначенні груп об'єктів на зображенні за різними ознаками.

1.3 Обробка зображень на основі структур

Обробка зображень на основі структур – це підхід до обробки зображень, який полягає в аналізі та використанні структури зображення для отримання інформації про зображувані об'єкти. Структура зображення включає різні складові, такі як форму та текстуру, які можуть бути використані для знаходження та аналізу об'єктів на зображенні [5].

Одним з методів обробки зображень на основі структур є методи морфологічної обробки зображень. Цей підхід використовує математичні операції, такі як згортка та дилатація, для виокремлення елементів зображення зі структурою. Застосування цих операцій дозволяє виокремити різні об'єкти на зображенні, такі як краї, контури та інші.

Інший метод обробки зображень на основі структур – це метод сегментації зображення. Цей підхід полягає в розбитті зображення на окремі сегменти або області, що відповідають окремим об'єктам на зображенні. Для цього можуть використовуватися різні методи, такі як метод активного контуру або метод водяного жердини.

Також можна використовувати нейронні мережі для обробки зображень на основі структур. Наприклад, можна використовувати глибокі нейронні мережі для класифікації об'єктів на зображенні за їх формою та розміром.

В цілому, обробка зображень на основі структур може бути корисна для аналізу об'єктів на зображенні за різними характеристиками, такими як форма, текстура та інші.

1.4 Фрактальна обробка зображень

Фрактальна обробка зображень – це підхід до обробки зображень, який базується на фрактальній геометрії. Фрактал – це об'єкт, який має складну

форму, яку можна описати тільки за допомогою самоподібності – тобто, складові частини фрактала мають схожу форму зі всім об'єктом.

У фрактальній обробці зображень використовуються алгоритми, які використовують фрактальну геометрію для опису зображень. Основна ідея полягає в тому, щоб описати зображення за допомогою фрактальних складових частин, які можна повторювати, зменшуючи їх розмір, і при цьому зберігати форму зображення.

Один з найпоширеніших методів фрактальної обробки зображень – це фрактальне стиснення. У цьому методі зображення розділяється на малі квадрати, які називаються пікселями. Кожен піксель описується значенням яскравості, яке можна розглядати як точку в двовимірному просторі.

Потім, за допомогою фрактальної геометрії, ці точки зображення групуються в класи, кожен з яких може бути описаний фрактальним об'єктом. При цьому більші об'єкти зображення зменшуються, а менші – збільшуються, щоб зберігати форму зображення.

Інший метод фрактальної обробки зображень – це фрактальний аналіз. У цьому методі зображення розглядається як фрактальний об'єкт, і за допомогою алгоритмів фрактальної геометрії аналізуються його властивості. Цей метод може бути використаний для розпізнавання образів і класифікації зображень, тож фрактали – це геометричні об'єкти або процеси, що мають складну структуру та характеризуються самоподібністю на різних масштабах. Класифікація фракталів відбувається за різними ознаками, зокрема за їхньою математичною природою [6].

Фрактали можна класифікувати на геометричні (рис. 1.3), алгебраїчні та стохастичні, а також на рукотворні та природні. Рукотворні фрактали є тими, які були створені вченими і відповідають фрактальним властивостям при будь-якому масштабі. У природних фракталах існує обмеження на їхні розміри, при яких спостерігаються фрактальні властивості, тобто існує максимальний та мінімальний розміри області їхнього існування.

Геометричні фрактали побудовані на основі геометричних принципів (рис. 1.3), таких як самоподібність, звуження та інші. Вони можуть бути отримані шляхом відображення певних об'єктів у себе ж, зменшення їхнього масштабу та інших операцій. Прикладами можуть бути фрактальні гірки, які побудовані на основі самоподібної структури гірських порід.

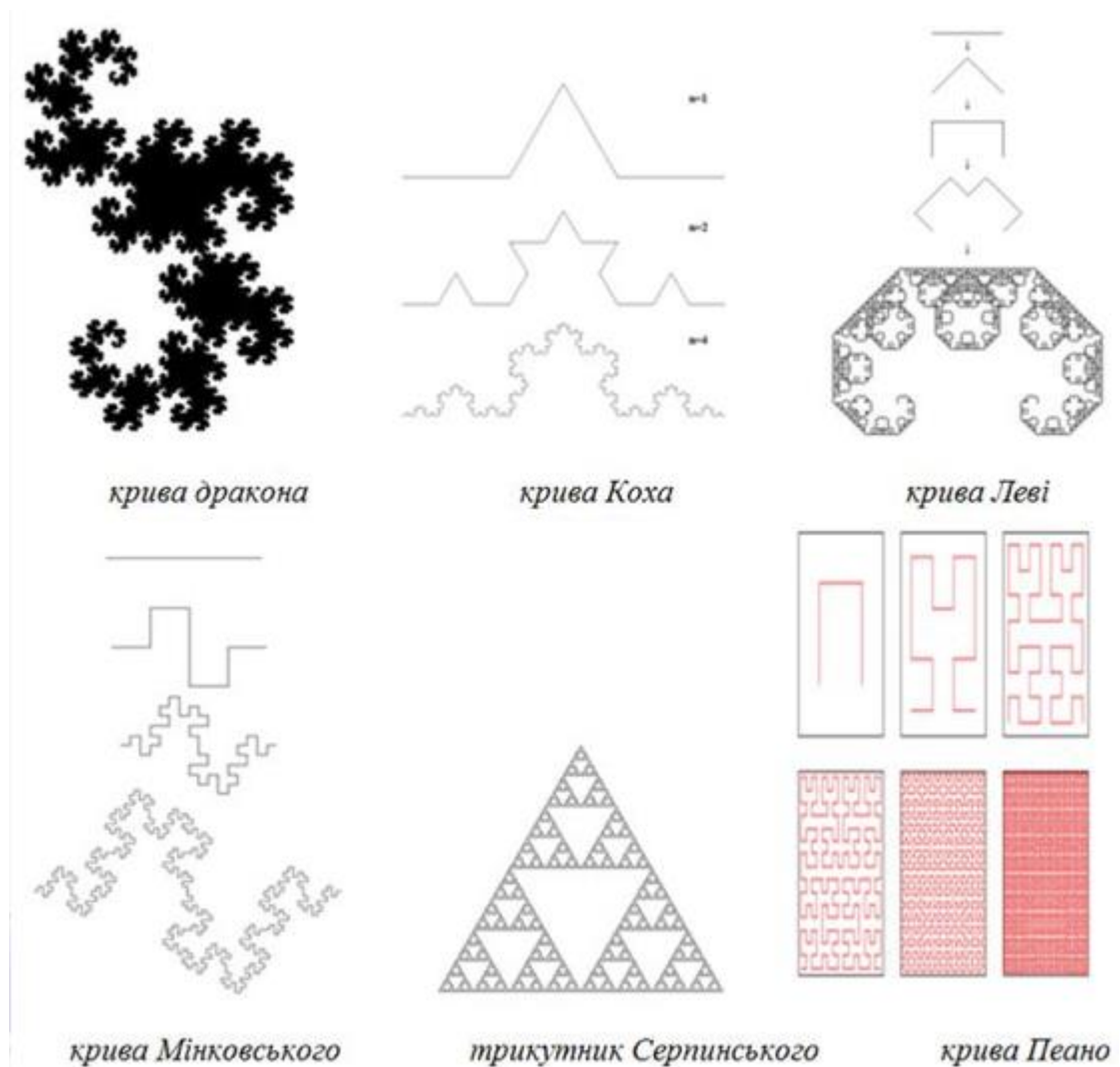


Рисунок 1.3 – Приклади геометричних фракталів

Алгебраїчні фрактали (рис. 1.4) базуються на ітераційному процесі над певними формулами, що складаються з алгебраїчних операцій, таких як додавання, множення і т. д. Прикладом може бути Серпінський трикутник, який можна отримати шляхом ітерації деякої формули над трикутником.

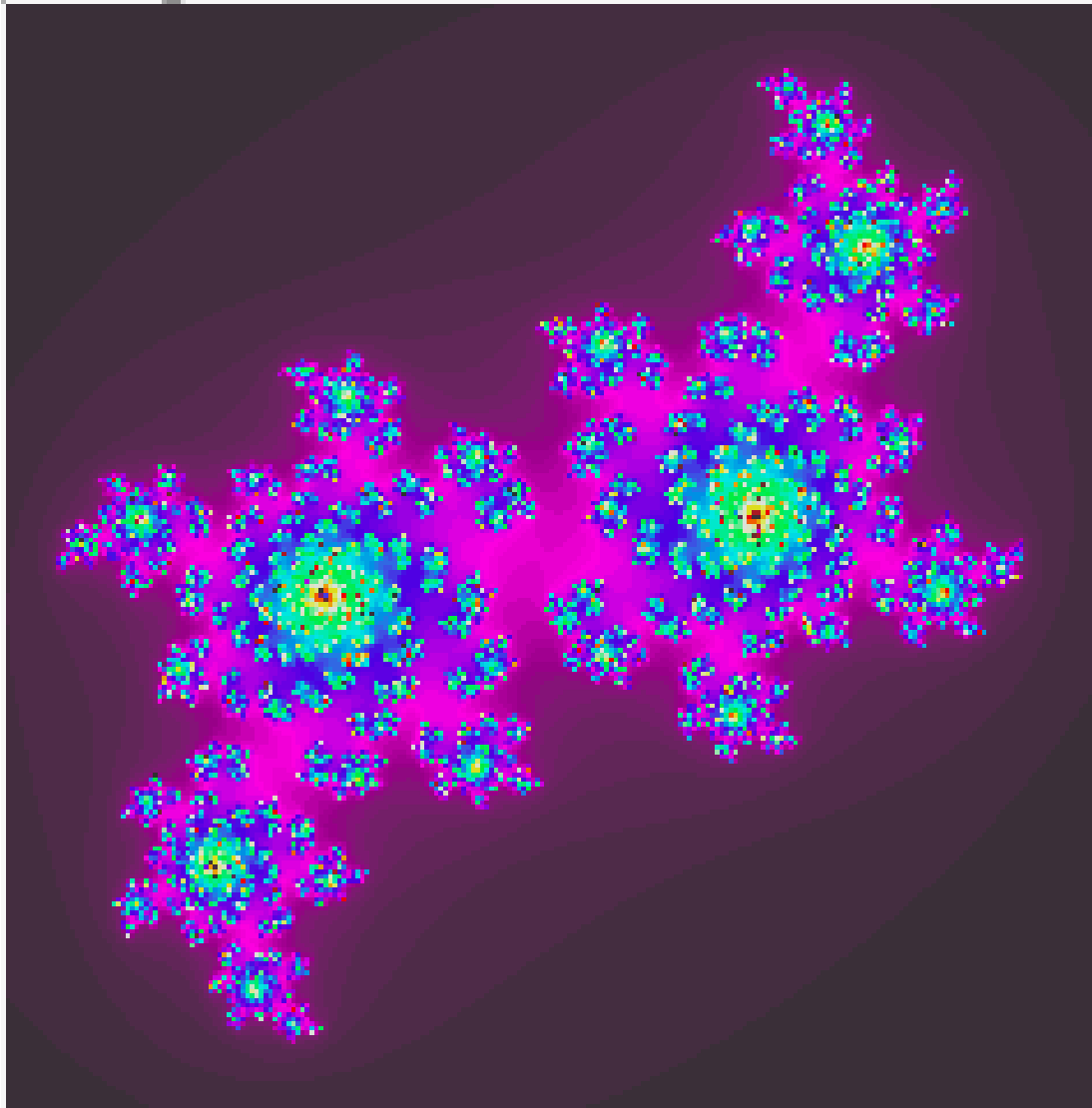


Рисунок 1.4 – Приклад алгебраїчного фракталу. Множина Жюліа

Стохастичні фрактали використовують випадкові процеси для побудови фрактальної структури. Одним з прикладів є фрактальний бріст, який може бути згенерований за допомогою випадкових процесів, що мімікують ріст кристалів.

Класифікація фракталів за їхньою математичною природою допомагає краще розуміти характеристики та властивості різних фрактальних структур, що може бути корисним у різних областях, таких як комп'ютерна графіка, аналіз даних та інше.

1.5 Постановка задачі

Задача обробки слабоструктурованих зображень є актуальною в багатьох галузях, таких як медицина, промисловість, транспорт, розваги та багато інших. Завдяки технологічному розвитку, з'явилася можливість отримувати велику кількість зображень, що потребують обробки та аналізу.

Об'єктом роботи є застосунок для аналізу слабоструктурованих зображень.

Метою даної кваліфікаційної роботи є створення зручного та ефективного інструменту, який дозволяє користувачам проводити аналіз та обробку зображень.

Для досягнення мети необхідно вирішити наступні завдання:

- провести аналіз існуючих методів обробки слабоструктурованих зображень та аналізу контурів;
- розробити алгоритми детектування країв та виявлення контурів на слабоструктурованих зображеннях;
- розробити алгоритми знаходження геометричних примітивів, таких як лінії, криві та інші важливі ознаки, на основі обробки слабоструктурованих зображень;
- реалізувати комп'ютерний застосунок, який використовує розроблені алгоритми для аналізу слабоструктурованих зображень;
- провести експериментальне тестування розробленого застосунку на різних наборах слабоструктурованих зображень та оцінити його ефективність та точність аналізу.

Вирішення цих завдань дозволить розробити ефективний застосунок, який забезпечує аналіз та виявлення геометричних структур на слабоструктурованих зображеннях.

2 МАТЕМАТИЧНЕ ОБГРУНТУВАННЯ ОБРАНИХ МЕТОДІВ

2.1 Усунення шумів їх класифікація і їх методи усунення

Усунення шумів є важливою складовою в обробці зображень, оскільки шум може призвести до спотворення результатів аналізу та зменшення якості зображення.

Шуми на зображеннях можуть бути різного типу та походження. До основних видів шумів належать:

– білий шум (рис. 2.1), що виникає через додання випадкових значень з гаусового розподілу до кожного пікселя зображення;

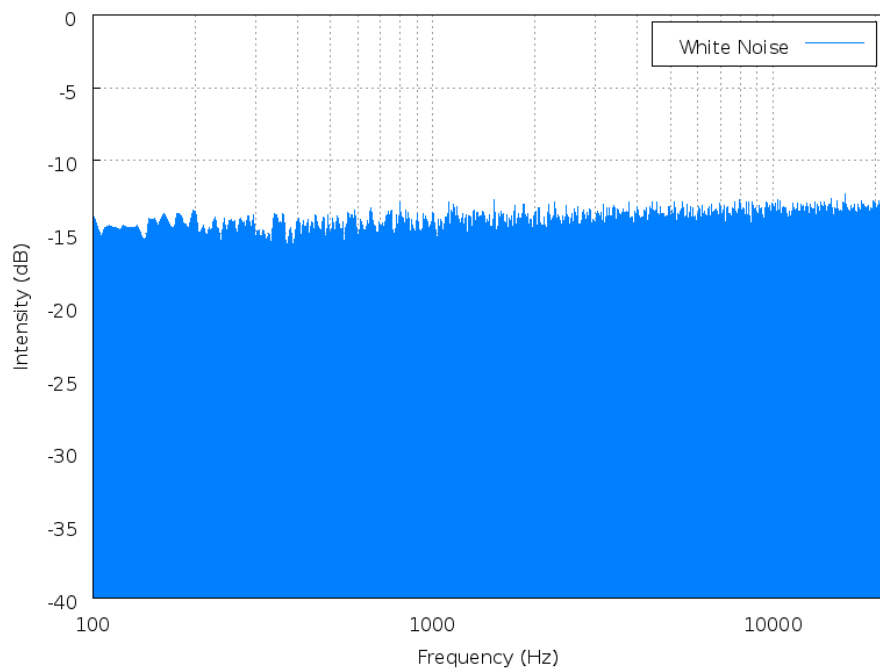


Рисунок 2.1 – Білий шум

– шум соль і перець (рис. 2.2), який може виникати внаслідок випадкового додавання дуже яскравих (білих) або дуже темних (чорних) пікселів до зображення;



Рисунок 2.2 – Шум соль та перець

– шум Пуассона (рис. 2.3), що виникає при фотографуванні з обмеженою кількістю світла, наприклад, у вуличних фотографіях;

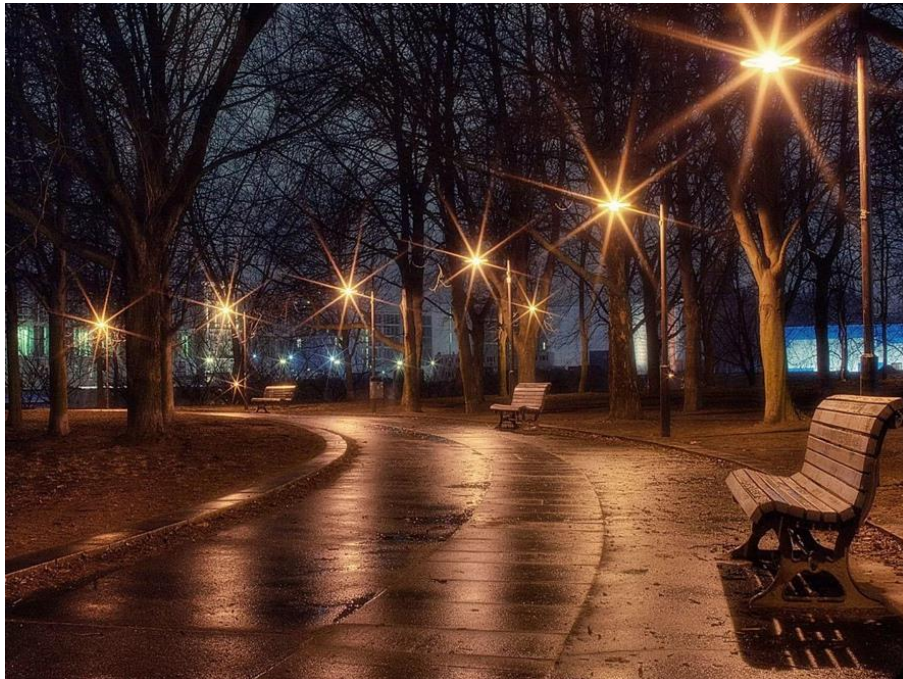


Рисунок 2.3 – Шум Пуассона

– шум від апаратного обладнання, такий як кольоровий шум, який може виникнути через дефекти в сенсорах камери.

Для усунення шумів на зображеннях існують різні методи, серед яких:

– фільтрація зображення (рис. 2.4), застосування фільтрів до зображення для видалення шумів;

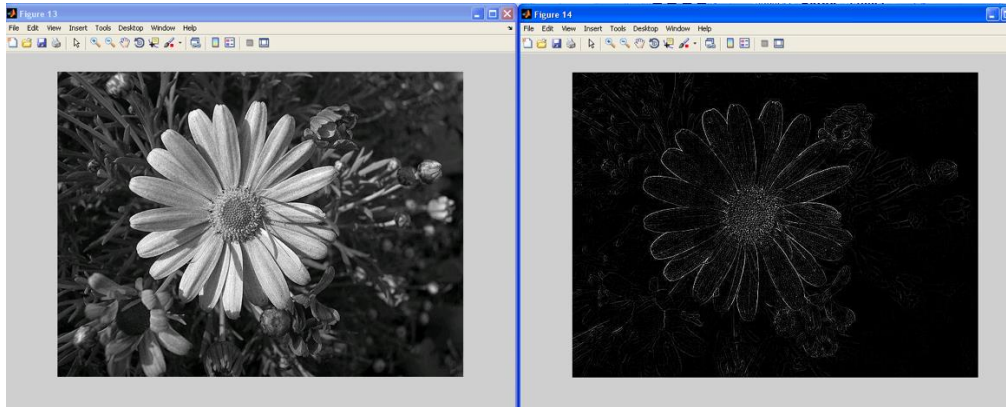


Рисунок 2.4 – Фільтрація зображень

– медіанна фільтрація (рис. 2.5), використання медіани вмісту області замість середнього значення;

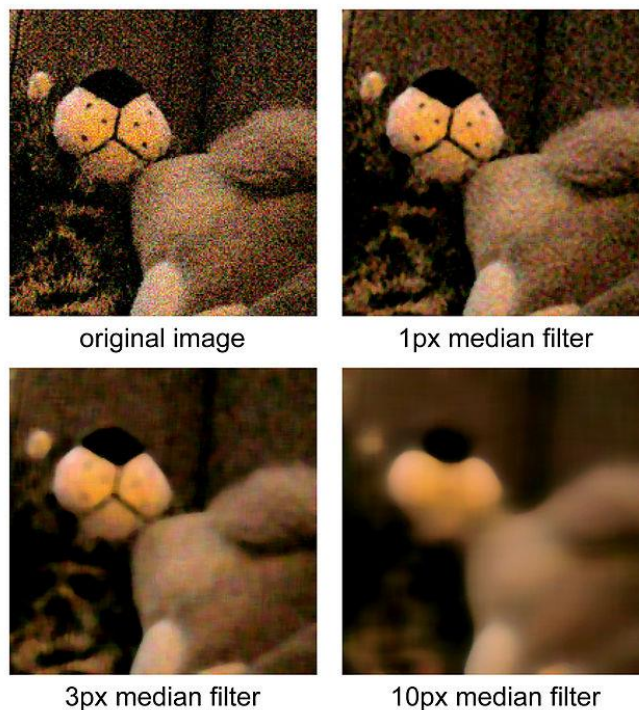


Рисунок 2.5 – Медіанна фільтрація зображень

– фільтрація за допомогою хвильового перетворення (рис. 2.6), використання хвильового перетворення для розбиття зображення на декілька частин, що можна фільтрувати окремо.

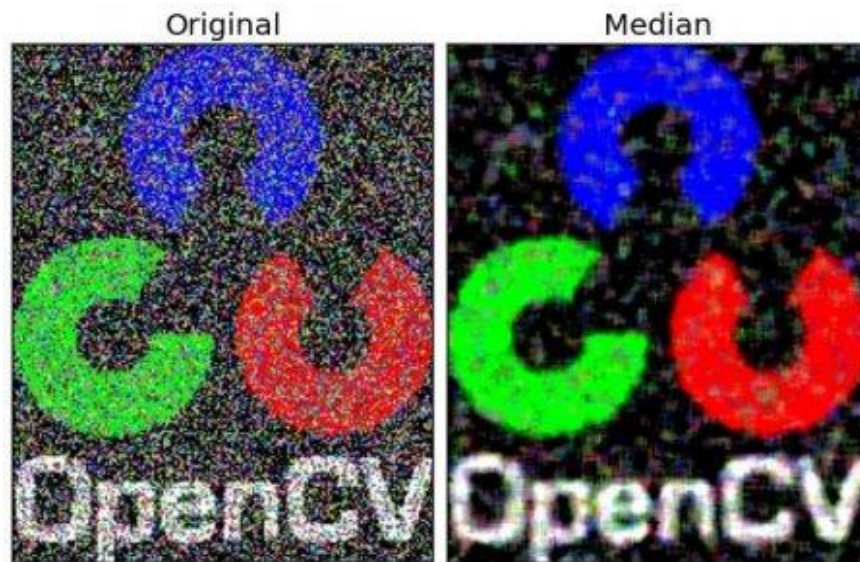


Рисунок 2.6 – Фільтрація за допомогою хвильового перетворення

Кожен метод має свої переваги та недоліки, і вибір методу залежить від конкретної задачі та характеру шуму на зображенні.

2.2 Робастність. Дослідження викидів і стійкість до перешкод

Робастність (англ. *robustness*) – це властивість алгоритмів обробки даних, що вказує на стійкість їх роботи при наявності викидів або інших непередбачуваних аномалій в даних [7].

В контексті обробки зображень, робастність може включати в себе стійкість до різних видів шумів, різних типів артефактів і збурень зображень, а також стійкість до різних видів перешкод, наприклад, до зміщень об'єктів або часткового закриття областей зображення.

Одним із способів забезпечення робастності алгоритмів обробки зображень є використання методів, які враховують розподіл даних і можуть

працювати з викидами або іншими аномаліями. Наприклад, методи медіани, які використовують медіану вмісту пікселів замість середнього значення, можуть бути більш стійкими до викидів.

Іншим способом забезпечення робастності є використання алгоритмів, які можуть автоматично виявляти і видаляти аномалії. Наприклад, методи найбільшої площі контуру можуть використовуватися для автоматичного видалення шумів та інших аномалій в зображенні.

Ще одним способом забезпечення робастності є використання методів статистичної обробки даних, таких як робастна статистика. Ці методи можуть бути використані для виявлення викидів і інших аномалій в даних та їх відкидання з обробки.

2.3 Гаусовський «білий» шум

Гаусівський шум є одним з найпоширеніших видів шуму в обробці сигналів і зображень. Він також відомий як «білий» шум, оскільки його спектральна щільність потужності є рівномірною на всьому діапазоні частот. Гаусівський шум є стохастичним процесом, який можна описати за допомогою нормального розподілу.

Гаусівський шум може бути доданий до зображень з різним рівнем інтенсивності. Його ефект на зображення полягає в тому, що він додає випадкові значення пікселів, що призводить до спотворення зображення. Зазвичай, чим більше інтенсивність шуму, тим більше він впливає на якість зображення.

Для усунення Гаусівського шуму існують різні методи, наприклад, фільтри Гауса, медіанні фільтри, фільтри Калмана та інші. Фільтр Гауса згладжує зображення, зменшуючи контрастність і деталі зображення, але зберігаючи основні риси зображення. Медіанний фільтр працює краще для видалення шумів у випадку, коли Гаусівський шум має велику дисперсію.

У загальному випадку, гаусівський шум є складним завданням для усунення. Оскільки він може бути доданий до зображень з різним рівнем інтенсивності, іноді важко відрізнити шум від корисної інформації. Тому, вибір методу усунення шуму повинен бути обраний з урахуванням конкретної ситуації та потреб користувача.

2.4 Чергування чорних та білих частинок (шум солі та перцю)

Шум солі та перцю є одним з типів шуму, який може виникати на цифрових зображеннях. Він полягає в тому, що на зображенні деякі пікселі можуть бути замінені на чорні або білі точки. Цей шум може бути спричинений різними факторами, такими як пошкодження датчика зображення, недосконалість передачі даних або помилки в програмному забезпеченні.

Для усунення шуму солі та перцю можна використовувати фільтри. Наприклад, медіанний фільтр може бути ефективним для видалення такого типу шуму. Цей фільтр замінює кожен піксель на медіанне значення його околиці, що дозволяє видалити зображення білі або чорні точки, які не відповідають реальному зображенню.

Іншим методом усунення шуму солі та перцю є використання адаптивного медіанного фільтра. Цей фільтр використовує медіанне значення не околиці кожного пікселя, але різних розмірів блоків. Він спочатку обробляє малий блок пікселів, щоб визначити медіанне значення, і якщо це значення не відповідає дійсному зображенню, то розмір блоку збільшується і фільтр повторюється.

Цей метод дозволяє ефективно видаляти шум солі та перцю з зображень будь-якої роздільної здатності, включаючи високоякісні зображення з високою чіткістю та деталізацією.

2.5 Методи фільтрації зображень

Зображення, які мають низьку якість, низьку роздільну здатність або змінену форму можуть бути важкими для аналізу. Однак, інструменти фільтрації можуть використовуватись для покращення якості зображень, зменшення шуму, збільшення контрастності, а також для покращення інших характеристик зображень.

Методи фільтрації зображень можуть бути поділені на декілька категорій: лінійні та не лінійні фільтри, фільтри зміни розміру та масштабу, фільтри з урахуванням кольору та текстури.

Лінійні фільтри є найбільш поширеними. Основна ідея лінійного фільтрування полягає в тому, що кожний вихідний піксель формується як лінійна комбінація пікселів вихідного зображення, які перетинаються з областю фільтрування. Лінійні фільтри можуть бути розподілені на фільтри згладжування, фільтри підвищення різкості та фільтри виявлення границь.

Фільтри згладжування допомагають видалити шум з зображення, зменшити контрастність та покращити рівномірність яскравості. Фільтри згладжування можуть бути лійними та не лійними. Лінійні фільтри згладжування використовуються для підсилення рівномірності яскравості, зменшення шуму та зменшення контрастності.

Фільтри підвищення різкості використовуються для покращення роздільної здатності та візуальної якості зображення. Ці фільтри збільшують контрастність, додають різкості та розкривають деталі, які можуть бути затерті або занадто.

Аналіз слабоструктурованих зображень – це процес отримання інформації з зображень, які мають низьку якість або змінену форму. Фільтрація зображень є важливою складовою для поліпшення якості зображень та підвищення їхньої зрозумілості. Методи фільтрації зображень допомагають видалити шум, зменшити контрастність та покращити рівномірність яскравості [8].

Методи фільтрації зображень – це процес обробки зображень з метою видалення шуму, покращення якості та виділення важливих деталей зображення. Фільтрація зображень застосовується в багатьох галузях, таких як комп'ютерне зору, медицина, автоматизоване виробництво і багато інших.

Розглянемо основні методи фільтрації зображень.

– медіанний фільтр є одним із найпоширеніших методів фільтрації зображень. Він використовується для видалення шуму із зображень. Основна ідея полягає у заміні кожного пікселя на медіанну значення яскравості в околиці даного пікселя. Цей метод добре працює для видалення шуму, але може призвести до втрати деталей зображення;

– фільтр Гауса використовується для розмиття зображення та зменшення шуму. Він ґрунтується на математичній функції Гауса, яка описує розподіл ймовірності випадкової величини. Фільтр Гауса застосовується шляхом згортки зображення з ядром Гауса. Цей метод добре працює для видалення шуму та розмиття зображення, але може призвести до втрати деталей зображення;

– фільтр Собеля використовується для виділення меж на зображенні. Він заснований на обчисленні градієнта яскравості у кожній точці зображення. Фільтр Собеля застосовується шляхом згортки зображення з ядром Собеля. Цей метод добре працює для виділення кордонів на зображенні, але може призвести до помилкових виявлення кордонів;

– фільтр Лапласа використовується для виділення кордонів зображення. Він заснований на обчисленні другої похідної яскравості у кожній точці зображення. Фільтр Лапласа застосовується шляхом згортки зображення з ядром Лапласа. Цей метод добре працює для виділення кордонів на зображенні, але може призвести до помилкових виявлення кордонів;

– фільтр медіанної різниці використовується для виділення текстур на зображенні. Він заснований на обчисленні різниці медіанного значення яскравості у двох околицях кожної точки зображення. Фільтр медіанної різниці застосовується шляхом згортки зображення з ядром медіанної різниці.

Цей метод добре працює для виділення текстур на зображенні, але може призвести до втрати деталей зображення;

– фільтр Білатеральної фільтрації використовується для розмиття зображення та зменшення шуму, зберігаючи при цьому ребра та текстури. Він заснований на пакунку зображення з ядром, яке враховує як просторову, так і яскравість. Цей метод добре працює для розмиття зображення та зменшення шуму, зберігаючи при цьому ребра та текстури [9-13].

На закінчення методи фільтрації зображень є важливим інструментом обробки зображень. Кожен метод має свої переваги та недоліки, і вибір методу залежить від конкретного завдання обробки зображення.

Давайте розглянемо більш детально кожний фільтр.

Медіанний фільтр – це метод обробки зображень, який використовується для зменшення шуму на зображенні. Цей метод полягає в заміні кожного пікселя на медіану значень пікселів у вікні фільтра.

Вікно фільтра – це квадратна область пікселів, яка переміщується по всьому зображенню. Розмір вікна може бути різним, залежно від задачі обробки зображення. Зазвичай він складається з непарного числа пікселів, щоб було можливо визначити медіану.

Медіана – це значення, яке знаходиться в середині відсортованого масиву значень. Якщо відсортований масив має парну кількість елементів, то медіана визначається як середнє значення двох центральних елементів [14-16].

Медіанний фільтр є ефективним методом для зменшення шуму на зображенні, оскільки він зберігає різкість границь об'єктів на зображенні, в той час як інші методи фільтрації можуть згладжувати границі і знижувати якість зображення.

Однак, медіанний фільтр має деякі недоліки. Він не ефективний для зменшення шуму на зображеннях з великою кількістю шуму, оскільки медіана може бути зіскладнена для визначення в таких випадках. Також, медіанний фільтр може знижувати деталі на зображенні, якщо розмір вікна фільтра вибирається надто великим. У практичних застосуваннях медіанний фільтр

використовується для зменшення шуму на зображеннях, що отримуються з медичних зображень, зображень з відеоспостереження, зображень з астрономічних досліджень та інших областей.

Один з найпоширеніших методів фільтрації зображень – це фільтр Гауса. Фільтр Гауса є лінійним фільтром, який використовується для згладжування зображення. Він базується на математичній моделі Гаусівської функції розподілу.

Гаусівська функція розподілу є неперервною функцією, яка описує розподіл випадкових величин. Вона має форму колокола і характеризується двома параметрами – середнім значенням та дисперсією. Чим більше дисперсія, тим ширший колокол. Фільтр Гауса використовує гаусівську функцію розподілу для створення ядра фільтра. Ядро фільтра – це матриця, яка використовується для обробки зображення. Ядро фільтра містить вагові коефіцієнти, які відображають внесок кожного пікселя у згладжування зображення.

Ядро фільтра Гауса має форму круга або еліпса, і його розмір залежить від дисперсії Гаусівської функції розподілу. Чим більша дисперсія, тим більшим повинен бути розмір ядра фільтра. Процес фільтрації зображення фільтром гауса полягає в тому, що кожен піксель зображення замінюється на середнє значення пікселів, які оточують його в межах ядра фільтра. Це дозволяє згладити зображення та зменшити шум.

Фільтр Гауса є ефективним методом фільтрації зображень, який знаходить застосування в багатьох областях, таких як комп'ютерне зорове сприйняття, медична діагностика, обробка зображень в реальному часі та багато інших.

Метод фільтрації зображень Собеля фільтр – це один з найпоширеніших методів обробки зображень, який використовується для відшукування різниць в інтенсивності пікселів на зображенні. Цей метод був розроблений іменем двох американських вчених Роберта Собеля та Ірвіна Собеля, які вперше опублікували його в 1968 році.

Суть методу полягає в тому, що на вхідному зображенні застосовується диференційний оператор, який дозволяє визначити градієнти зображення. Градієнт зображення – це вектор, який вказує напрямок зростання інтенсивності на зображенні. Для розрахунку градієнтів зображення застосовуються два диференційних оператора, які називаються операторами Собеля: горизонтальний та вертикальний [17].

Горизонтальний оператор Собеля складається з матриці 3×3 (рис. 2.7) з ваговими коефіцієнтами, які використовуються для розрахунку зміни інтенсивності на зображенні в горизонтальному напрямку. Ця матриця має наступний вигляд:

$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

Рисунок 2.7 – Горизонтальний оператор Собеля

Подальша обробка градієнтних зображень може включати операції, такі як виділення границь, визначення орієнтації градієнта, детектування кутів та інші алгоритми комп'ютерного зору, які використовують інформацію про градієнти для різних цілей, наприклад, для сегментації об'єктів або виділення особливостей на зображенні.

Таким чином, метод з використанням операторів Собеля дозволяє отримати інформацію про градієнти зображення

На рисунку 2.8 зображений вертикальний оператор Собеля, який складається з аналогічної матриці, але вагові коефіцієнти змінені для розрахунку зміни інтенсивності на зображенні в вертикальному напрямку.

$$\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

Рисунок 2.8 – Вертикальний оператор Собеля

Для обчислення градієнта в кожній точці зображення матриця Собеля застосовується до кожного пікселя зображення. Для цього спочатку обчислюється сума творів значень пікселів навколо поточного пікселя на відповідні значення матриці Собеля. Потім, для обчислення градієнта по горизонталі та вертикалі, ці суми підсумовуються за відповідними напрямками. Після обчислення градієнта в кожній точці зображення можна виділити межі на зображенні. Для цього зазвичай використовують порогову фільтрацію, при якій всі пікселі з градієнтом вище певного порогу вважаються межами, а інші – фоном.

Метод фільтрації зображень Собеля має кілька переваг перед іншими методами виділення кордонів. По-перше, він є швидким та ефективним. По-друге, він дозволяє виділяти межі на зображенні з високою точністю. По-третє, він може бути легко адаптований для роботи із зображеннями різних розмірів та дозволів. Однак метод фільтрації зображень Собеля також

має деякі недоліки. По-перше, може призводити до появи шуму на зображенні. По-друге, він може не виділяти деякі межі на зображенні, особливо якщо вони мають складну форму або перебувають у слабо контрастних областях.

Фільтр Лапласа використовується для виявлення країв на зображенні та підсилення їх контрастності. Фільтр Лапласа є диференційним оператором другого порядку, який використовується для визначення другої похідної зображення. Це означає, що він виявляє зміни яскравості на зображенні, що вказують на наявність країв. Фільтр Лапласа може бути реалізований як 3×3 матриця, яка складається з восьми навколишніх пікселів та центрального пікселя. Значення центрального пікселя віднімається від суми значень восьми навколишніх пікселів. Це дає значення, яке вказує на наявність краю на зображенні. Фільтр Лапласа може бути застосований до зображення як одиничний фільтр, або він може бути застосований кілька разів для підсилення контрастності країв. Однак, використання фільтра Лапласа може призвести до появи шуму на зображенні, тому його застосовують з обережністю.

Узагальнюючи, фільтр Лапласа є ефективним інструментом для виявлення країв на зображенні та підсилення їх контрастності. Він може бути застосований як одиничний фільтр або кілька разів для підсилення контрастності країв. Однак, використання фільтра Лапласа може призвести до появи шуму на зображенні, тому його застосовують з обережністю.

Фільтр медіанної різниці – це метод фільтрації зображень, який використовується для видалення шуму зображення. Цей метод полягає в тому, що для кожного пікселя зображення визначається медіана різниці між значеннями пікселів у визначеній області навколо цього пікселя. Ця медіана потім використовується для заміни значення пікселя [18].

Для застосування фільтра медіанної різниці до зображення, спочатку визначається розмір області, яка буде використовуватися для визначення медіани різниці. Зазвичай цей розмір вибирається таким чином, щоб він був

достатньо великим, щоб включити деяку кількість пікселів, але не таким великим, щоб він включав занадто багато шуму. Після визначення розміру області, для кожного пікселя зображення визначається медіана різниці між значеннями пікселів у визначеній області навколо цього пікселя. Ця медіана потім використовується для заміни значення пікселя. Фільтр медіанної різниці є ефективним методом видалення шуму зображення, оскільки він дозволяє видалити шум, не впливаючи на деталі зображення. Однак, він може бути досить повільним для великих зображень, оскільки для кожного пікселя потрібно визначити медіану різниці між значеннями пікселів у визначеній області навколо цього пікселя.

Білатеральна фільтрація – це метод фільтрації зображень, який використовує дві функції ваги для зменшення шуму на зображенні. Перша функція ваги використовується для визначення ваги пікселів на основі їх відстані від центрального пікселя. Друга функція ваги використовується для визначення ваги пікселів на основі їх яскравості. Білатеральна фільтрація дозволяє зменшити шум на зображенні, не втрачаючи деталей. Це досягається за рахунок того, що фільтр враховує як яскравість пікселів, так і їх відстань від центрального пікселя. Якщо пікселі мають близьку яскравість та відстань до центрального пікселя, то вони отримують високу вагу, що дозволяє зберегти їх деталі. Якщо пікселі мають відмінну яскравість або відстань до центрального пікселя, то вони отримують низьку вагу, що дозволяє зменшити їх вплив на зображення. У білатеральній фільтрації важливо правильно вибрати параметри фільтрації, такі як радіус фільтра, який визначає відстань, на яку впливає фільтр, та параметри функцій ваги. Неправильний вибір параметрів може призвести до втрати деталей на зображенні або до збереження шуму.

Узагальнюючи, білатеральна фільтрація є ефективним методом фільтрації зображень, який дозволяє зменшити шум на зображенні. Вона може бути застосована до різних типів зображень та дозволяє покращити якість зображень, що були пошкоджені під час передачі або зберігання.

3 РЕАЛІЗАЦІЯ ПРОТОТИПУ ЗАСТОСУНКА

3.1 Figma. Основна інформація

Figma – це онлайн-інструмент для дизайну і прототипування, який був запущений в 2016 році. Його створили Ділан Філд та Еван Волф, які раніше працювали в Dropbox. Вони зрозуміли, що існуючі інструменти для дизайну не задовольняють потреби команд, які працюють над проектами разом, тому вирішили створити свій власний інструмент. На рисунку 3.1 зображено логотип компанії.



Рисунок 3.1 – Логотип Figma

Figma використовується для створення макетів, інтерфейсів, іконок, прототипів та інших елементів дизайну. Він дозволяє працювати в режимі реального часу, що дозволяє командам працювати над проектами разом, незалежно від місця розташування. Крім того, Figma має вбудовані інструменти для спілкування та коментування, що дозволяє командам легко обговорювати проекти та вносити зміни. На рисунку 3.2 зображено інтерфейс Figma.

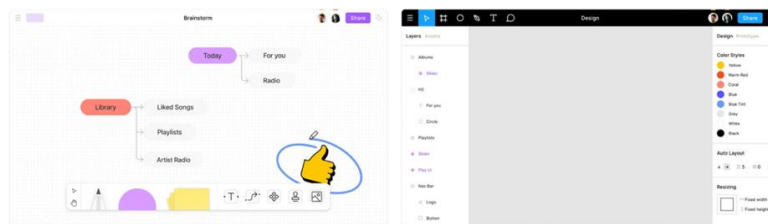


Рисунок 3.2 – Інтерфейс Figma

Зробивши висновки, можна сказати, що Figma – це потужний інструмент для дизайну та прототипування, який дозволяє командам працювати над проєктами разом в режимі реального часу. Його можливості для мобільного дизайну роблять його особливо корисним для дизайнерів, які працюють над проєктами для різних пристроїв [19].

Мобільний дизайн (рис. 3.3) є важливою складовою сучасного дизайну, оскільки мобільні пристрої, такі як смартфони і планшети, стали невід’ємною частиною нашого повсякденного життя. Оскільки мобільні пристрої мають обмежений екран і специфічні особливості, мобільний дизайн вимагає особливого підходу та оптимізації для забезпечення зручного і ефективного користувацького досвіду.

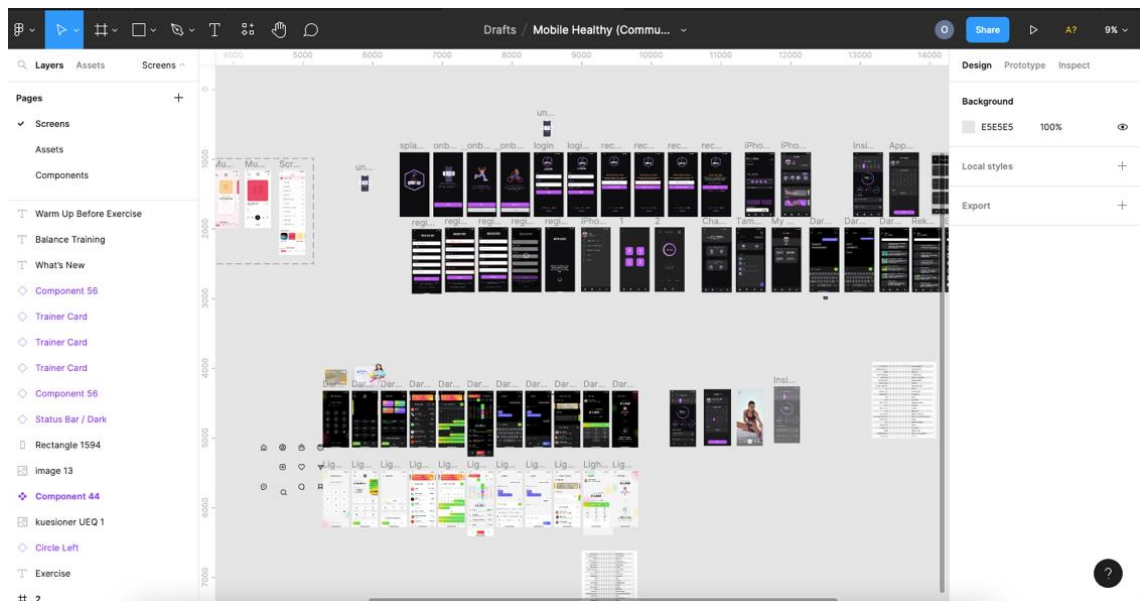


Рисунок 3.3 – Приклад дизайну мобільного застосунку

Figma є потужним інструментом для дизайну мобільних застосунків. Він має багато функцій, які дозволяють дизайнерам створювати високоякісні макети для мобільних пристроїв.

Одним з головних переваг Figma є його можливості для мобільного дизайну. Інструмент дозволяє створювати макети для різних розмірів екранів, що дозволяє дизайнерам працювати над проєктами для різних пристроїв. Крім

того, Figma має вбудовані інструменти для перевірки дизайну на різних пристроях, за допомогою цієї функції дизайнери можуть переконатися, що їхні проєкти виглядають належним чином на різних пристроях.

Одна з головних переваг Figma для мобільного дизайну – це можливість створювати адаптивні макети. Це означає, що дизайн може бути оптимізований для різних розмірів екранів, що дозволяє забезпечити користувачам зручний інтерфейс незалежно від того, на якому пристрої вони використовують застосунок. Figma також має багато інструментів для створення іконок, кнопок, меню та інших елементів інтерфейсу. Це допомагає дизайнерам створювати детальні макети, які допомагають розробникам розуміти, як повинен виглядати кінцевий продукт [20].

Ще одна надзвичайно важлива функція Figma для мобільного дизайну – це можливість створювати прототипи. Ця функція дозволяє дизайнерам створювати інтерактивні макети, які демонструють, як застосунок буде працювати в реальному житті.

За допомогою прототипів, дизайнери можуть відтворити взаємодію користувача з застосунком. Вони можуть створювати перехід між екранами, анімацію, взаємодію з елементами і багато іншого. Це дозволяє перевірити функціонал та ефективність застосунку перед його реалізацією. Прототипи в Figma є дуже потужним інструментом для спілкування з командою, замовниками та розробниками. Вони дозволяють всім зацікавленим сторонам відчувати функціональність застосунку та сприяють зрозумілості та злагоді у процесі розробки [21].

Зрештою, Figma є потужним інструментом для мобільного дизайну, який дозволяє дизайнерам створювати високоякісні макети та прототипи. Він має багато функцій, які допомагають забезпечити, що застосунок буде зручним та легким у використанні для користувачів.

3.2 Опис програмної реалізації

При відкритті застосунку відкривається сторінка завантаження застосунку (рис. 3.4): на цій сторінці зображено лого застосунку, користувач може завантажити застосунок на свій пристрій.



Рисунок 3.4 – Завантаження застосунку

Сторінка входу в особистий кабінет (рис. 3.5): користувач може увійти в свій особистий кабінет за допомогою облікового запису Google, Apple, Facebook або електронної пошти. Якщо облікового запису немає, то користувач може зареєструватися натиснув на кнопку.

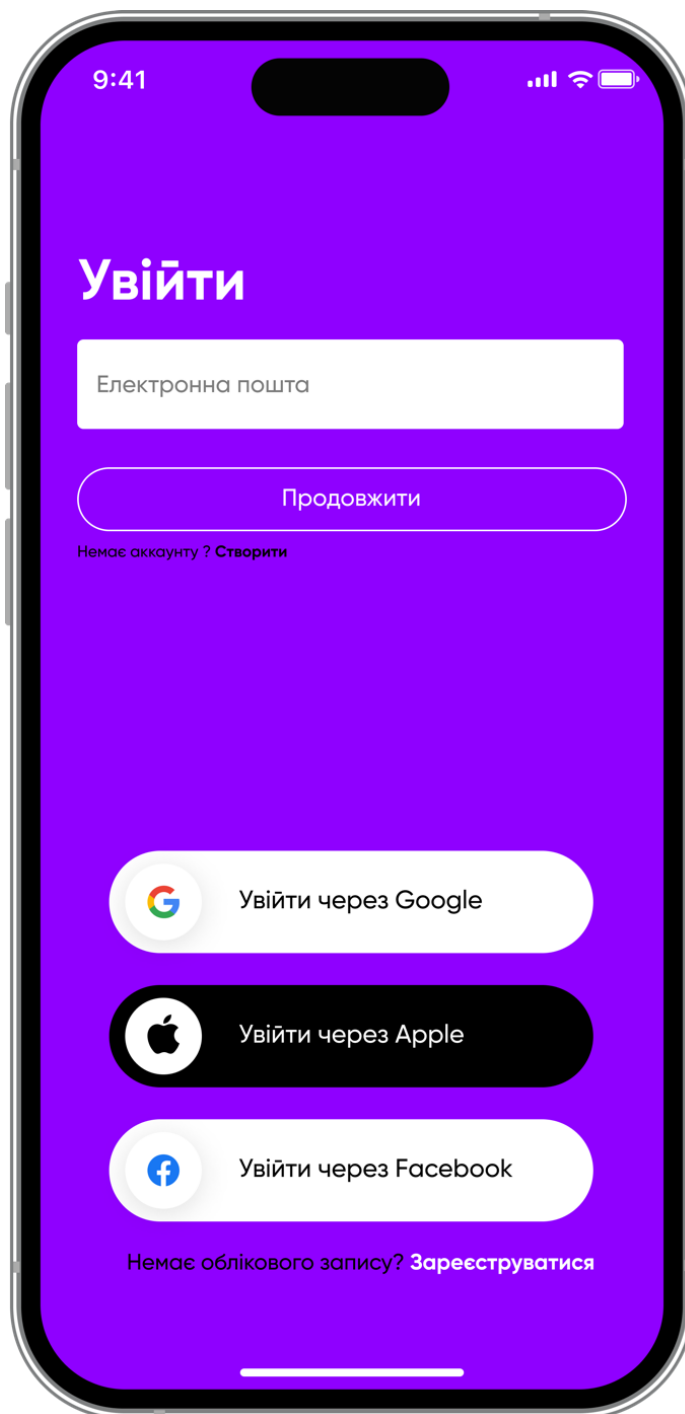


Рисунок 3.5 – Вхід у застосунок

Після того, як користувач обрав спосіб входу: Google, Apple, Facebook або електронна пошта – відкривається нова сторінка де користувачу потрібно буде ввести пароль від облікового запису (рис. 3.6).

Якщо клієнт забув пароль у нього є можливість встановити пароль і тоді відкриється наступна сторінка.

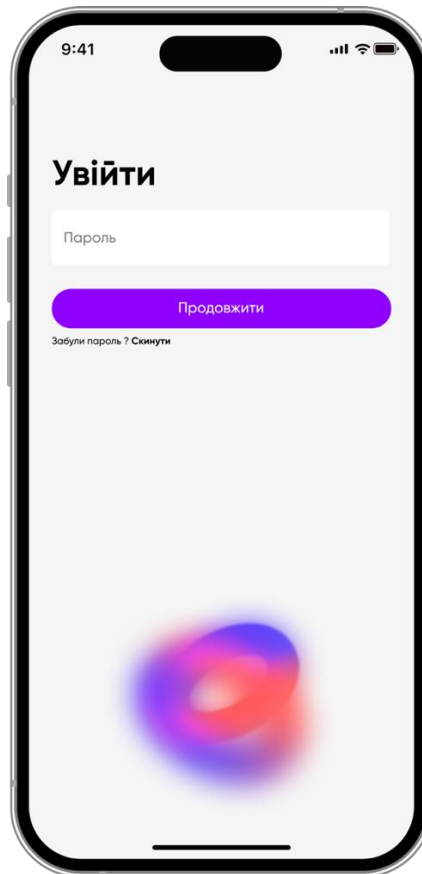


Рисунок 3.6 – Вхід у застосунок за допомогою пароля

На рисунку 3.7 показано відновлення пароля: якщо користувач забув свої дані для входу, він може відновити пароль на цій сторінці за допомогою електронної пошти, яку він використовував при реєстрації (рис. 3.7).

Можливість відновлення пароля є важливою функцією для багатьох застосунків з наступних причин.

Забезпечення доступу: відновлення пароля дозволяє користувачам знову отримати доступ до своїх облікових записів у випадку, якщо вони забули свій поточний пароль. Це важливо, оскільки без можливості відновлення пароля користувач може втратити доступ до своїх даних, облікового запису або послуг, пов'язаних з застосунком.

Безпека: можливість відновлення пароля забезпечує безпеку облікових записів користувачів. Замість того, щоб використовувати прості або легко запам'ятовувані паролі, користувачі можуть використовувати складні,

унікальні паролі, оскільки вони завжди матимуть можливість їх відновити в разі потреби.

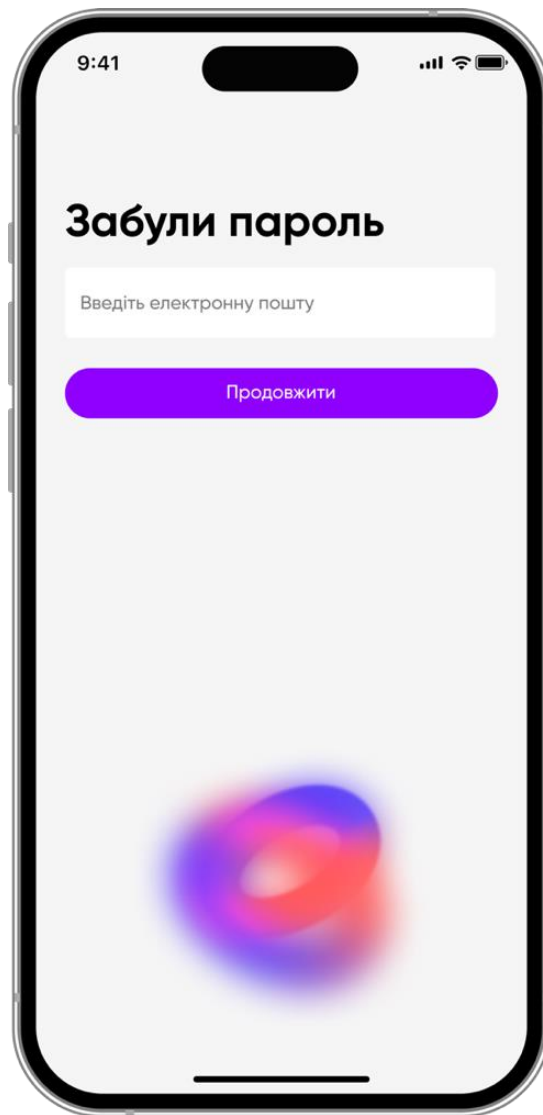


Рисунок 3.7 – Відновлення паролю

Захист від несанкціонованого доступу: можливість відновлення пароля допомагає захистити обліковий запис від несанкціонованого доступу. Коли користувачі забувають свої паролі, вони можуть запитати відновлення пароля, і процес відновлення може включати додаткові перевірки, такі як відправка підтверджувального коду на зареєстровану електронну пошту або номер телефону. Це допомагає переконатися, що тільки власник облікового запису може отримати доступ до нього [22-28].

На наступній сторінці відкривається інформація про те, що було надіслано електронний лист, щоб користувач міг змінити пароль (рис. 3.8).

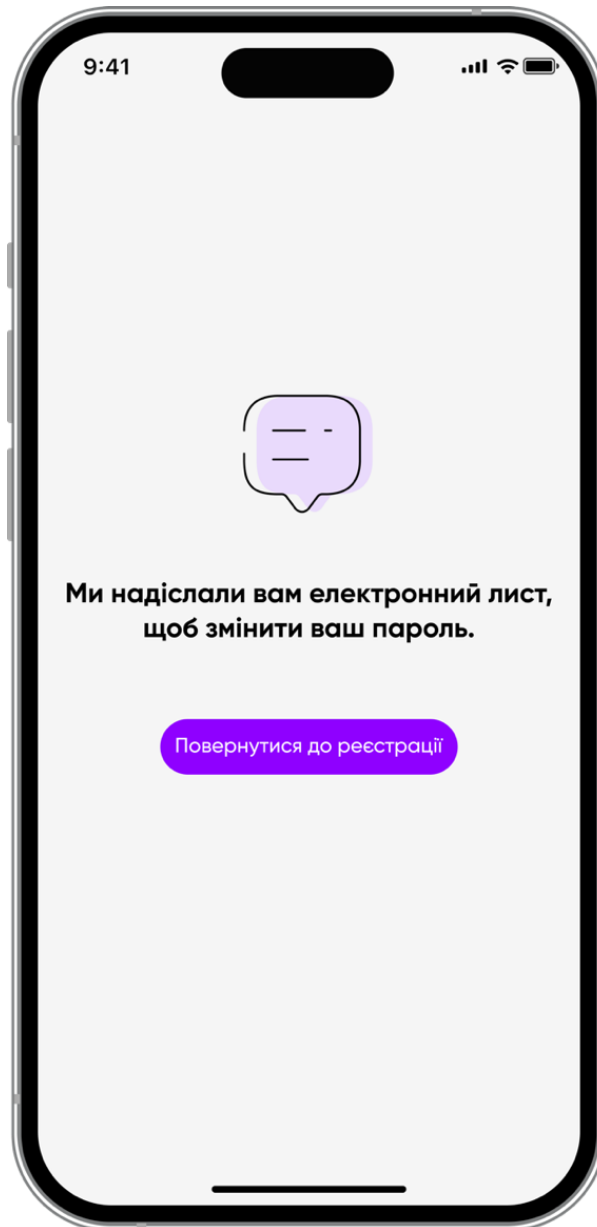


Рисунок 3.8 – Зміна паролю

На рисунку 3.9 користувач може переглянути налаштування: на цій сторінці клієнт може налаштувати свій профіль, пароль, встановити додаткові функції. Також користувач може подивитися свої попередні завантаження, оплатити додаткові функції, створити пароль для застосунка чи фейс айді.

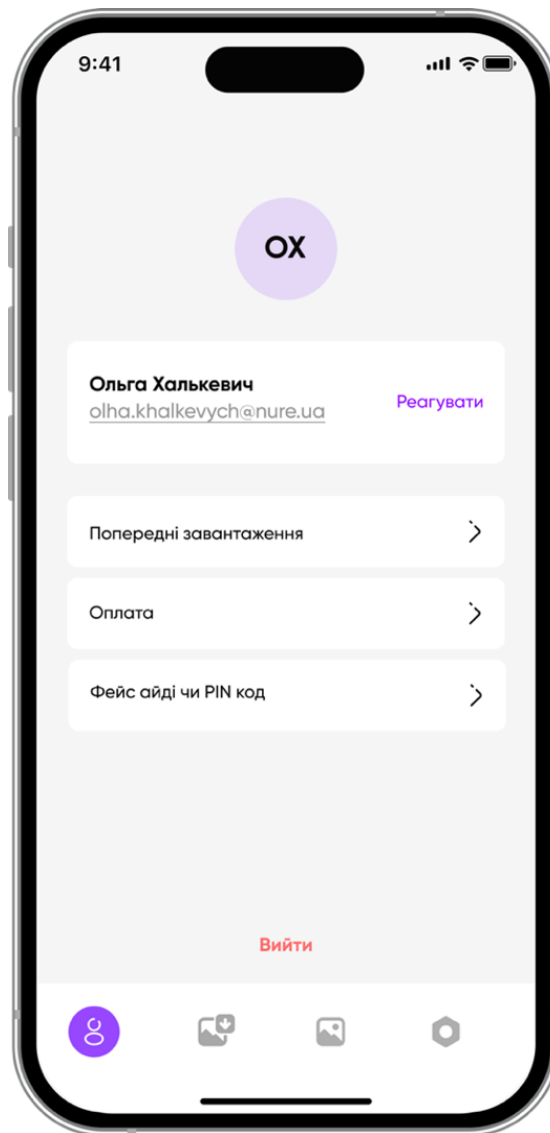


Рисунок 3.9 – Особистий кабінет

Сторінка попередніх завантажень (рис. 3.10): на цій сторінці користувач може переглядати свої минулі роботи.

Створення сторінки, де можна переглянути попередні завантаження фото, може бути корисною функцією для застосунків по обробці фотографій з наступних причин:

Зручність для користувачів: Сторінка з попередніми завантаженнями дозволяє користувачам легко оглядати і знаходити свої попередні фото. Вони можуть бачити, які зображення вони вже завантажили та обробили раніше, що полегшує повторну роботу з ними або нагадує про наявність попередніх версій фотографій [29].

Повторне використання: Часто користувачі хочуть застосовувати одні й ті ж фільтри до різних фотографій. Сторінка з попередніми завантаженнями дозволяє зручно вибирати попередні фото і повторно застосовувати до них обрані налаштування.

Візуальний контекст: Перегляд попередніх завантажень допомагає користувачам порівнювати різні версії оброблених фотографій. Вони можуть порівнювати зображення, редагування та вибирати найкращий варіант залежно від своїх потреб.

Управління пам'яттю: Сторінка з попередніми завантаженнями дозволяє користувачам керувати своїми зображеннями та видаляти непотрібні версії. Це допомагає звільнити простір на пристрої та підтримувати організованість у зберіганні фотографій [30].



Рисунок 3.10 – Попередні завантаження

Додавання сторінки з оплатою (рис. 3.11) та платними функціями в застосунку може мати декілька цілей та переваг:

- монетизація: Додавання платних функцій та можливості оплати дозволяє розробникам отримувати прибуток зі свого застосунку. Якщо застосунок надає унікальні або покращені фільтри та інструменти для обробки фотографій, користувачі можуть бути готові платити за них, особливо якщо вони допомагають досягти професійного або особистого успіху в обробці фото [31];

- посилення функціональності: Платні функції можуть пропонувати розширений набір фільтрів, ефектів, інструментів редагування або режими, що не доступні у безкоштовній версії застосунку. Це дозволяє задовольнити потреби більш вимогливих користувачів, які шукають більш глибокі та кращі можливості обробки своїх фотографій;

- ексклюзивний контент: Платні функції можуть також включати доступ до ексклюзивного контенту, такого як спеціальні фільтри, набори стилів або колекції фото. Це надає користувачам унікальну можливість створювати особливі та вражаючі оброблені зображення, які не знайдуть в інших застосунках або безкоштовних варіантах;

- підтримка розробки та оновлення: Отримання прибутку з платних функцій може забезпечити розробникам фінансові ресурси для підтримки та поліпшення застосунку.

Платні функції в застосунках є частим явищем в сучасному цифровому світі. Вони представляють собою додаткові можливості, які користувачі можуть отримати після оплати певної суми грошей або підписки на платформу.

Застосунки з платними функціями зазвичай надають можливість користувачам вибирати, які функції вони хочуть отримати та наскільки готові платити за них. Це дозволяє користувачам контролювати свої витрати та отримувати тільки ті функції, які їм дійсно потрібні.

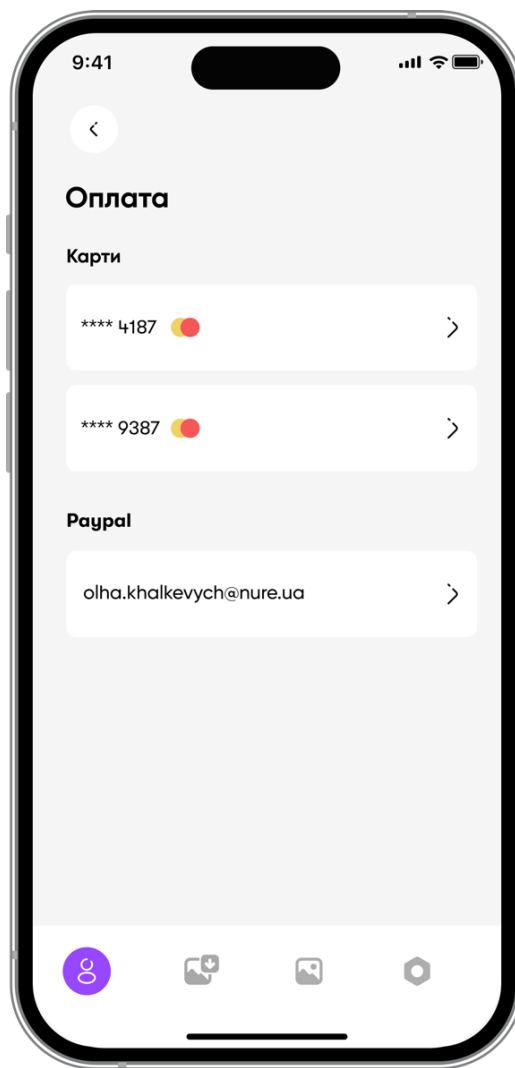


Рисунок 3.11 – Оплата

Для того, щоб користувач мав доступ до платних функцій, необхідно додати реквізити своєї банківської карти в застосунок. На рисунку 3.12 показано, як можна ввести банківські дані.

З метою зручності для наших клієнтів, надаємо можливість додати кілька банківських карт. Це дозволяє користувачам використовувати будь-яку з їх зареєстрованих карт для здійснення платежів та отримання платних функцій без необхідності повторного вводу даних.

Коли користувач вибирає опцію додати нову карту, він отримує можливість ввести необхідну інформацію про свою банківську карту, включаючи номер карти, дату закінчення терміну дії, ім'я власника та

CVV-код. Після вводу цих даних, користувач може зберегти карту у своєму профілі для майбутнього використання [32].

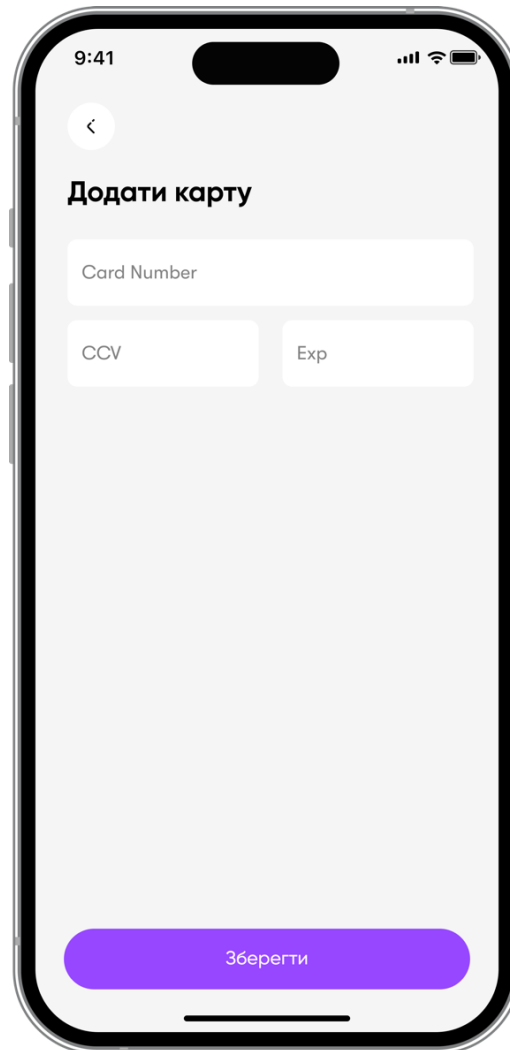


Рисунок 3.12 – Додати карту

Функція «Face ID» (рис. 3.13) або розпізнавання обличчя входу в застосунках має кілька цілей та переваг:

Зручність: Використання розпізнавання обличчя для входу до застосунку забезпечує зручність користувачам. Вони можуть уникнути необхідності введення пароля або PIN-коду, що займає час та може бути незручним на мобільних пристроях з малим екраном або в ситуаціях, коли руки зайняті.

Безпека: Розпізнавання обличчя може забезпечувати високий рівень безпеки для входу до застосунків. Кожне обличчя має унікальні риси, що

робить його важким для підробки або шахрайства. Таким чином, функція «Face ID» допомагає захистити дані та особисту інформацію користувача від несанкціонованого доступу.

Швидкість і ефективність: Розпізнавання обличчя може бути швидким та ефективним процесом входу до застосунків. Він зазвичай вимагає лише кілька секунд для розпізнавання обличчя, що дозволяє користувачам швидко отримувати доступ до застосунку без зайвих зусиль.

Технологічний прогрес: Використання функції «Face ID» є свідченням технологічного прогресу та інновацій. Розпізнавання обличчя є однією з передових технологій біометричної ідентифікації, яка надає користувачам новітній та сучасний спосіб входу до застосунків.



Рисунок 3.13 – Додати фейс айді

На рисунку 3.14 вхід за допомогою пароля з чотирьох цифр. У застосунках такий пароль може бути використаний з декількома цілями та перевагами:

Простота використання: Пароль з чотирьох цифр є простим і швидким способом входу до застосунків. Користувачам не потрібно запам'ятовувати складний пароль або виконувати довгий ввід символів. Введення чотирьох цифр може займати мінімальний час і забезпечувати швидкий доступ до застосунку.

Захист інформації: Використання пароля з чотирьох цифр є базовим рівнем захисту інформації. Хоча це не найбільш потужний вид захисту, він все ж допомагає запобігти несанкціонованому доступу до застосунку. Для багатьох користувачів це може бути достатньо для захисту їх особистих даних.

Компроміс між безпекою та зручністю: Пароль з чотирьох цифр є компромісом між безпекою та зручністю. Він забезпечує базовий рівень безпеки, але одночасно дозволяє швидкий вхід до застосунку без складнощів, пов'язаних з довгими паролями або біометричними методами ідентифікації.

Сумісність зі старішими пристроями: Застосунки, які підтримують вхід за допомогою пароля з чотирьох цифр, можуть бути сумісні зі старішими пристроями, які не мають розширених методів біометричної ідентифікації, таких як розпізнавання обличчя або відбитків пальців.

Простота відновлення доступу: У разі забуття пароля користувачі можуть легко відновити доступ до застосунку, зокрема шляхом використання процедури скидання пароля або отримання коду для відновлення доступу.

Сторінка вибору методів фільтрації зображень (рис. 3.15) є важливою складовою нашого застосунку. На цій сторінці користувач може обрати один з шести методів фільтрації зображень: медіанний фільтр, гаусівський фільтр, собеля фільтр, фільтр лапласа, фільтр медіанної різниці та фільтр білатеральної фільтрації.



Рисунок 3.14 – Pin код

Ця можливість надає користувачеві більшу гнучкість та контроль над обробкою їх зображень. Кожен з цих методів має свої унікальні властивості та ефекти, які можуть бути застосовані до зображення. Користувач може експериментувати з різними методами фільтрації, щоб досягти бажаного візуального ефекту або поліпшити якість зображення.

Наступною відкривається сторінка завантаження фото: після вибору методу фільтрації зображень на попередній сторінці, користувач може додати своє фото на цій сторінці (рис. 3.16).

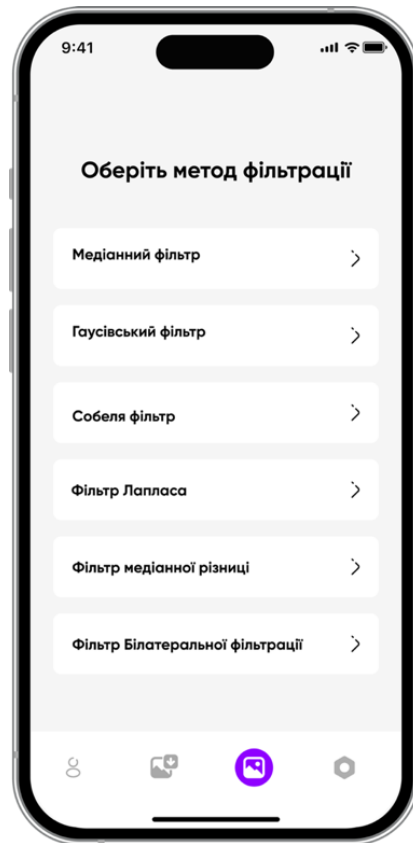


Рисунок 3.15 – Метод фільтрації

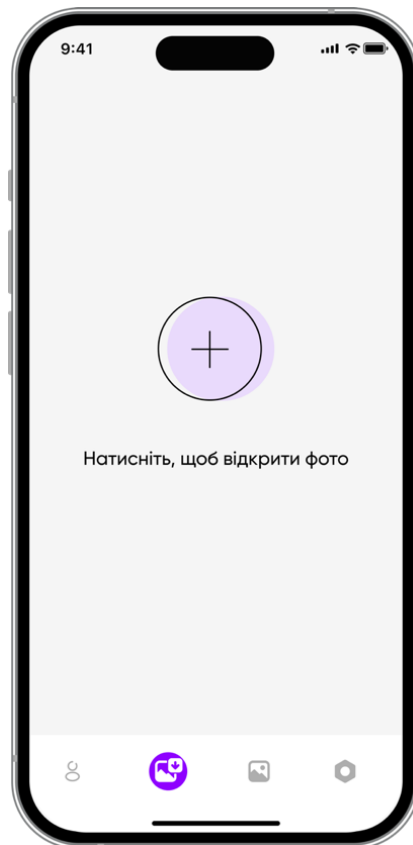


Рисунок 3.16 – Відкрити фото

Після натискання на кнопку відкривається особиста галерея користувача, на якій він обирає потрібне фото, яке він хоче покращити (рис.3.17).

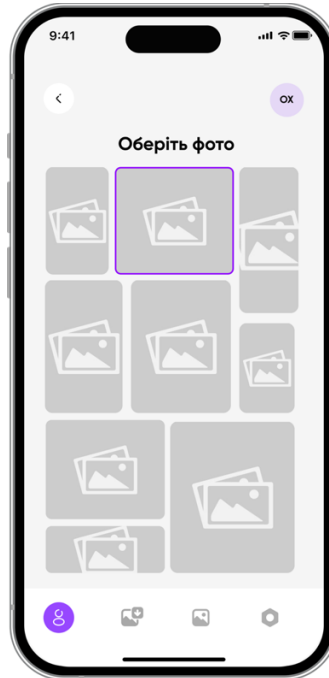


Рисунок 3.17 – Галерея користувача

На рисунку 3.18 зображена сторінка завантаження застосунку.

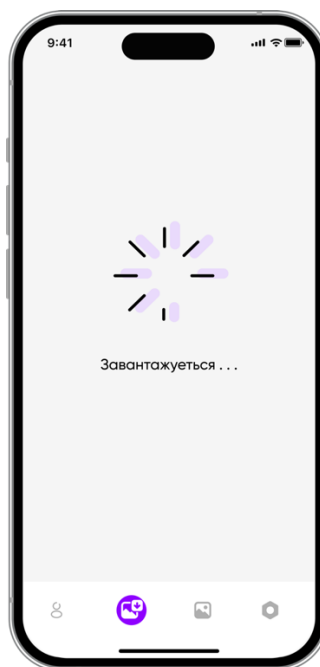


Рисунок 3.18 – Завантаження

Сторінка готового фото (рис. 3.19) є остаточним кроком в процесі обробки зображення в нашому застосунку. На цій сторінці користувач може переглянути готове фото з застосованим методом фільтрації та зберегти його у галерею.

Після вибору та застосування бажаного методу фільтрації на попередній сторінці, користувач може перейти на сторінку готового фото, де він зможе побачити результат своєї обробки. На цій сторінці відображається зображення з використаним методом фільтрації, щоб користувач міг оцінити вплив фільтру на вигляд фото.

Крім перегляду, користувач також має можливість зберегти готове фото у галерею свого пристрою. Це дозволяє зберегти оброблене зображення та поділитися ним з іншими людьми або використовувати його для подальших цілей, наприклад, друку або публікації в соціальних мережах.



Рисунок 3.19 – Готове фото

На рисунку 3.20 зображена сторінка інформації про методи фільтрації. На цій сторінці користувач може знайти детальну інформацію про кожен з шести методів фільтрації зображень.

Ми розуміємо, що користувачам важливо мати доступ до повної інформації про кожен метод фільтрації, щоб приймати обґрунтовані рішення щодо вибору найкращого методу для своїх потреб. На сторінці інформації про методи фільтрації надається докладний опис та пояснення щодо кожного методу. Користувач може дізнатися, як працює медіанний фільтр і які ефекти він має на зображення. Він також зможе дізнатися про властивості гаусівського фільтра та те, як він допомагає згладити зображення. Детальний опис надається для собея фільтра, фільтра лапласа, фільтра медіанної різниці та фільтра білатеральної фільтрації, щоб користувач міг зрозуміти їх особливості та вплив на зображення.

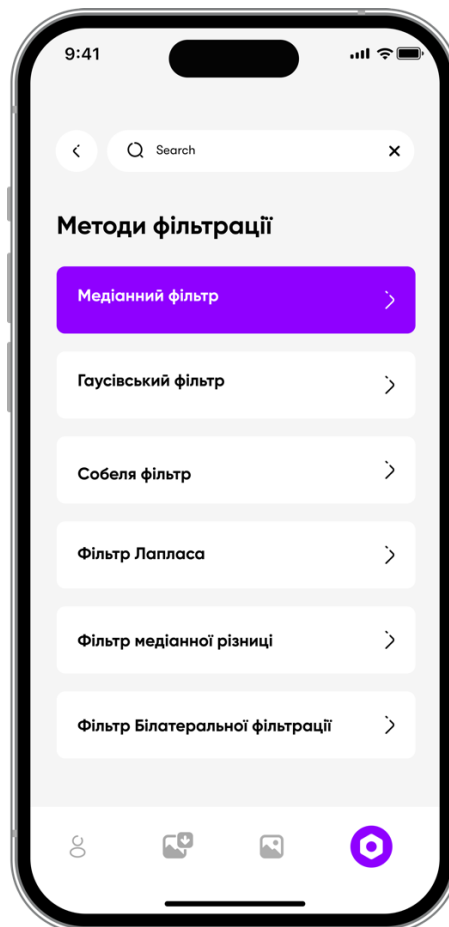


Рисунок 3.20 – Інформація про всі методи фільтрації

При натисканні на обраний метод фільтрації відкривається фото-приклад і детальна інформація про конкретний метод (рис. 3.21 – 3.26).

Для користувачів дуже важливо мати можливість бачити конкретні результати застосування кожного методу фільтрації на зображеннях. Тому, коли користувач натискає на обраний метод, відкривається фото-приклад, що демонструє, як виглядає зображення після застосування даного фільтру.

Крім фото-прикладу, також надається детальна інформація про обраний метод фільтрації. Користувач може ознайомитися з особливостями та принципами роботи методу, його впливом на зображення, а також перевагами та обмеженнями, які варто враховувати при його застосуванні.

Цей підхід дозволяє користувачам отримати більш глибоке розуміння кожного методу фільтрації та його потенційного впливу на їх зображення. Візуальні приклади разом з докладною інформацією створюють зручну та інформативну платформу для вибору найкращого методу фільтрації залежно від конкретних потреб користувача.



Рисунок 3.21 – Медіанний фільтр



Рисунок 3.22 – Гаусівський фільтр



Рисунок 3.23 – Собеля фільтр



Рисунок 3.24 – Фільтр Лапласа



Рисунок 3.25 – Фільтр медіаної різниці

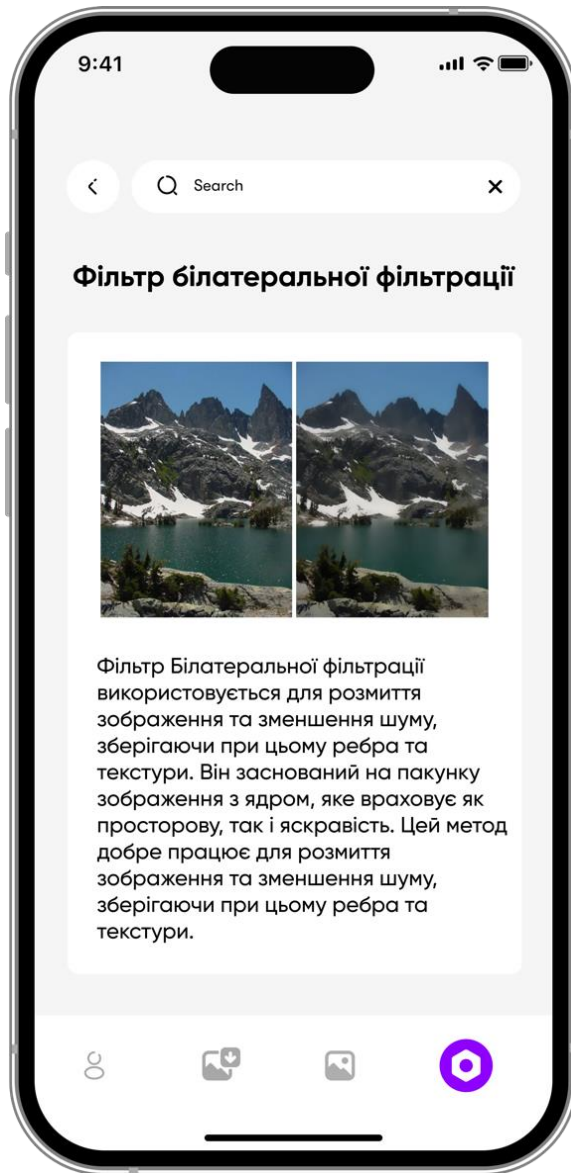


Рисунок 3.26 – Фільтр білатеральної фільтрації

ВИСНОВКИ

В рамках кваліфікаційної роботи було розроблено та реалізовано застосунок для аналізу слабоструктурованих зображень. Цей застосунок надає користувачам можливість використовувати різноманітні методи фільтрації зображень для поліпшення якості та вигляду фотографій.

Метою кваліфікаційної роботи було створення зручного та ефективного інструменту, який дозволяє аналізувати та обробляти слабоструктуровані зображення. Для досягнення цієї мети було використано сучасні технології та алгоритми обробки зображень. За допомогою цього застосунку користувачі зможуть застосовувати різноманітні фільтри, такі як медіанний фільтр, гаусівський фільтр, собеля фільтр, фільтр Лапласа, фільтр медіанної різниці та фільтр білатеральної фільтрації, для поліпшення зображень залежно від їх характеристик та потреб користувача.

Розроблений застосунок дозволяє легко завантажувати фотографії з різних джерел, таких як галерея, і застосовувати до них обрані методи фільтрації. Результат обробки відображається користувачеві, що дозволяє зробити відповідні корекції та зберегти готові зображення у галерею.

Висновок полягає в тому, що розроблений застосунок успішно виконує свою функцію аналізу та обробки слабоструктурованих зображень. Він надає широкий набір методів фільтрації, які дозволяють користувачам досягти бажаних результатів.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Sadek, S., Al-Hamadi, A., Michaelis, B., & Sayed, U. (2010). Efficient Region-Based Image Querying. arXiv preprint arXiv:1006.4588..
2. Sadek, S., Al-Hamadi, A., Michaelis, B., & Sayed, U. (2009, November). Cubic-splines neural network-based system for Image Retrieval. In 2009 16th IEEE International Conference on Image Processing (ICIP) (pp. 273-276). IEEE.
3. Li, D. C., & Fang, Y. H. (2008). An algorithm to cluster data for efficient classification of support vector machines. *Expert Systems with Applications*, 34(3), 2013-2018.
4. Perronnin, F. (2008). Universal and adapted vocabularies for generic visual categorization. *IEEE Transactions on pattern analysis and machine intelligence*, 30(7), 1243-1256.
5. Van Gemert, J. C., Geusebroek, J. M., Veenman, C. J., & Smeulders, A. W. (2008). Kernel codebooks for scene categorization. In *Computer Vision—ECCV 2008: 10th European Conference on Computer Vision, Marseille, France, October 12-18, 2008, Proceedings, Part III 10* (pp. 696-709). Springer Berlin Heidelberg.
6. Li, D. C., & Fang, Y. H. (2008). An algorithm to cluster data for efficient classification of support vector machines. *Expert Systems with Applications*, 34(3), 2013-2018.
7. Sadek, S., Al-Hamadi, A., & Michaelis, B. (2013). Toward real-world activity recognition: An SVM based system using fuzzy directional features. *WSEAS Trans Inf Sci Appl*, 10, 116-127.
8. Sadek, S., Al-Hamadi, A., Michaelis, B., & Sayed, U. (2012). A fast statistical approach for human activity recognition.
9. Sadek, S., Al-Hamadi, A., Michaelis, B., & Sayed, U. (2009). An efficient method for noisy cell image segmentation using generalized α -entropy. In *Signal Processing, Image Processing and Pattern Recognition: International Conference, SIP 2009, Held as Part of the Future Generation Information Technology*

Conference, FGIT 2009, Jeju Island, Korea, December 10-12, 2009. Proceedings (pp. 33-40). Springer Berlin Heidelberg.

10. Mohanalin, J., & Kumar, N. (2009). Tsallis entropy based microcalcification segmentation. *ICGST-GVIP Journal*, 9(1), 49-55.

11. Abe, S., & Okamoto, Y. (Eds.). (2001). *Nonextensive statistical mechanics and its applications* (Vol. 560). Springer Science & Business Media.

12. Sadek, S., Mofaddel, M. A., & Michaelis, B. (2013). Multicolor skin modeling with application to skin detection. *Journal of Computations & Modelling*, 3(1), 153-167.

13. Shannon, C. E., Weaver, W., & Wiener, N. (1950). The mathematical theory of communication. *Physics Today*, 3(9), 31-32.

14. Strzałka, D., & Grabowski, F. (2008). Towards possible q-generalizations of the Malthus and Verhulst growth models. *Physica A: Statistical Mechanics and its Applications*, 387(11), 2511-2518.

15. Singh, B., & Singh, A. P. (2008). Edge detection in gray level images based on the Shannon entropy. *Journal of Computer Science*, 4(3), 186.

16. Sadek, S., Al-Hamadi, A., Michaelis, B., & Sayed, U. (2011). An action recognition scheme using fuzzy log-polar histogram and temporal self-similarity. *EURASIP Journal on Advances in Signal Processing*, 2011, 1-9.

17. Sadek, S., Al-Hamadi, A., Michaelis, B., & Sayed, U. (2009, November). Robust image classification using multi-level neural networks. In *2009 IEEE International Conference on Intelligent Computing and Intelligent Systems* (Vol. 4, pp. 180-183). IEEE.

18. E Woods, R., & C Gonzalez, R. (2008). *Digital image processing*.

19. Sadek, S., Al-Hamadi, A., Michaelis, B., & Sayed, U. (2010, August). Towards robust human action retrieval in video. In *Proceedings of the British Machine Vision Conference (BMVC'10)*.

20. Sadek, S., & Abdel-Khalek, S. (2013). Generalized α -entropy based medical image segmentation. *Journal of Software Engineering and Applications*, 2014.

21. Bakheet, S., & Al-Hamadi, A. (2016). A discriminative framework for action recognition using f-HOL Features. *Information*, 7(4), 68.
22. Wang, L., Qiao, Y., & Tang, X. (2015). Action recognition with trajectory-pooled deep-convolutional descriptors. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4305-4314).
23. Худов, В. Г., Кучук, Г. А., Маковейчук, О. М., & Крижний, А. В. (2016). Аналіз відомих методів сегментування зображень, що отримані з бортових систем оптико-електронного спостереження. *Системи обробки інформації*, (9), 77-80.
24. Кобилін, О. А., & Творошенко, І. С. (2021). Методи цифрової обробки зображень.
25. Романюк, О. І. (2020). Дослідження методів та алгоритмів визначення границь зображень.
26. Hsu, C. N., & Dung, M. T. (1998). Generating finite-state transducers for semi-structured data extraction from the web. *Information systems*, 23(8), 521-538.
27. Lim, M. J., Kwon, Y. M., & Chung, D. K. (2016). Intelligent system to search user information via facial recognition using adaboost algorithm. *International Information Institute (Tokyo). Information*, 19(5), 1395.
28. Lu, W. L., Okuma, K., & Little, J. J. (2009). Tracking and recognizing actions of multiple hockey players using the boosted particle filter. *Image and Vision Computing*, 27(1-2), 189-205.
39. Bakheet, S., Al-Hamadi, A., & Mofaddel, M. A. (2017). Recognition of human actions based on temporal motion templates. *Br. J. Appl. Sci. Technol*, 20(5), 1-11.
30. Bodyanskiy, Y., Kobylin, I., Rashkevych, Y., Vynokurova, O., & Peleshko, D. (2018, February). Hybrid fuzzy-clustering algorithm of unevenly and asynchronously spaced time series in computer engineering. In *2018 14th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET)* (pp. 930-935). IEEE.

31. Bodyanskiy, Y., Vynokurova, O., Szymański, Z., Kobylin, I., & Kobylin, O. (2016, August). Adaptive robust models for identification of nonstationary systems in data stream mining tasks. In 2016 IEEE First International Conference on Data Stream Mining & Processing (DSMP) (pp. 263-268). IEEE.

32. Bodyanskiy, Y., Vynokurova, O., Kobylin, I., & Kobylin, O. (2016). Adaptive fuzzy clustering of short time series with unevenly distributed observations in Data Stream Mining tasks. *Information Technology and Management Science*, 19(1), 23-28.