МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Харківський національний університет імені В.Н.Каразіна
Факультет математики і інформатики
Кафедра теоретичної та прикладної інформатики

Кваліфікаційна робота

Бакалавр

на тему: **Розробка інтерактивних додатків «Математика навколо нас».**

Development of interactive applications "Mathematics around us"

Виконала: студентка 4 курсу, групи МФ-41
спеціальність 122 «Комп'ютерні науки»
освітньо-професійна програма
«Інформатика»
Васильєва Є.П.
Керівник: Бєлова Л.П.
Рецензент:

Харків – 2024 року

ЗМІСТ

ВСТУП 3
РОЗДІЛ 1. ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ 4
1.1 Огляд існуючих програм для обробки текстових зображень 4
1.2 Опис предметної області проекту 6
1.2.1 Визначення цифрового зображення 6
1.2.2 Текстові зображення 8
РОЗДІЛ 2. МОРФОЛОГІЧНА ОБРОБКА ЗОБРАЖЕНЬ 11
2.1 Основні поняття та визначення 11
2.1.1 Бінарні зображення 11
2.1.2 Структурний елемент 12
2.2 Основні морфологічні операції 14
2.2.1 Ерозія 14
1.2.2 Дилатації 16
2.2.3 Відкриття та закриття 18
РОЗДІЛ 3. РОЗРОБКА ПРОГРАМНОГО ПРОДУКТУ ТА ДЕМОНСТРАЦІЯ РЕЗУЛЬТАТІВ 23
3.1 Вибір інструментів розробки 23
3.1.1 Мова програмування MATLAB 23
3.1.2 Image Processing Toolbox 23
3.2 Архітектура розробленої програми 24
3.3 Демонстрація результатів роботи 28
ВИСНОВКИ 33
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ 35

ВСТУП

Обробка зображень - це величезна галузь з безліччю алгоритмів та методів, які використовуються для досягнення різноманітних результатів. Вона відіграє важливу роль у сучасному світі. З розвитком цифрових технологій збільшується вимога до якісних та зручних у використанні зображень з текстом. Інтерес до них зростає у документації, презентаціях, веб-сайтах, соціальних мережах та інших сферах. Однак, існують труднощі, пов’язані з якістю таких зображень, такі як наявність фонових об’єктів, шум та нерівномірність товщини ліній.

Вивчення літературних джерел, зокрема книги Р. Гонсалеса та Р. Вудса «Цифрова обробка зображень. М: Техносфера,2005», дозволило глибше зрозуміти методи математичної морфології, такі як дилатація, ерозія, звуження та розширення. Ці методи демонструють високу ефективність у покращенні текстових зображень, видаленні фонових об’єктів, підвищенні контрастності та рівномірності ліній.

Мета дипломної роботи полягає у вивченні та дослідженні методів покращення якості текстових зображень за допомогою математичної морфології.

Основне завдання полягає у створенні ефективних алгоритмів для видалення фону та регулювання товщини ліній з метою поліпшення якості та зручності використання таких зображень у різних галузях.

Результати цієї роботи можуть бути використані у різних галузях, де необхідна обробка текстових зображень. Розроблені алгоритми можуть бути використані у системах розпізнавання тексту, автоматизованій обробці документів та інших сферах, що потребують якісних текстових зображень. Створена програма автоматизує процес обробки текстових зображень, що дозволяє значно полегшити роботу з великою кількістю таких зображень.

РОЗДІЛ 1. ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Огляд існуючих програм для обробки текстових зображень

Для обробки текстових зображень існує багато програмних засобів, кожен з яких має свої переваги та недоліки. Розглянемо деякі з них:

1.Adobe Photoshop

Adobe Photoshop є однією з найпопулярніших програм для редагування зображень. Вона надає потужні інструменти для обробки текстових зображень, такі як фільтри, маски та інструменти корекції кольору. Однак, Photoshop має деякі обмеження:

- **Складність використання**: Для новачків Photoshop може бути складним у використанні через велике число функцій та інструментів.
- **Часовитратність**: Обробка великої кількості зображень може бути тривалою через відсутність автоматизації для специфічних задач.
- **Вартість**: Photoshop є комерційним продуктом з досить високою вартістю ліцензії.

2.GIMP (GNU Image Manipulation Program)

GIMP є безкоштовною та відкритою альтернативою Photoshop. Вона також надає широкий спектр інструментів для редагування зображень:

- **Безкоштовність**: GIMP є безкоштовним програмним забезпеченням з відкритим кодом.
- **Гнучкість**: Можливість налаштування та розширення через плагіни.
- **Недоліки**: Як і у випадку з Photoshop, обробка великої кількості зображень може бути тривалою через ручну роботу.

3.CorelDRAW

CorelDRAW є ще однією комерційною програмою для редагування зображень, яка надає інструменти для обробки текстових зображень:

- **Інтуїтивний інтерфейс**: Легкість у використанні порівняно з Photoshop.
- **Обмеження**: Висока вартість ліцензії та обмежена автоматизація.

4.Python

Python є потужною мовою програмування, яка має багато бібліотек для обробки зображень, таких як OpenCV, Pillow та Scikit-image:

- **OpenCV**: Бібліотека для обробки зображень, яка надає широкий спектр інструментів для роботи з текстовими зображеннями, включаючи методи математичної морфології.
- **Pillow**: Бібліотека для роботи з зображеннями, яка дозволяє виконувати базові операції з обробки зображень.
- **Scikit-image**: Бібліотека, що надає розширені можливості для аналізу та обробки зображень.

Використання Python для обробки зображень має такі переваги:

- **Автоматизація**: Можливість автоматизації процесів обробки зображень.
- **Гнучкість**: Можливість розробки спеціалізованих алгоритмів під конкретні задачі.
- **Відкритий код**: Більшість бібліотек є відкритими та безкоштовними.

Недоліки:

- **Потреба в програмуванні**: Для використання Python необхідні навички програмування.
- **Продуктивність**: Обробка великих зображень може бути повільною без оптимізації коду.

5.Вбудовані сканери документів на iPhone та Android

Сучасні смартфони, як iPhone та Android, оснащені вбудованими сканерами документів, які дозволяють легко створювати текстові зображення за допомогою камери.

- **iPhone:** Вбудований сканер документів у додатку "Notes" дозволяє сканувати документи, автоматично виявляючи краї паперу, усуваючи тіні та коригуючи перспективу. Недоліком є обмеженість в можливостях редагування після сканування.

- **Android:** Додаток "Google Drive" має функцію сканування документів, яка також автоматично виявляє краї, коригує перспективу та усуває тіні. Недоліком може бути недостатня точність автоматичних коригувань у складних умовах освітлення.

1.2 Опис предметної області проекту

1.2.1 Визначення цифрового зображення

Цифрове зображення - це візуальне відображення, яке можна зберігати та обробляти за допомогою комп’ютера. Воно складається з масиву пікселів, кожен з яких має власне значення яскравості та кольору. Розмір зображення визначається кількістю пікселів по вертикалі (висота) та горизонталі (ширина). Наприклад, зображення розміром 1920×1080 пікселів містить у собі всього 2 073 600 пікселів.

Цифрові зображення можуть бути представлені у різних форматах, таких як растрові та векторні. Растрові зображення складаються з пікселів і є найбільш поширеними для фотографій та сканованих документів. Векторні зображення, на відміну від растрових, складаються з геометричних примітивів, таких як лінії та криві, і використовуються переважно для графіки та малюнків.

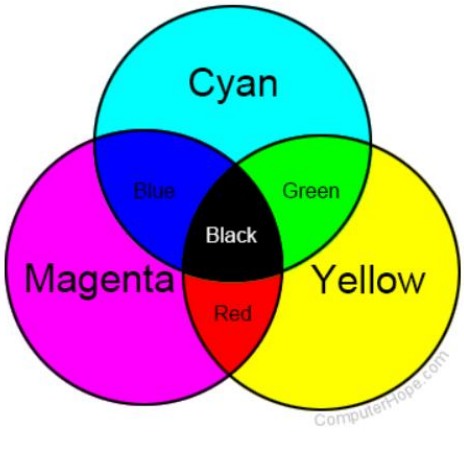
Піксель – найменша одиниця цифрового зображення, який характеризується кольором в моделі RGB або в альтернативній моделі HSB відтінком, насиченістю та яскравістю. Дві кольорові моделі взаємозамінні, є алгоритми, що дозволяють перетворювати зображення в іншу модель представлення кольорів. RGB-піксель складається з 3-х цілих чисел від 0 до 255, що представляють інтенсивність червоного, зеленого та синього кольорів.

Кольорові моделі

Цифрові зображення можуть бути монохромними (чорно-білими) або кольоровими. Для представлення кольорових зображень використовуються різні кольорові моделі. Найпоширенішими з них є:

- ***RGB (Red, Green, Blue):*** Кожен піксель представлений трьома значеннями, що відповідають інтенсивностям червоного, зеленого та синього кольорів. Комбінуючи ці три основні кольори в різних пропорціях, можна отримати будь-який колір.
- ***CMYK (Cyan, Magenta, Yellow, Black):*** Використовується переважно для друку. Кожен піксель описується чотирма значеннями, що відповідають основним друкарським фарбам.
- ***Grayscale*** (Відтінки сірого): Кожен піксель представлений одним значенням яскравості, що варіюється від чорного до білого.

R	V	B	Couleur
0	0	0	noir
0	0	1	nuance de noir
255	0	0	rouge
0	255	0	vert
0	0	255	bleu
128	128	128	gris
255	255	255	blanc



Роздільна здатність і якість зображення

Роздільна здатність цифрового зображення визначає його якість та детальність і вимірюється в пікселях на дюйм (ppi - pixels per inch). Чим вища роздільна здатність, тим більш детальним і чітким буде зображення. Наприклад, зображення з роздільною здатністю 300 ppi є більш якісним, ніж зображення з роздільною здатністю 72 ppi.

Формати зображень

Існує багато різних форматів цифрових зображень, кожен з яких має свої особливості і використовується в залежності від конкретних потреб:

- ***JPEG (Joint Photographic Experts Group):*** Широко використовується для фотографій та веб-графіки завдяки хорошому компромісу між розміром файлу та якістю зображення. Використовує стиснення з втратами.
- ***PNG (Portable Network Graphics):*** Використовується для зображень з прозорістю та графіки, де важливо зберегти високу якість. Стиснення без втрат.
- ***GIF (Graphics Interchange Format):*** Підтримує анімацію та використовується для простих графічних зображень з обмеженою палітрою кольорів.
- ***TIFF (Tagged Image File Format):*** Використовується для зберігання зображень з високою якістю без стиснення або зі стисненням без втрат. Часто застосовується в професійній фотографії та друкарстві.

Застосування цифрових зображень

Цифрові зображення використовуються у багатьох сферах, включаючи:

- ***Медицина:*** Медичні зображення, такі як рентгенівські знімки, МРТ та КТ, використовуються для діагностики та лікування.
- ***Наука та дослідження***: Використання зображень у мікроскопії, астрономії та інших наукових галузях.
- ***Розваги та медіа:*** Фотографії, відео, анімація та графіка.
- ***Безпека та спостереження:*** Відеоспостереження, розпізнавання обличь та інших біометричних характеристик.

Таким чином, цифрове зображення є універсальним інструментом, що знаходить застосування в багатьох галузях та технологіях, і його обробка є важливою складовою сучасних інформаційних систем.

1.2.2 Текстові зображення

Текстові зображення - це специфічний тип цифрових зображень, які містять текст. Вони зазвичай складаються з двох основних елементів: тексту та фону. Текст представлений пікселями одного кольору (наприклад, чорного), а фон - пікселями іншого кольору (наприклад, білого). Такі зображення використовуються для передачі інформації у вигляді символів, слів та речень.

Особливості текстових зображень

Текстові зображення мають кілька характерних особливостей, які відрізняють їх від інших типів зображень:

- ***Контрастність:*** Однією з основних характеристик текстових зображень є висока контрастність між текстом і фоном. Це забезпечує легкість читання та чіткість символів.
- ***Чіткі межі:*** Текстові зображення зазвичай мають чіткі межі між текстом і фоном. Це означає, що перехід між кольорами тексту та фону відбувається різко, без поступового змішування кольорів.
- ***Монохромність:*** Текстові зображення часто є монохромними, тобто складаються з двох кольорів – одного для тексту і одного для фону. Однак можуть існувати і кольорові текстові зображення, де текст і фон мають різні кольори.
- ***Висока роздільна здатність:*** Для забезпечення чіткості тексту і легкості його читання, текстові зображення часто мають високу роздільну здатність.

Виклики в обробці текстових зображень

Обробка текстових зображень ставить перед собою декілька специфічних завдань і викликів:

- ***Шум:*** Текстові зображення можуть містити шум, який погіршує читабельність тексту. Шум може з'явитися в результаті сканування, фотографування або збереження зображення у форматі з втратами.
- ***Розмиття:*** Нечіткість або розмиття тексту може виникнути через низьку якість зображення, рух під час зйомки або погану роздільну здатність.
- ***Деформація:*** Текст може бути деформований через неправильне сканування або фотографування під кутом.
- ***Артефакти стиснення:*** Використання форматів з втратами, таких як JPEG, може призвести до появи артефактів, що погіршують якість тексту.

Методи покращення текстових зображень

Для покращення якості текстових зображень використовуються різні методи обробки:

- ***Бінаризація:*** Перетворення зображення в двоколірний формат (чорно-білий) для виділення тексту на фоні. Одним із поширених методів є метод Отсу.
- ***Фільтрація шуму:*** Використання різних фільтрів, таких як медіанний фільтр або гаусівський фільтр, для зменшення шуму.
- ***Морфологічні операції:*** Застосування операцій математичної морфології, таких як ерозія та дилатація, для покращення контурів тексту та видалення артефактів.
- ***Корекція деформацій:*** Використання методів геометричної трансформації для виправлення викривлень і вирівнювання тексту.

Застосування текстових зображень

Текстові зображення використовуються в багатьох галузях, включаючи:

- ***Документне управління:*** Скановані копії документів, таких як контракти, листи та інші офіційні папери.
- ***Розпізнавання тексту (OCR):*** Перетворення текстових зображень у редагований текст за допомогою технологій оптичного розпізнавання символів.

- ***Презентації та навчальні матеріали:*** Використання текстових зображень у слайдах, діаграмах та інших навчальних матеріалах.
- ***Веб-сторінки та графіка:*** Інтернет-меми, цитати, рекламні банери, де текст є ключовим елементом зображення.

Таким чином, текстові зображення є важливою складовою цифрового контенту, яка потребує спеціалізованих методів обробки для забезпечення їх високої якості та читабельності.

РОЗДІЛ 2. МОРФОЛОГІЧНА ОБРОБКА ЗОБРАЖЕНЬ

Морфологічна обробка зображень є важливою областю в цифровій обробці зображень, яка базується на теорії множин та включає різні методи обробки зображень, спрямовані на виділення та аналіз структурних властивостей зображень. Ці методи дозволяють ефективно вирішувати задачі сегментації, видалення шумів, виділення контурів та інші завдання, що виникають при обробці зображень.

2.1 Основні поняття та визначення

2.1.1 Бінарні зображення

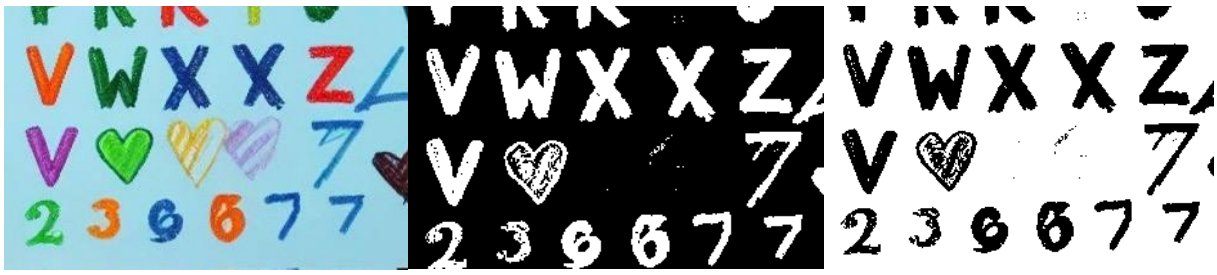
Бінарне зображення - це зображення, в якому кожен піксель може мати лише одне з двох значень: чорне або біле. Чорний піксель зазвичай позначається значенням 0, а білий - значенням 1. Такі зображення часто використовуються в морфологічній обробці для спрощення аналізу та обробки структурних елементів.

Бінарні зображення широко використовуються в різних областях, таких як медична діагностика, розпізнавання образів, сегментація зображень та багато інших. Вони можуть представляти різні типи об'єктів, такі як текст, форми, контури, елементи зображень тощо.

Інверсія бінарного зображення полягає у зміні чорних пікселів на білі і навпаки. Це може бути корисним у різних ситуаціях, таких як підвищення контрастності, покращення видимості об'єктів або просто для зміни естетичних характеристик зображення.

Вибір кольору для букв та фону в бінарних зображеннях залежить від конкретного застосування. У багатьох випадках букви представлені білими пікселями, а фон - чорними, тому що це сприяє читабельності тексту. Однак можна також використовувати зворотну схему, де букви представлені чорними пікселями, а фон - білими.

Приклад:



2.1.2 Структурний елемент

Структурний елемент (SE) - це невеликий набір пікселів, який використовується для сканування та аналізу зображення в процесі морфологічних операцій. Він може мати різну форму та розмір, залежно від конкретної задачі. Наприклад, для пошуку ліній на зображенні можна створити лінійний структурний елемент.

***Форма та розмір**:* Форма та розмір структурного елемента впливають на результат морфологічної операції. Наприклад, великі структурні елементи можуть призвести до більшої дилатації або ерозії, ніж малі структурні елементи.

***Орієнтація**:* Для деяких задач, таких як виявлення країв або ліній, може бути корисно використовувати структурні елементи певної орієнтації. Наприклад, вертикальний структурний елемент може бути використаний для виявлення вертикальних ліній на зображенні.

***Композитні структурні елементи*:** Можна комбінувати кілька структурних елементів разом, щоб створити композитний структурний елемент. Це може бути корисно для виконання складних морфологічних операцій.

Існують два типи структурних елементів: плоскі та неплоскі.

Плоский структурний елемент - це бінарне значення околиці, де істинні (1-значні) пікселі включені в морфологічну операцію, а хибні (0-значні) пікселі - ні. Ви можете використовувати плоскі структурні елементи з бінарними та відтінками сірого зображеннями.

Неплоский структурний елемент включає додаткове зміщення для кожного пікселя в околиці. Пікселі в околиці, які мають кінцеве реальне значення, використовуються в морфологічній операції.

Ось приклади структурних елементів:

- ***Квадратний структурний елемент:*** Це плоский структурний елемент, який має форму квадрата. Він може бути використаний для дилатації або ерозії зображення.
- ***Круглий структурний елемент*:** Це плоский структурний елемент, який має форму круга. Він може бути використаний для дилатації або ерозії зображення.
- ***Лінійний структурний елемент*:** Це плоский структурний елемент, який має форму лінії. Він може бути використаний для дилатації або ерозії зображення.
- ***Неплоский структурний елемент*:** Це структурний елемент, який має додаткове зміщення для кожного пікселя в околиці.

Приклад: Приклади побудови структурних елементів

```
% Квадратний структурний елемент
SE_square = strel('square', 3);
SE_square_big = imresize(SE_square.Neighborhood, 50, 'nearest');
figure, imshow(SE_square_big), title('Квадратний структурний елемент');

% Круглий структурний елемент
SE_disk = strel('disk', 5);
SE_disk_big = imresize(SE_disk.Neighborhood, 50, 'nearest');
figure, imshow(SE_disk_big), title('Круглий структурний елемент');

% Лінійний структурний елемент
SE_line = strel('line', 10, 45);
SE_line_big = imresize(SE_line.Neighborhood, 50, 'nearest');
figure, imshow(SE_line_big), title('Лінійний структурний елемент');
```

2.2 Основні морфологічні операції

2.2.1 Ерозія

Ерозія — це морфологічна операція, яка «стискає» або «зменшує» об'єкти на двійкових зображеннях. Вона визначається як логічне І (AND) між структурним елементом та зображенням. Ерозія застосовується для зменшення розмірів об'єктів, видалення дрібних шумів, а також для розділення об’єктів, які з’єднані.

Процес ерозії полягає в тому , що структурний елемент переміщується по всьому зображенню. У кожній позиції порівнюються всі пікселі структурного елемента з пікселями зображення. Якщо всі пікселі структурного елемента збігаються з пікселями об'єкта (білі пікселі) на зображенні, то центральний піксель результату встановлюється як білий. Якщо хоча б один піксель структурного елемента збігається з фоном (чорні пікселі), то центральний піксель результату встановлюється як чорний.

Математичне визначення ерозії:

$$A \ominus B = \{z | (B)z \cap \neg A \neq \emptyset\}$$

Де A — вихідне зображення, B — структурний елемент, (B)z — структурний елемент, зсунутий на позицію z, ¬A — доповнення множини A,∅ — порожня множина.

Це означає, що ерозія A по B включає всі ті позиції z, де структурний елемент B, зсунутий на z, перетинається з доповненням множини A.

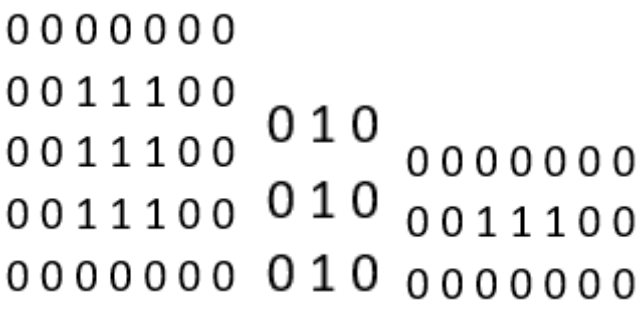
Властивості ерозії :

- ***Комутативність***: Ерозія є комутативною операцією, тобто порядок множин при застосуванні ерозії не впливає на результат. $A \ominus B = B \ominus A$.
- ***Асоціативність:*** Ерозія є асоціативною операцією, що дозволяє об'єднувати кілька структурних елементів послідовно. $(A \ominus B) \ominus C = A \ominus (B \ominus C)$.
- ***Монотонність:*** Якщо множина A включається в множину C, то ерозія множини A буде також включатися в ерозію множини C з тим самим структурним елементом B. якщо $A \subseteq C$, то $A \ominus B \subseteq C \ominus B$.

Припустимо, у нас є прямокутник на двійковому зображенні(Рис 1.2.1.1 ,зображення **а)**). Візьмемо структурний елемент із 3 пікселів вертикальної лінії.(Рис 1.2.2.1 ,зображення **б)**). Структурний елемент не повністю

перекривається об'єктом на зображенні, всі пікселі в області, яку охоплює структурний елемент, встановлюються як чорні. Таким чином, об'єкт стає зменшеним. (Рис 1.2.2.1 , зображення **в)**)

Рис 2.2.1.1



а) б) в)

Приклад:

У пакеті IPT операція ерозії виконується функцією imdilate.

```
IM2 = imerode(IM,SE)
```

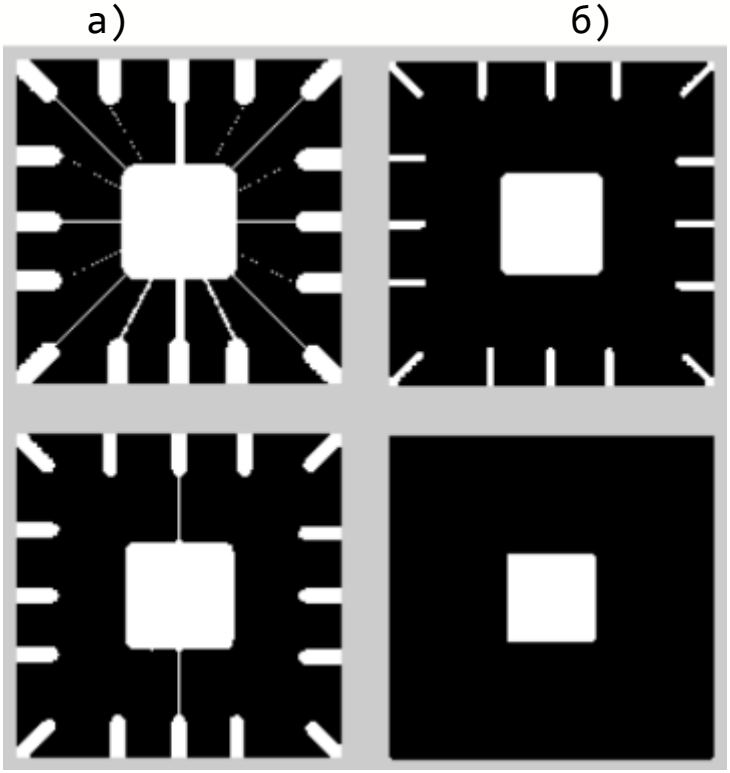
Функція приймає бінарне зображення IM та об'єкт або бінарну матрицю структуроутворюючого елемента SE. Повертається логічна матриця IM2.

На зображенні Рис 1.2.1.2 ,а) дано приклад двійкове зображення плати. Застосовуємо ерозію до цього зображення.

Використовуючи наступні команди, ми можемо видалити тонкі проводи, залишаючи інші структури недоторканими:

```
>> A = imread('wirebond-mask.tif');;
>> se1=strel('disk',10);
>> A1=imerode(A,se1);
>> se2=strel('disk',5);
>> A2 = imerode(A, B);
>> se3=strel('disk',20);
>> A3=imerode(A,se3);
>> figure, subplot(2,2,1); imshow(A);
>> subplot(2,2,2);imshow(A1);
>> subplot(2,2,3); imshow(A2);
>> subplot(2,2,4); imshow(A3);
>> A4=imerode(A,ones(15,15));
>> figure, imshow(A4);
```

Рис 1.2.1.2. Ілюстрація ерозії **а)** Ерозія з колом радіусу 10. **б)** Ерозія з колом радіусу 5. **в)** Ерозія з колом радіуса 20



1.2.2 Дилатації

Дилатації- це морфологічна операція, яка «нарощує» або «потовщує» об'єкти на двійкових зображеннях. Вона визначається як логічне АБО (OR) між структурним елементом та зображенням. Дилатації застосовується для збільшення розмірів об'єктів, заповнення прогалин та також може бути використане для з’єднання близько розташованих об’єктів.

Процес дилатації полягає в обнесенні центру структурного елемента по області зображення та відмітці тих положень центра, коли структурний елемент перекриває частину об'єкта на зображенні. Результат дилатації можна строго визначити математично в термінах операцій над множинами:

$$A \oplus B = \{z | (B)z \cap A \neq \emptyset\}$$

де A — вихідне зображення, $(B)z$ — структурний елемент, зсунутий на позицію z, а $\emptyset$ — порожня множина.

Це означає, що дилатація A по B — це всі ті позиції z, де структурний елемент B, зсунутий на z, має хоча б один спільний піксель з об'єктами на зображенні A.

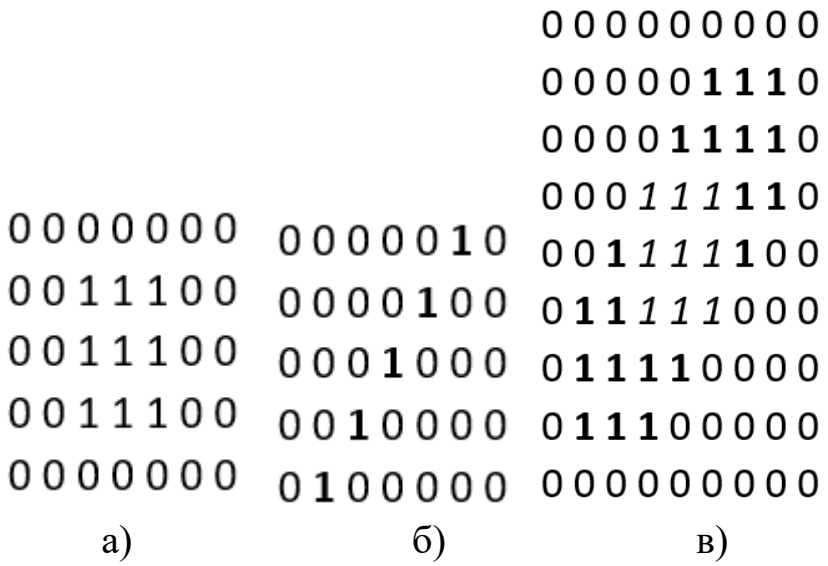
Властивості дилатації

- ***Комутативність:*** Дилатація є комутативною операцією, тобто A⊕B=B⊕A. Це означає, що порядок множин при дилатації не впливає на результат.
- ***Асоціативність:*** Дилатація є асоціативною операцією, тобто (A⊕B)⊕C=A⊕(B⊕C). Це дозволяє об'єднувати кілька структурних елементів послідовно.
- ***Монотонність:*** Якщо A⊆C, тоді A⊕B⊆C⊕B. Це означає, що дилатація зберігає порядок множин.

Припустимо, у нас є прямокутник на двійковому зображенні(Рис 2.2.2.1 ,зображення **а)**). Візьмемо структурний елемент із 5 пікселів по діагоналі.(Рис 2.2.2.1 ,зображення **б)**). Структурний елемент проходить по кожному пікселю зображення. Якщо будь-який з пікселів одиниці перекриває білий піксель на зображенні, то центральний піксель результату встановлюється в білий.

У результаті ми отримуємо збільшений об'єкт(Рис 2.2.2.1 , зображення **в)**)

Рис 2.2.2.1



Приклад:
У пакеті IPT операція дилатації реалізується функцією imdilate. Її основна форма виклику має вигляд

```
A2=imdilate(A,B),
```

де A і A2 - це двійкові зображення, а B - матриця з нулів і одиниць, яка утворює структуроутворюючий елемент.

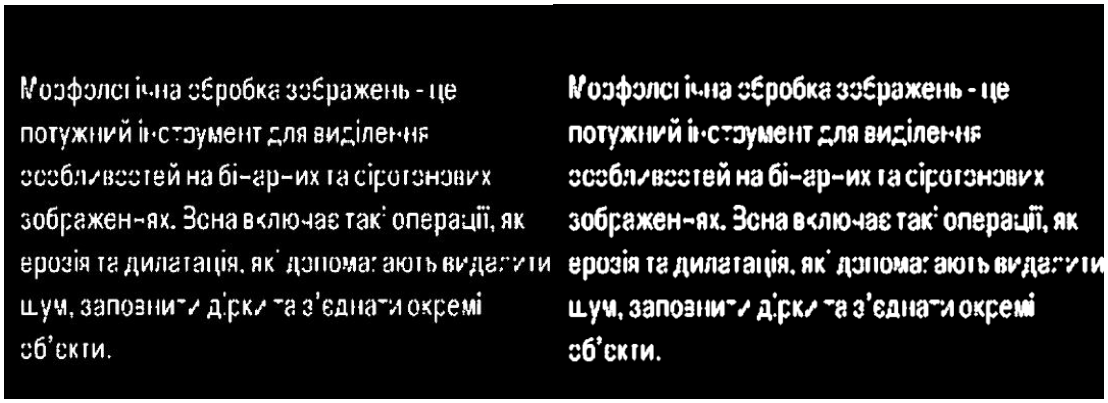
На зображенні **Рис 2.2.2.2 ,**а) дано приклад двійкового зображення, яке містить текст із розірваними літерами. Застосовуємо дилатацію до цього зображення із структуроутворюючим елементом

0 1 0
1 1 1
0 1 0

Наступна послідовність команд зчитує зображення з файлу, будує матрицю структуроутворюючого елемента, здійснює дилатацію та показує результат.

```
>> A = imread('broken-text.tif');
>> imshow(A);
>> B= [0 1 0; 1 1 1; 0 1 0]
>> A2 = imdilate(A, B);
>> figure, imshow(A2);
```

Рис. 2.2.2.2 . Простий приклад далатації. ***а)*** Вихідне зображення, що містить розірвані символи; ***б)*** Зображенняпісля застосування дилатації



а) б)

2.2.3 Відкриття та закриття

Відкриття - це морфологічна операція в обробці зображень, яка використовується для видалення дрібних об'єктів та шумів, зберігаючи форму більших об'єктів.

Процес відкриття полягає в послідовному застосуванні ерозії (зменшення) та дилатації (розширення) до зображення. Спочатку виконується ерозія, яка

видаляє маленькі об'єкти та шуми, а потім застосовується дилатація, що відновлює форму більших об'єктів.

Математично відкриття визначається як:

$$A \circ B = (A \ominus B) \oplus B$$

де A - вихідне зображення, B - структурний елемент, $\ominus$ - ерозія, а $\oplus$ - дилатація.

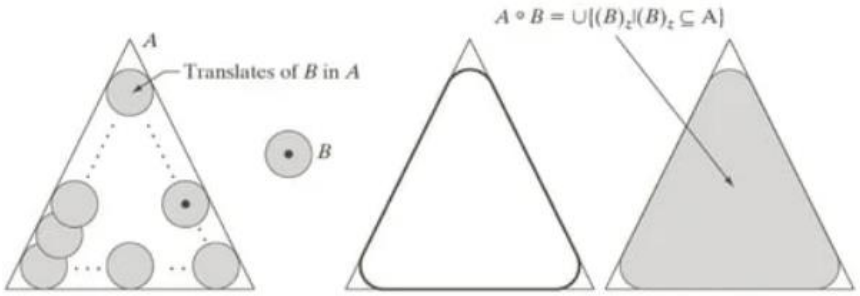
Цю операцію можна також задати еквівалентною формулою

$$A \circ B = U\{(B)_z | (B)_z \subseteq A\}$$

Де A - вихідне зображення, B - структурний елемент. (B)z - це зсув структурного елемента B, U - це оператор об’єднання, $\subseteq$ - це оператор підмножини, який вказує, що зсув структурного елемента повністю вписується в зображення.

На рисунку 2.2.3.1, а) показано множину A і структурний елемент B, який має форму круга. Рисунку б) показує той же великий трикутник A, але тепер зображено лише ті кільця, які повністю містяться в A, не перетинаючи його країв. Рисунок в) показує великий трикутник A з темною областю всередині, яка представляє об’єднання всіх перекладів B, що повністю вписуються в A. Ця область і є відкриттям A по B. Білі області показують ті частини A, де структурний елемент B не повністю поміщається, тому вони не належать до відкриття.

Рис. 2.2.3.1



а) б) в)

Закриття - це морфологічна операція в обробці зображень, яка використовується для заповнення дрібних прогалин та об'єднання близько розташованих об'єктів, зберігаючи форму.

Процес закриття полягає в послідовному застосуванні дилатації (розширення) та ерозії (зменшення) до зображення. Спочатку виконується дилатація, яка з'єднує близько розташовані об'єкти та заповнює прогалини, а потім застосовується ерозія, що відновлює форму та розділяє об'єкти.

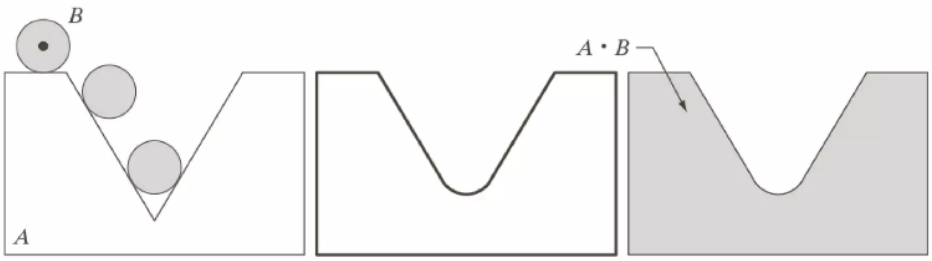
Математично закриття визначається як

$$A \cdot B = (A \oplus B) \ominus B$$

де A - вихідне зображення, B - структурний елемент, $\ominus$ - ерозія, а $\oplus$ - дилатація.

На Рис. 2.2.3.2, а) побудовано кілька зрушень множини B, які не перетинаються з A. Взявши доповнення до поєднання всіх таких зрушень, ми отримаємо область, затемнену на рис. в), яке є шуканим замикання.

Рис. 2.2.3.2



а) б) в)

Операції розмикання та замикання реалізовані в пакеті 1PT за допомогою функцій 1тореп та з.пс1озе. Обидві ці функції мають прості форми виклику

```
C = imopen(A, B)
C = imclose(A, B),
```

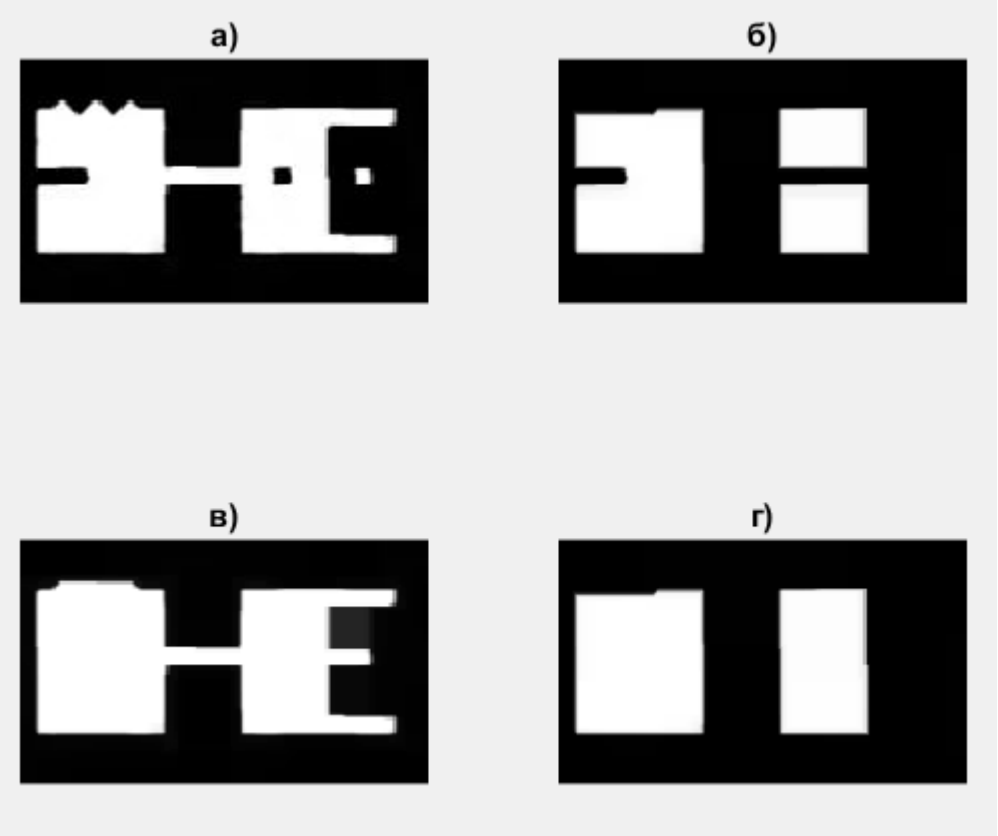
де A - це двійкове зображення, а B - матриця з 0 і 1, яка задає структуроутворюючий елемент. Strel-об'єкт SE можна використовувати замість аргументу B.

Приклад. Використання функцій imopen та imclose.

Зображення , яке має ряд характерних особливостей, що допомагають краще зрозуміти дію відкриття та замикання на такі об’єкти, як: гострий виступ, тонкі перемички, довгі поглиблення, ізольовані отвори, маленькі окремі об’єкти та зубчасті елементи кордонів. Наступна послідовність команд відкриває зображення за допомогою квадратного структурного елемента 20x20.

```
f = imread('shapes.tif');
se = strel('square', 20);

% Відкриття
fo = imopen(f, se);

% Закриття
fc = imclose(f, se);
fc1 = imclose(fo, se);

% Відображення зображень
figure;

% Початкове зображення
subplot(2, 2, 1);
imshow(f);
title('а)');

% Зображення після відкриття
subplot(2, 2, 2);
imshow(fo);
title('б)');

% Зображення після закриття
subplot(2, 2, 3);
imshow(fc);
title('в)');

% Зображення після обробки хакриттю
subplot(2, 2, 4);
imshow(fc1);
title('г)');
```

Рис. 2.2.3.3 .Ілюстрація розмикання та замикання, а) Вихідне зображення, б) Розмикання. в) Замикання, г) Замикання б)


а)
б)
в)
г)

РОЗДІЛ 3. РОЗРОБКА ПРОГРАМНОГО ПРОДУКТУ ТА ДЕМОНСТРАЦІЯ РЕЗУЛЬТАТІВ

3.1 Вибір інструментів розробки

3.1.1 Мова програмування MATLAB

Для розробки програмного продукту була обрана мова програмування MATLAB з урахуванням її потужних можливостей в області обробки сигналів та зображень. MATLAB є широко використовуваним інструментом у наукових дослідженнях, інженерії та інших галузях, завдяки своїй зручній синтаксичній структурі та багатому набору функцій.

Основні переваги використання MATLAB для цього завдання включають:

- **Потужність в обробці сигналів та зображень:** MATLAB має вбудований набір інструментів, що дозволяють виконувати різноманітні операції з обробки сигналів та зображень, включаючи фільтрацію, морфологічні операції, сегментацію та інші.
- **Багатий набір функцій та інструментів**: MATLAB постачається з широким спектром функцій та інструментів, таких як Image Processing Toolbox, що полегшують реалізацію складних алгоритмів обробки зображень.
- **Швидкість розробки та тестування:** Завдяки інтерактивному середовищу розробки, MATLAB дозволяє швидко розробляти, тестувати та налагоджувати програми, забезпечуючи при цьому високий рівень продуктивності.
- **Інтеграція з іншими мовами та інструментами:** MATLAB має можливості для інтеграції з іншими мовами програмування та зовнішніми інструментами, що розширює можливості розробки програмного продукту.

Обрання MATLAB для розробки програмного продукту дозволить ефективно виконати поставлені завдання з покращення текстових зображень методами математичної морфології, забезпечуючи при цьому високу якість та продуктивність роботи.

3.1.2 Image Processing Toolbox

Для реалізації алгоритмів обробки зображень було використано Image Processing Toolbox, що є одним з ключових інструментів у роботі з цифровими зображеннями в середовищі MATLAB. Цей інструментарій надає широкий спектр функцій для обробки, аналізу та візуалізації

зображень, що дозволяє легко виконувати різноманітні операції з обробки зображень.

Основні переваги використання Image Processing Toolbox включають:

- ***Морфологічна обробка:*** Завдяки вбудованим функціям для морфологічної обробки, таким як розширення, зменшення, відкриття, закриття тощо, можна легко виконувати операції над об'єктами на зображенні для поліпшення їх форми та властивостей.
- ***Фільтрація та видалення шуму:*** Image Processing Toolbox містить різноманітні фільтри для зменшення шуму на зображеннях, включаючи медіанний, гаусівський та Wiener фільтри, що дозволяє покращити якість зображення перед подальшою обробкою.
- ***Сегментація:*** Інструментарій має функції для сегментації зображень, що дозволяють виділити окремі об'єкти на зображенні або розділити зображення на різні регіони за рівнем яскравості, коліром тощо.
- ***Візуалізація та аналіз:*** Image Processing Toolbox має можливості для візуалізації та аналізу зображень, такі як побудова гістограм, відображення контурів, розрахунок статистичних показників тощо, що дозволяє зробити глибокий аналіз властивостей зображень.

Використання Image Processing Toolbox разом з MATLAB дозволяє ефективно виконувати різноманітні операції з обробки зображень для покращення їх якості та отримання необхідної інформації з них.

3.2 Архітектура розробленої програми

Реалізована програма для покращення текстових зображень використовує методи математичної морфології для обробки зображень. Програма написана мовою програмування MATLAB та складається з кількох функцій, які виконують послідовні етапи обробки зображень. У цьому розділі детально описано архітектуру програми, включаючи її функції та процеси обробки зображень.

Основні етапи обробки зображення

1. ***Завантаження зображення:*** Завантаження вихідного зображення для подальшої обробки.
2. ***Перетворення зображення в двійкове***: Перетворення кольорового або відтінкового зображення у двійкове зображення.
3. ***Інверсія зображення:*** Інверсія кольорів зображення для подальшої морфологічної обробки.

4. ***Морфологічні операції:*** Застосування операцій розширення (дилатації) та ерозії для покращення контурів тексту.
5. ***Заміна фону на оригінальний колір тексту***: Відновлення оригінальних кольорів тексту та фону.
6. ***Збереження та відображення результатів:*** Збереження оброблених зображень та їх відображення для візуального аналізу.

Опис функцій програми

1. Функція process_text_image

Функція ***process_text_image*** виконує основні етапи обробки текстових зображень. Вона завантажує зображення, перетворює його у двійкове, виконує інверсію та застосовує морфологічні операції для покращення якості тексту.

```
function processedImage = process_text_image(imagePath)
  % Завантаження зображення
  originalImage = imread(imagePath);

  % Перетворення зображення в двійкове
  if size(originalImage, 3) == 3 % Перевірка, чи зображення кольорове
      grayImage = rgb2gray(originalImage);
  else
      grayImage = originalImage;
  end
  binaryImage = imbinarize(grayImage);

  % Інверсія зображення, щоб текст був білим, а фон чорним
  invertedImage = ~binaryImage;

  % Виконання морфологічних операцій на інвертованому зображенні
  % Розширення зображення (збільшення об'єктів)
  dilatedImage = imdilate(invertedImage, strel('disk', 3)); % Застосовуємо
розширення зі структурним елементом диск радіусом 3.

  % Ерозія зображення (зменшення об'єктів)
  erodedImage = imerode(dilatedImage, strel('disk', 1)); % Застосовуємо
ерозію зі структурним елементом диск радіусом 1.

  % Заміна фону на оригінальний колір тексту
  resultImage = originalImage;
  for channel = 1:size(originalImage, 3)
      resultImage(:, :, channel) = ...
          originalImage(:, :, channel) .* uint8(erodedImage) + ...
          255 * uint8(binaryImage);
  end

  % Повернення обробленого зображення як вихідного аргументу
```

```
  processedImage = resultImage;
end
```

Опис етапів функції process_text_image

1. **Завантаження зображення:**
   - Вхідне зображення завантажується за допомогою функції imread. Це може бути зображення у форматі JPEG, PNG, BMP тощо.
   - Вхідний параметр imagePath містить шлях до файлу зображення.

2. **Перетворення зображення в двійкове:**
   - Якщо зображення кольорове, воно перетворюється у відтінкове (грейскейл) за допомогою функції rgb2gray.
   - Якщо зображення вже відтінкове, воно використовується без змін.
   - Потім відтінкове зображення бінаризується за допомогою функції ***imbinarize***, яка перетворює його у двійкове (чорно-біле) на основі порогового значення.

3. **Інверсія зображення:**
   - Інверсія кольорів виконується для зручності подальших морфологічних операцій: текст стає білим, а фон – чорним.

4. **Морфологічні операції:**
   - ***Дилатація (розширення):***

•Застосовується операція розширення (***imdilate***) зі структурним елементом у вигляді диска радіусом 3.

•Це збільшує білі об'єкти (текст), що допомагає з'єднати розірвані частини тексту та видалити дрібні чорні шумові артефакти.

   - ***Ерозія (зменшення):***

•Застосовується операція ерозії (***imerode***) зі структурним елементом у вигляді диска радіусом 1.

•Це зменшує білі об'єкти, що допомагає очистити текст від шуму та зробити його більш чітким.

5. **Заміна фону на оригінальний колір тексту:**
   - Відновлення оригінальних кольорів тексту та фону здійснюється шляхом поєднання обробленого двійкового зображення з оригінальним зображенням.

- Це забезпечує збереження кольорової інформації та покращення візуальної якості тексту.

6. Повернення обробленого зображення:

- Функція повертає оброблене зображення як вихідний аргумент.

2. Функція removeBackgroundAndEnhanceText

Функція removeBackgroundAndEnhanceText виконує додаткову обробку зображення для видалення фону та покращення чіткості тексту. Вона дозволяє налаштовувати товщину ліній тексту за допомогою морфологічних операцій.

```
function outputImage = removeBackgroundAndEnhanceText(imagePath,
targetThickness)
    % Завантаження зображення
    inputImage = imread(imagePath);

    % Перетворення зображення в двійкове
    if size(inputImage, 3) == 3 % Перевірка, чи зображення кольорове
        grayImage = rgb2gray(inputImage);
    else
        grayImage = inputImage;
    end
    binaryImage = imbinarize(grayImage);

    % Інверсія зображення, щоб текст був білим, а фон чорним
    invertedImage = ~binaryImage;

    % Створення структурних елементів для розширення та ерозії
    seDilation = strel('disk', ceil(targetThickness / 3));
    seErosion = strel('disk', floor(targetThickness / 3));

    % Виконання морфологічних операцій на інвертованому зображенні
    dilatedImage = imdilate(invertedImage, seDilation);
    erodedImage = imerode(dilatedImage, seErosion);

    % Інверсія зображення назад, щоб текст був чорним
    outputImage = ~erodedImage;
end
```

Опис етапів функції removeBackgroundAndEnhanceText

1. ***Завантаження зображення***:

- Завантаження вхідного зображення з диска за допомогою функції ***imread.***

2. Перетворення в двійкове зображення:

- Перетворення кольорового зображення у відтінкове за допомогою ***rgb2gray*** (якщо необхідно).

•Перетворення відтінкового зображення у двійкове за допомогою ***imbinarize***.

3. Інверсія зображення:

•Інверсія двійкового зображення для подальшої морфологічної обробки.

4. Створення структурних елементів:

•Створення структурних елементів для дилатації та ерозії на основі заданої товщини тексту (***targetThickness***).

5. Морфологічні операції:

•Дилатація: Збільшення об'єктів за допомогою структурного елемент

3.3 Демонстрація результатів роботи

Ця програма для обробки текстових зображень виконує ряд важливих операцій, що покращують якість тексту та роблять його більш чітким та привабливим. Варто зазначити, що разом із цими перевагами в програмі присутні й недоліки, які впливають на кінцевий результат.

Переваги програми:

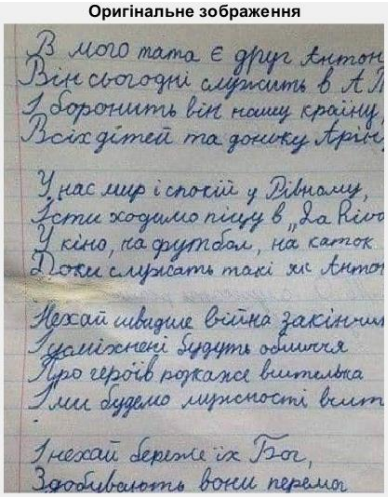
- **Покращення чіткості тексту**: Морфологічні операції дозволяють видалити непотрібний шум та з'єднати розірвані частини тексту. Це призводить до підвищення рівня чіткості тексту та полегшує його подальше сприйняття та аналіз.
- **Видалення фону**: Програма ефективно видаляє фонові об'єкти, залишаючи лише текст. Це створює зручні умови для подальшого використання обробленого тексту у різних контекстах, де необхідне виключно його використання.
- **Збереження кольорів тексту**: Кольорова інформація тексту зберігається, що дозволяє зберегти візуальну привабливість зображення та підкреслити його естетичний аспект.

Недоліки програми:

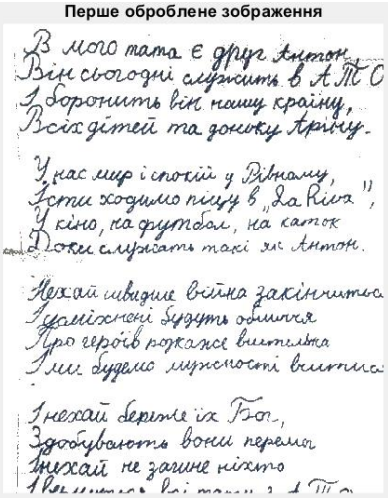
- **Зміна пікселів тексту на чорне при збільшенні товщини ліній**: При збільшенні товщини ліній тексту можуть втрачати деяку деталізацію, а пікселі тексту можуть бути змінені на чорне. Це може призводити до втрати якості та чіткості тексту, особливо на віддалених ділянках зображення.
- **Обмеженість роботи з певними типами зображень**: Програма може не завжди ефективно обробляти складні текстові зображення, де присутні різні шрифти, кольори та стилі. Це обмежує універсальність програми для використання в різних сценаріях.

Хоча наша програма має свої недоліки, переваги використання її переважають, особливо коли враховується автоматизація обробки великої кількості текстових зображень.

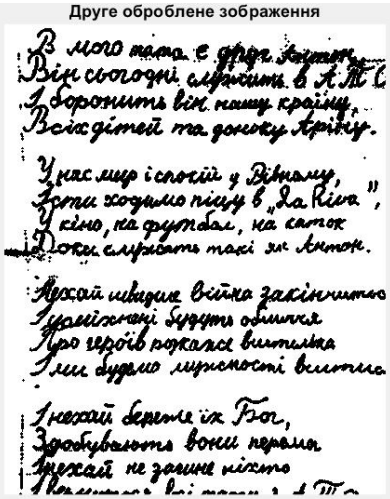
Оригінальне зображення



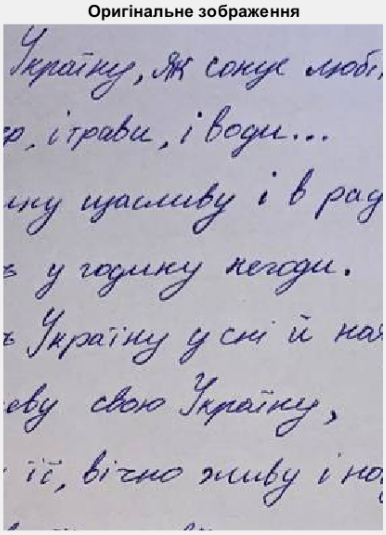
Перше оброблене зображення



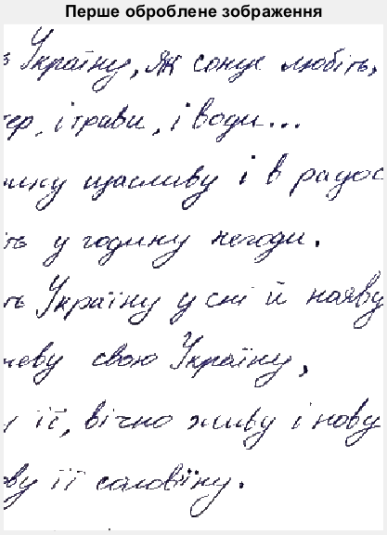
Друге оброблене зображення



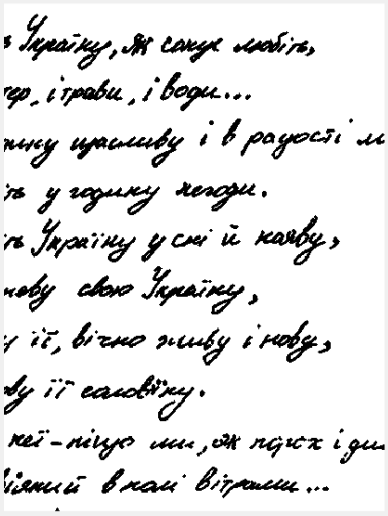
Оригінальне зображення

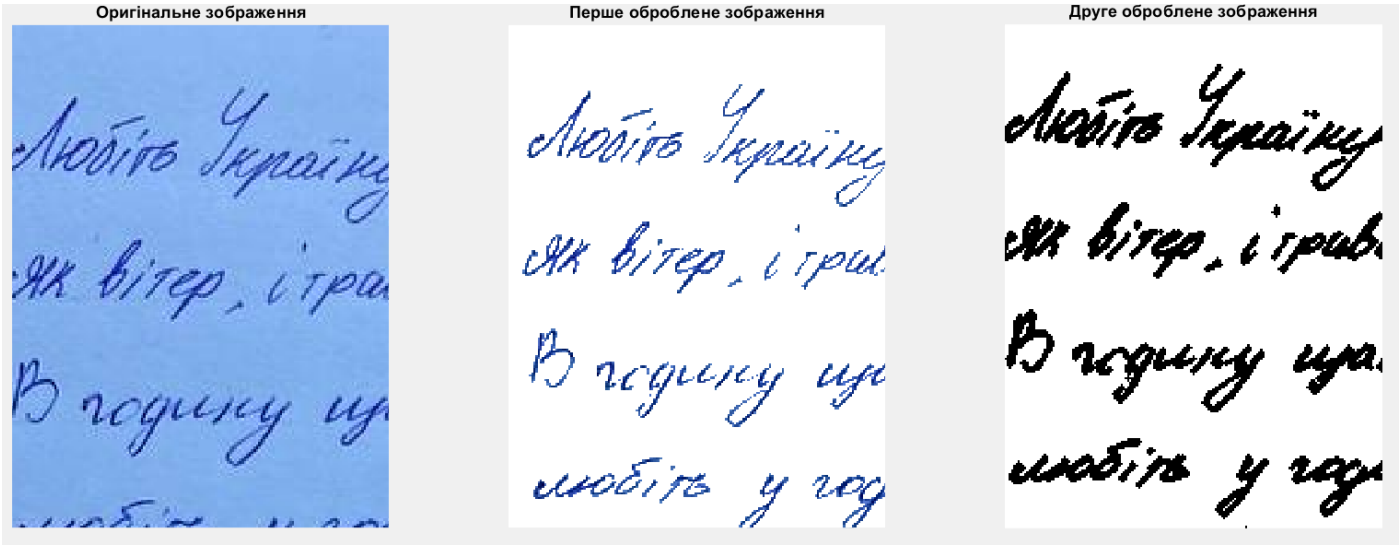


Перше оброблене зображення



Друге оброблене зображення





У ході тестування було виявлено додатковий недолік програми: вона може неефективно видаляти фон, якщо він дуже нерівномірний або має значні варіації в яскравості чи кольорі. У таких випадках програма може не виявити всі нерівності фону або не здійснити його повноцінного висвітлення, що може призвести до збереження деяких фрагментів фону на обробленому зображенні. Це може вплинути на остаточну якість обробленого тексту та його сприйняття.

Щоб вирішити цю проблему, треба спочатку виявити на зображенні найтемніше місце фону. У нашому випадку це правий нижній діагональний кут зображення.

Це можна зробити завдяки цій функції:

```
img = imread('photo_3.jpg'); % Зчитуємо зображення
img = im2double(img); % Конвертуємо зображення в формат double

k = 1; % Визначаємо коефіцієнт для діагоналі
```

```
intensity = 1; % Визначаємо початкову інтенсивність висвітлення

% Визначаємо правий нижній кут зображення
for i = 1:size(img,1)
    for j = 1:size(img,2)
        if j > size(img,2) - i + k
            % Застосовуємо градієнт висвітлення
            img(i,j,:) = imadjust(img(i,j,:), [0; intensity - 0.4 * (j -
(size(img,2) - i + k)) / size(img,2)], []);
        end
    end
end

imshow(img); % Відображаємо зображення
imwrite(img, 'resD.jpg'); % Зберігаємо зображення
```

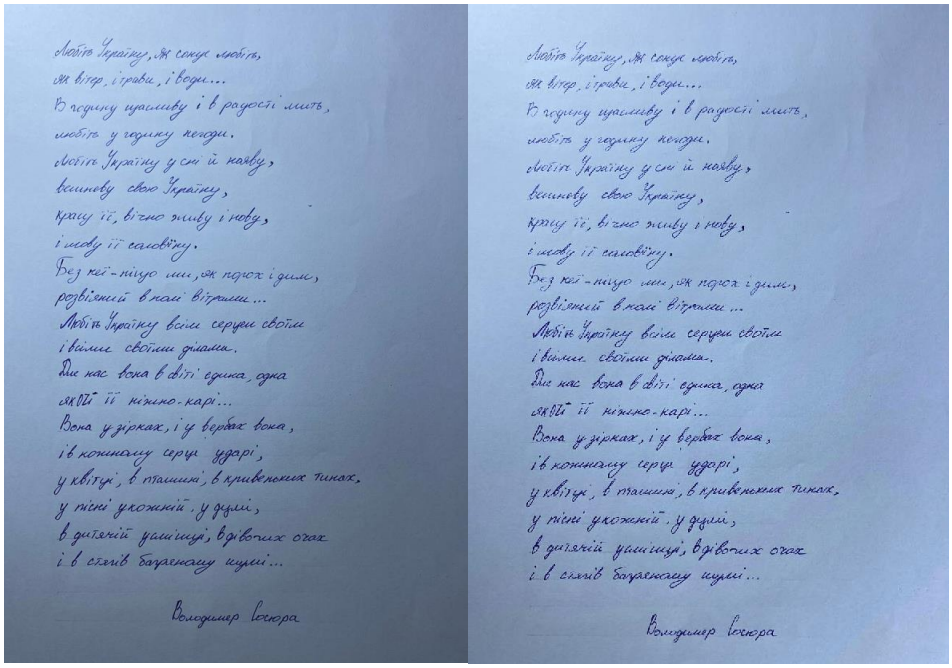
Отримуємо результат, як на фото (Рис. 3.3.2 б), а) – це початковий результат. Важливо підібрати відповідні параметри в цьому рядку, щоб вирівняти фон: **intensity = 1;.**

Бачимо, що фон майже рівномірний. Тепер спробуємо знову застосувати до цього фото морфологічні методи для покращення результату (Рис. 3.3.3).

Важливо враховувати наступні аспекти:

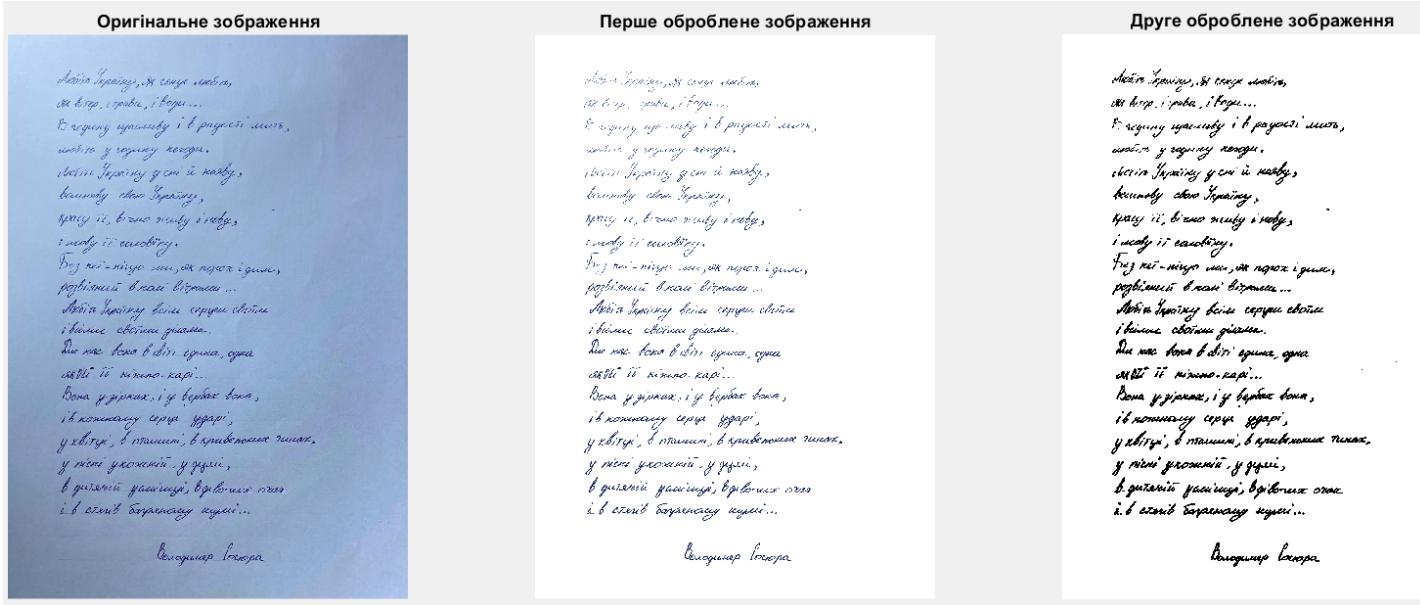
1. **Налаштування параметрів**: Підбір оптимальних значень параметрів, таких як інтенсивність та коефіцієнт для діагоналі, щоб забезпечити максимально рівномірне освітлення фону.
2. **Морфологічна обробка:** Після вирівнювання фону, застосування морфологічних методів для подальшого покращення якості зображення.

Рис 3.3.2



а) б)

Рис 3.3.3



ВИСНОВКИ

В роботі було виконано аналіз існуючих методів математичної морфології, які підходять для покращення якості цифрових текстових зображень, а також розроблено алгоритм, спрямований на виправлення дефектів, підвищення чіткості та зручності використання таких зображень. Показано, що методи математичної морфології є ефективними для розв'язання поставлених в роботі задач. На основі проведеного аналізу були розроблені алгоритми, які включають перетворення зображення в двійкове, інверсію зображення, морфологічні операції (розширення та ерозія) та збереження оригінальних кольорів тексту.

Створена програма автоматизує процес обробки текстових зображень, що дозволяє значно полегшити роботу з великою кількістю таких зображень. У процесі тестування програми було виявлено кілька недоліків, серед яких зміна пікселів тексту на чорне при збільшенні товщини ліній та недостатня ефективність обробки зображень з нерівномірним фоном. Для вирішення цих проблем було запропоновано рішення, зокрема використання додаткових алгоритмів для вирівнювання фону.

Практична цінність роботи полягає у можливості застосування розроблених алгоритмів та програмних рішень у різних галузях, де використовується обробка текстових зображень, таких як системи розпізнавання тексту, автоматизована обробка документів та створення візуальних матеріалів. Результати роботи сприяють підвищенню ефективності та якості обробки текстових зображень, що є важливим для сучасного цифрового світу.

Подальші дослідження можуть бути спрямовані на вдосконалення розроблених алгоритмів та розширення їх функціональності, зокрема інтеграцію методів машинного навчання для покращення адаптивності алгоритмів до різних типів текстових зображень та фонових умов. Можливим напрямком досліджень є розробка універсальних рішень для обробки текстових зображень з різноманітними шрифтами, кольорами та стилями.

Загалом, результати цієї дипломної роботи підтверджують ефективність використання методів математичної морфології для покращення якості текстових зображень і відкривають нові перспективи для подальших досліджень та практичного застосування у цій галузі.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Р. Гонсалес, Р. Вудс «Цифрова обробка зображень. М: Техносфера,2005».

2.Morphological image processing [Електронний ресурс]. https://www.slideshare.net/pareshkamble/morphological-image-processing-246351698

3. Milan Sonka, Vaclav Hlavac, Roger Boyle «Image Processing, Analysis, and Machine Vision»

4. Р. Гонсалес, Р. Вудс С. Эддинс Цифровая обработка изображений в среде Matlab. М.: Техносфера, 2006.

5.В.Т. Фисенко, Т.Ю. Фисенко «Компьютерная обработка и распознавание изображений: учеб. пособие. - СПб: СПбГУ ИТМО, 2008»