Харківський національний університет імені В.Н. Каразіна
(повне найменування вищого навчального закладу)

Фізико-енергетичний факультет
(повне найменування інституту, назва факультету (відділення))

Інформаційні технології в фізико-енергетичних системах
(повна назва кафедри (предметної, циклової комісії))

ПОЯСНЮВАЛЬНА ЗАПИСКА

до дипломного проекту (роботи)

магістр
(освітньо-кваліфікаційний рівень)

на тему:

«Морфологічна обробка і структурування елементів зображення»

Виконав: студент _6_курсу,

групи__НФ-63____

напряму підготовки (спеціальності)

6.040204 прикладна фізика
(шифр і назва напряму підготовки, спеціальності)

Боженов М.О.
(прізвище та ініціали)

Керівник к.т.н., доцент каф. ІТФЕС, Лісін Д.О.
(прізвище та ініціали)

Рецензент_____2017 року
(прізвище та ініціали)

РЕФЕРАТ

Магістерська робота виконана студентом Боженовим Максимом Олеговичем на тему «Морфологічна обробка і структурування елементів зображення ».

Об'єктом дослідження магістерської роботи є алгоритми обробки зображень. Основним результатом роботи є розроблений програмний продукт .

Пояснювальна записка містить 50 аркушів основного тексту, 36 рисунків, 21 джерело використаної літератури.

Ключові слова: Обробка зображень, поліпшення зображень, морфологія зображення, діляція, ерозія.

РЕФЕРАТ

Магистерская работа выполнена студентом Боженов Максимом Олеговичем на тему «Морфологическая обработка и структурирование элементов изображения».

Объектом исследования магистерской работы является алгоритмы обработки изображений. Основным результатом работы является разработанный программный продукт.

Пояснительная записка содержит 50 листов основного текста, 36 рисунков, 21 источник литературы.

Ключевые слова: Обработка изображений, улучшение изображений, морфология изображений, диляция, эрозия.

ABSTRACT

The masters work was carried out by Maxim Bozhenov, student of the subject "Morphological processing and structuring of image elements".

The object of research of the masters work is the image processing algorithms. The main result of the work is the developed software product.

The explanatory note contains 50 sheet of the main text, 36 figures, 21 sources of literature.

Keywords: Image processing, image enchancement, image morphology, dilation, erosion, hit or miss.

ЗМІСТ

ВСТУП……..6
1.ТЕОРЕТИЧНІ ДОСЛІДЖЕННЯ.....……...8
1.1 Загальна постановка задачі цифрової обробки зображень...…….8
1.2 Походження цифрової обробки зображень...……....10
1.3 Компоненти системи цифрової обробки зображень.....……...13
2.МОРФОЛОГІЧНА ОБРОБКА ЗОБРАЖЕНЬ.....…...18
2.1 Попередні відомості.....…….....18
2.2 Ерозія і дилатація….....……22
2.3 Відмикання і замикання.....…….....25
2.4 Перетворення Hit or Miss.....……27
2.5 Деякі базові морфологічні алгоритми….…….……28
2.5.1 Виділення границі...…….……28
2.5.2 Заповнення дирок.....……..29
2.5.3 Виділення зв’язних компонент……...30
2.5.4 Побудова опуклої оболонки.....…….……31
3. КОМПЮТЕРНА СИСТЕМА ДЛЯ МОРФОЛОГІЧНОЇ ОБРОБКИ
ТА СТРУКТУРУВАННЯ ЕЛЕМЕНТІВ ЗОБРАЖЕННЯ .....……...33
4. ВИСНОВКИ……...48
5. СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ……49
6.ДОДАТОК……...51

ВСТУП

Задача обробки цифрових зображень у теперішній час є надзвичайно актуальною тому, що цифрова фотографія, також як і програмне забезпечення для її обробки використовуються усюди, від аматорського до професійного рівня. Програмне забезпечення, що виконує постобробку цифрових зображень стає все більш складним, але також і більш дорогим. Роздільна здатність сучасних пристроїв для фото- та відеозйомки стає все більшою, що призводить до необхідності використовувати найбільш швидкі та економні до ресурсів алгоритми. Більшість програмних продуктів, що використовуються для цифрової обробки зображень є платною і більш того, дуже дорогою.

Методи обробки цифрових зображень широко використовуються також і у наукових дослідженнях. Фільтрація зображень, виявлення меж зображень, виділення контурів об'єктів на зображенні, створення тривимірної моделі із серії двовимірних цифрових фотографій – усе це підтримується більшостю із сучасних професійних програмних продуктів, що орієнтовані на обробку цифрових зображень.

Підсумовуючи усе вищевказане, можна стверджувати. що задача створення бескоштовного програмного продукту, що орієнтований на використання у наукових дослідженнях, що потребують цифрової обробки зображень є актуальною та перспективною у різих областях науки.

Метою даної роботи є розробка і програмна реалізація системи. що призначена для різноманітної цифрової обробки зображень і має вигляд плагіну для пакету прикладних програм MATLAB. Ця система у разі необхідності може бути підключена до абиякого програмного комплексу у MATLAB як модуль, але також вона може використовуватися і автономно.

Для досягнення цієї мети були поставлені та виконані наступні завдання:

- Постановка завдання;

- Проведення аналізу та вибір методів цифрової обробки зображень;
- Розробка алгоритму програми;
- Розробка інтерфейсу програми;
- Реалізація програмного продукту «Image Morphology»;
- Тестування програмного продукту.

Об’єктом даної роботи є алгоритми цифрової обробки зображень, які у подальшому використовуються для створення системи «Morphology».

Предмет роботи – постобробка цифрових фотографій, що представлені у вигляді двох- або тривимірних масивів.

1. ТЕОРЕТИЧНІ ДОСЛІДЖЕННЯ.

1.1 ЗАГАЛЬНА ПОСТАНОВКА ЗАДАЧІ ЦИФРОВОЇ ОБРОБКИ ЗОБРАЖЕНЬ

Зображення може бути визначено як двохмістна функція $f(x,y)$, де x та y - просторові координати, а значення f для абиякої пари координат називається інтенсивністю рівня сірого зображення у цій точці. Коли значення x, y та значення f є усі фінітними та дискретними, ми називаємо таке зображення цифровим зображенням. Задача цифрової обробки фотографій є задачею обробки цифрових зображень засобами (цифрових) комп'ютерів.

Цифрове зображення складається із фінітної кількості елементів, кожен з яких має своє розташування та значення. Ці елементи називаються елементами зображення або пікселами.

Бачення є найбільш продвинутим почуттям людини, тому немає нічого дивного у тому, що зображення займають найважливіше місце серед усього, що сприймає людина. Проте, на відміну від людей, які обмежені візуальною частиною електромагнітного спектру, машини для створення та обробки зображень охоплюють практично весь спектр електромагнітних волн, починаючи від гамма-10 радіохвиль.

Вони можуть працювати із зображеннями, отриманими із джерел, що люди не звикли асоціювати із зображеннями. Ці джерела включають в себе ультразвук, електронну мікроскопію, і комп'ютерні зображення. Таким чином, цифрова обробка зображення включає в себе широку і різноманітну сферу застосування.

Не існує загальної згоди щодо того, де проходить кордон між областю обробки зображень та іншими суміжними областями, такими як аналіз зображень та комп'ютерне бачення. Іноді ці області розрізняють шляхом визначення обробки зображень як дисципліни, в

якій як вхід, так і вихід процесу є зображення. Дехто вважає, що це є лімітуючим і дещо штучним обмеженням. Наприклад, відповідно до цього визначення, навіть тривіальна задача обчислення середньої інтенсивності зображення (яка повертає одне число) не буде вважатися операцією з обробки зображень. З іншого боку, є такі поля, як комп'ютерне бачення, кінцевою метою яких є використання комп'ютерів для емуляції людського зору, включаючи обучення і можливість зробити висновки і певні дії, засновані на візуальних даних. Сама ця область є філією штучного інтелекту (AI), метою якої є моделювання людського інтелекту. Область AI зараз є на самих ранніх стадіях дитинства в плані розвитку, і прогрес проходить набагато повільніше, ніж передбачалося спочатку. Область аналізу зображень знаходиться між обробкою зображень і комп'ютерним зором.

Немає чіткої межі в континуумі між обробкою зображень з одного кінця і комп'ютерного зору з іншого. Проте, корисно розглянути три типи автоматизованих процесів в цьому континуумі: низько-, середньо- і високорівневі процеси. Процеси низького рівня включають примітивні операції, такі як попередня обробка зображення для зменшення шуму, підвищення контрастності і різкості зображення. Процес низького рівня характеризується тим фактом, що обидва його входи і виходи - зображення. Обробка середнього рівня на зображеннях включає в себе такі завдання, як сегментація (розподіл зображення на області або об'єкти), опис цих об'єктів, щоб звести їх до форми, придатної для комп'ютерної обробки і класифікації (розпізнавання) окремих об'єктів. Середньорівньовий процес характеризується тим, що його входи як правило, представляють собою зображення, але його виходи - атрибути, витягнуті з цих зображень (наприклад край, контури об’єктів, або ідентифікація окремих об'єктів). І, нарешті, обробка більш високого рівня включає в себе «почуття» рішень ансамблю розпізнаних об'єктів, як і в аналізі зображень, а також, на самому дальньому кінці

континууму, виконання когнітивних функцій, зазвичай пов'язаних із зором.

На підставі попередніх коментарів, можна побачити, що логічне місце перекриття між обробкою зображень і аналізом зображень є область розпізнавання окремих ділянок або об'єктів в зображенні. Таким чином, те, що ми називаємо цифровою обробкою зображень включає в себе процеси, входи і виходи яких – зображення і крім того, охоплює способи вилучення атрибутів із зображень, включаючи розпізнавання окремих об'єктів. В якості ілюстрації для уточнення цій понять, розглянемо область автоматизованого аналізу тексту. Процеси отримання зображення в області, що містить текст, попередню обробку, цього зображення, витяг (сегментування) окремих символів, що описують символи в формі, придатній для комп'ютерної обробки і розпізнавання цих окремих символів знаходиться в рамках того, що ми називаємо цифровою обробкою зображень. Осмислення змісту сторінки можна розглядати як в області аналізу зображень і навіть комп'ютерного зору, в залежності від рівня складності, що мається на увазі твердженням «сенс». Цифрова обробка зображень, як ми визначили її, успішно використовуються в широкому спектрі областей виняткової соціальної та економічної цінності.

1.2. ПОХОДЖЕННЯ ЦИФРОВОЇ ОБРОБКИ ЗОБРАЖЕНЬ

Одним з перших застосувань цифрових зображень булозастосування в газетної індустрії, коли зображення були вперше послані за допомогою підводного кабеля між Лондоном і Нью-Йорком. Введення кабельної системи передачі зображення Bartlane на початку 1920-х років зменшило час, необхідний для транспортування зображення через Атлантику з більш ніж за тиждень до менш ніж за три години.

Спеціалізоване друкарське обладнання кодувало зображення для передачі по кабелю, а потім реконструйовувало їх на приймальному кінці. Малюнок 1.1 був переданий таким чином, і відтворений на принтері телеграфного, забезпечений шрифтами імітації напівтонов.

Деякі з первинних проблем в поліпшенні якості зображення цих ранніх цифрових зображень були пов'язані з вибором процедур друку, і розподілом рівнів інтесивності. Метод печаті, використовуваний при отримані Рис. 1.1 був залишений у кінці 1921 року у користь методики, заснованої на фотографічній репродукції, виготовленій зі стрічок перфорованих на телеграфному приймальному терміналі.



Рис. 1.1

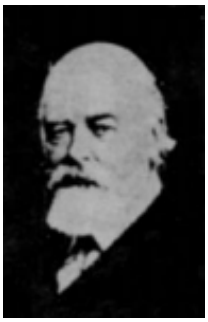


Рис. 1.2

Малюнок 1.2 показує зображення, отримане за допомогою цього методу. Покращення над рис. 1.1 очевидні, як в тональній якості, так і в роздільній здатності.

Ранні системи Bartlane були здатні кодувати зображення в п'яти різних рівнях сірого. Ця можливість була збільшена до 15 рівнів в 1929 р. Малюнок 1.3 є типовим для того типу зображень, які можуть бути отримані з використанням 15-тонального обладнання. Протягом цього періоду, впровадження системи для розробки плівки пластини за допомогою світлових променів, які були модульовані по кодованому зображенню стрічки, значно поліпшили процес відтворення. Хоча ці приклади включають цифрові зображення, вони не розглядаються як результати цифрової обробки зображень в контексті нашого визначення, тому що комп'ютери не були залучені в їх створенні. Таким чином, історія цифрової обробки зображень тісно пов'язана з розвитком

цифрових обчислювальних машин. Цифрові зображення вимагають так багато пам'яті і обчислювальної потужності, що прогрес в області цифрової обробки зображень, який залежить від розвитку цифрових обчислювальних машин і допоміжних технологій, які включають в себе зберігання, відображення і передачі даних.



Рис. 1.3

Ідея комп'ютера сходить до винаходу абаки в Азії якнайменш, більше 5000 років тому. Значно пізніше, за останні два століття, пройшли зміни, які призвели до появи того, що зветься сьогодні «комп'ютер». Проте, основою для того, що ми називаємо сучасним цифровим комп'ютером сходить до 1940-х років із введенням Джоном фон Нейманом двох ключових понять: (1) пристрій для зберігання збереженої програми і даних, і (2) умовного розгалуження. Ці дві ідеї є основою центрального процесора (CPU), який знаходиться в самому центрі комп'ютерів сьогодні. Починаючи з фон Неймана, виник ряд ключових досягнень, які зробили комп'ютери досить потужним, щоб бути використаними для обробки цифрових зображень. Якщо коротко, ці досягнення можуть бути підсумовані наступним чином:

(1) винахід транзистора в Bell Laboratories в 1948;

(2) розвиток в 1950-і і 1960-і роки з мов програмування високого рівня COBOL (Common Business-Oriented Language) і FORTRAN (FORmula TRANslator);

(3) винахід інтегральної схеми (IC) в Texas Instruments в 1958 році;

(4) розробка операційних систем на початку 1960-х років;

(5) розвиток мікропроцесора (один чіп, що складається з центрального процесору, пам'яті і вхідних і вихідних елементів управління) від Intel в на початку 1970-х;

(6) введення IBM персонального комп'ютера в 1981 році; і

(7) прогресуюча мініатюризація компонентів, починаючи з високого ступеню інтеграції (LI) в кінці 1970-х років, крізь дуже великі масштаби інтеграції (VLSI) в 1980-х роках, по сьогоденне використання ультра великої міри інтеграції (HBIC).

Одночасно з цими успіхами були проведені розробки в області систем відображення та масового зберігання, обидва з яких є основними вимогами для обробки цифрових зображень.

1.3 КОМПОНЕНТИ СИСТЕМИ ЦИФРОВОЇ ОБРОБКИ ЗОБРАЖЕНЬ

Зовсім недавно, в середині 1980-х років, чисельні моделі систем обробки зображень, що продавалися в усьому світі були дуже серйозними периферійниим пристроями, що був підключені до суттєво однакових хост-комп'ютерів. В кінці 1980-х і на початку 1990-х років, ринок змістився до апаратних засобів обробки зображень у вигляді окремих плат, призначених для забезпечення сумісності зі стандартними шинами і які могли вписуватися до інженерних робочих станцій і персональних комп'ютерів. На додаток до зниження витрат, зрушення цього ринку послужило в якості каталізатора для значного числа нових компаній, що спеціалізуються на розробці програмного забезпечення, написаного спеціально для обробки зображень.

Хоча великомасштабні системи обробки зображень як і раніше продаються для масивних додатків візуалізації, таких як обробка супутникових зображень, тенденція триває в напрямку мініатюризації і змішання загального призначення невеликих комп'ютерів з

спеціалізованими апаратними засобами обробки зображень. Малюнок 1.4 показує основні компоненти, що включає типова система загального призначення, яка використовується для функції обробки цифрових зображень.

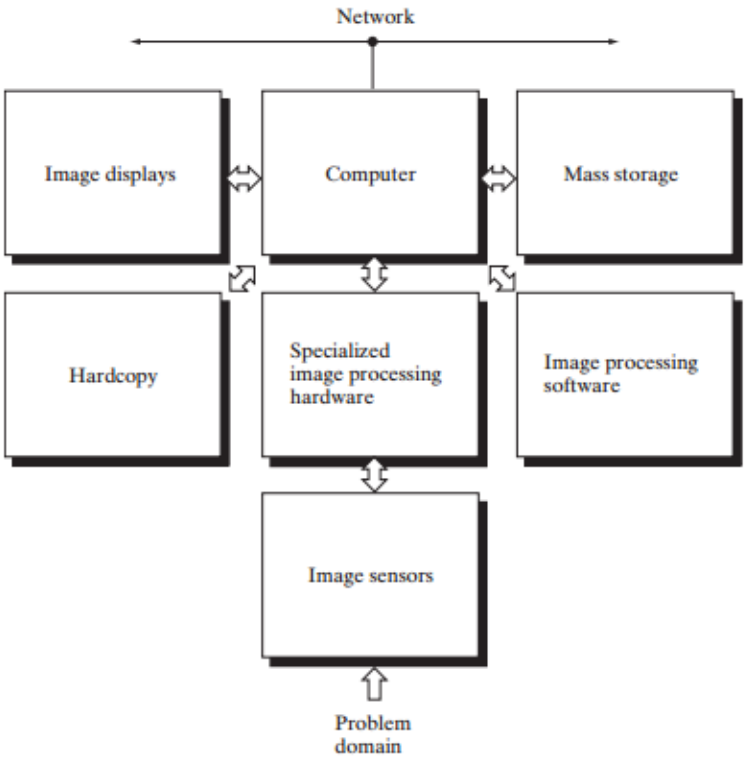


Рис. 1.4

Для зчитування зображення необхідні два елементи. Першим є фізичний пристрій, який є чутливим до енергії, випромінюваної об'єктом, який ми хочемо зобразити. Другий називається цифровим перетворювачем, являє собою пристрій для перетворення вихіда пристрою фізичного сприйняття в цифрову форму. Наприклад, в цифровій відеокамері, датчики формують електричний вихідний сигнал, пропорційний інтенсивності світла. Цифровий перетворювач перетворює ці виходи в цифрові дані.

Спеціалізоване обладнання для обробки зображень, як правило, складається з дігітайзера, а також апаратних засобів, які виконують інші примітивні операції, таких як арифметично-логічний пристрій (АЛП), який виконує арифметичні і логічні операції паралельно на цілих зображеннях. Одним із прикладів того, як використовується АЛУ, є усереднення

зображень так само швидко, як вони перетворюються в цифрову форму, з метою зниження рівня шуму.

Цей тип апаратних засобів іноді називається підсистемами frontend, і їх найбільш відмітною характеристикою є швидкість. Іншими словами, даний пристрій виконує функції, які вимагають швидкого проходу даних (пропускну здатність, наприклад, оцифровка і усереднення відеозображення зі швидкістю 30 кадрів/с), чого типовий комп'ютер не може виконати.

Комп'ютер в системі обробки зображення є комп'ютером загального призначення і може варіюватися від ПК до суперкомп'ютерів. У спеціалізованих додатках, іноді комп'ютери користувачів використовуються для досягнення необхідного рівня продуктивності, але інтерес цієї роботи тут в системах обробки зображень загального призначення. У цих системах, практично будь-яка добре обладнана машина PC-типу підходить для задач обробки зображень офф-лайн. Програмне забезпечення для обробки зображень складається з спеціалізованих модулів, які виконують конкретні задачі. Добре розроблений пакет також включає в себе можливість для користувача написати код, який, як мінімум, використовує спеціалізовані модулі. Більш складні програмні пакети дозволяють інтегрувати ці модулі і команди програмного забезпечення загального призначення принаймні хоча б однієї комп'ютерної мови.

Пристрій для накопичування даних є обов'язковим в додатках призначених для обробки зображень. Зображення розміром 1024x1024 пікселів, в якому інтенсивність кожного пікселя є 8-бітове число, потребує один мегабайт вільного простору, якщо зображення не стискається.

Коли маєш справу з тисячами або навіть мільйонами зображень, забезпечення адекватного зберігання в системі обробки зображень може бути проблемою. Цифрове зберігання для додатків обробки зображень можна розділити на три основні категорії:

(1) короткочасне зберігання для використання в процесі обробки,

(2) он-лайн для зберігання щодо швидкого виклику

(3) архівне зберігання, що характеризується нечастим доступ.

Зберігання вимірюється в байтах (вісім біт), кілобайтах (1024 байт), Мбайт (1024 КБайт), Гбайт (1024 МБайт), і Тбайт (1024 ГБайт).

Одним із способів забезпечення короткочасного зберігання є пам'ять комп'ютера. Інший – за допомогою спеціалізованих плат, що звуться каркасні буфери, які зберігають одне або кілька зображень і можуть бути швидко доступні, як правило, на відео швидкостях (наприклад, 30 повних зображень в секунду). Останній метод дозволяє практично миттєве масштабування зображення, а також прокрутки (вертикальні зміщення) і панорамування (горизонтальні зрушення). Буфери кадрів, як правило, розміщені в спеціалізованому апаратному блоці обробки зображень, показаному на Рис. 1.4. Он-лайн зберігання, як правило приймає форму магнітних дисків або зберігання оптичного носія. Ключовий фактор, що характеризує он-лайн зберігання - частий доступ до збережених даних. Нарешті, архівне зберігання характеризуються великим об’ємом пам’яті, що доступна для зберігання, але з нечастою необхідністю доступу. Магнітні стрічки і оптичні диски, розміщені в «музичних автоматах» є звичайними засобами збереження для архівних програм.

Дисплеї для відображення зображень сьогодні є в основному кольорові ТВ монітори. Монітори управляються виходами зображень і картами відображення зображень, які є невід'ємною частиною комп'ютерної системи. У деяких випадках, необхідно мати стерео дисплеї, і вони реалізовані у вигляді головних уборів, що містять два невеликих дисплея, вбудованих в окуляри, які носить користувач. Пристрої для запису зображень у вигляді твердої копії включають в себе лазерні принтери, камери з плівкою, чутливі до нагрівання пристрої, струменеві пристрої, а також цифрові пристрої, такі як оптичні та CD-ROM диски. Плівка забезпечує найвищу роздільну здатність, але папір є очевидним

засобом вибору для письмових матеріалів. Для презентацій, зображення відображаються на плівкових прозорих плівках або в цифровому середовищі, якщо для проектування зображення використовується спеціалізоване обладнання. Останній підхід отримує визнання в якості стандарту для представлення зображень. Навігація у мережі є майже функцією за замовчуванням в будь-який комп'ютерної системи, що використовується сьогодні. Через велику кількість даних, що обробляються програмами обробки зображень, ключовим фактором при передачі зображення є пропускна здатність. У виділеній мережі, це звичайно не є проблемою, але зв'язок з віддаленими вузлами через Інтернет не завжди настільки ефективний. На щастя, ця ситуація швидко поліпшується в результаті використання оптичного волокна і інших широкосмугових технологій.

2. МОРФОЛОГІЧНА ОБРОБКА ЗОБРАЖЕНЬ

2.1. ПОПЕРЕДНІ ВІДОМОСТІ

Слово «морфологія» зазвичай позначає галузь біології, яка має справу з формою і структурою тварин і рослин. Ми використовуваємо те ж саме слово тут, в контексті математичної морфології як інструмент для вилучення компонентів зображення, які є корисними в поданні і описі форми області, таких як кордони, скелети і опукла оболонка. Ми зацікавлені також в морфологічних методах пре- або постобработки, таких як морфологічна фільтрація, проріджування і обрізка.

У цій роботі розроблено і проілюстровано кілька важливих понять в математичної морфології. Багато з ідей, представлених тут, можуть бути сформульовані в термінах n-мірного евклідового простору, E^n. Тим не менш, наш інтерес розповсюджується також на бінарні зображення, компоненти яких є елементами Z^2.

Мовою математичної морфології є теорія множин. Таким чином, морфологія пропонує уніфікований і ефективний підхід до численних проблем обробки зображень. Множини в математичної морфології є об'єктами в зображенні. Наприклад, множина всіх білих пікселів в бінарному зображенні є повним морфологічним описом зображення. У бінарних зображеннях, множини є членами 2-D цілочисельного простору Z^2, де кожен елемент з набору є кортежем (2-D вектор), координати яких (x, y) є координатами білого (або чорного, в залежності від конвенції) пікселя в зображенні. Цифрові зображення відтінків сірого виду, розглянутого в попередніх розділах можуть бути представлені у вигляді наборів, компоненти яких знаходяться в Z^3. У цьому випадку дві компоненти кожного елемента множини відносяться до координат пікселя, а третій відповідає його дискретному значенню інтенсивності. Набори в

вищих просторах можуть містити інші атрибути зображення, такі як колір і час зміни компонентів.

У доповненні до основних визначень, поняття відображення і трансляції множини широко використовується в морфології. Відображення множини B, що позначається $\hat{B}$, визначається як:

$$\hat{B} = \{\omega \mid \omega = -b, \text{для } b \in B\}$$

Якщо B є множиною пікселів (2-D точок), що представляють об'єкт в зображенні, то $\hat{B}$ є просто набір точок в B чиї (x, y) координати були замінені на $(-x, -y)$. На малюнках 2.1 (а) і (б) показана проста множина і її відображення.

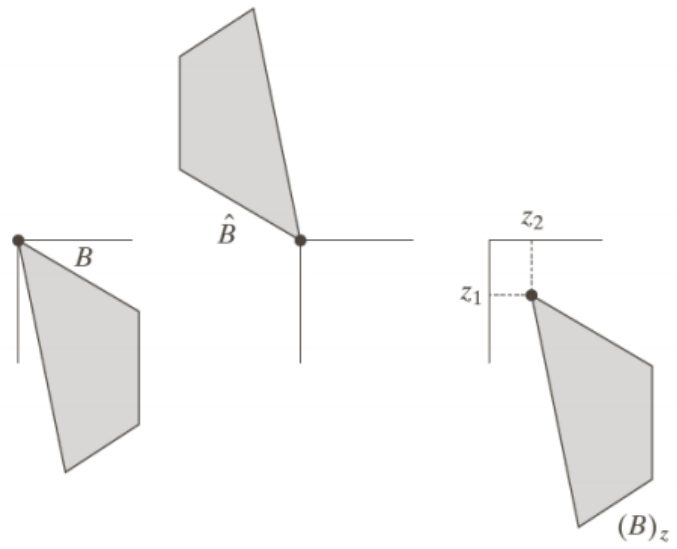


Рис. 2.1 (а),(б),(в)

Трансляція множини B від точки $z = (z_1, z_2)$, що позначається $(B)_z$, визначається як

$$(B)_z = \{c \mid c = b + z, \text{для } b \in B\}$$

Якщо B є множиною пікселів, що представляють об'єкт в зображенні, то $(B)_z$ це безліч точок в B чиї (x, y) координати були замінені на $(x + z_1, y + z_2)$ - мал 2.1 (в) ілюструє цю концепцію, використовуючи множину B з Рис. 2.1 (а).

Відображення і трансляція множин широко використовуються в морфології, щоб сформулювати операції на основі так званих структурних елементів (СЕ): невеликі множини або підзображення, що використовуються для зондування зображення при дослідженні цікавлящих досліджувача властивостей. Перший ряд на Рис. 2.2 відображає декілька прикладів структурних елементів, де кожен заштрихований квадрат позначає члена СЕ. У доповнення до визначення того, які елементи є членами СЕ, походження структурного елемента також повинна бути вказана.Начальні положення різних СЕ на Рис. 2.2 позначені чорною точка. Коли СЕ є симетричним, і крапка не показується, припускається, що початок координат знаходиться в центрі симетрії.

При роботі з зображеннями, ми вимагаємо, щоб структурні елементи були прямокутними масивами. Це досягається за рахунок додавання найменш можливого числа фонових елементів (показани незафарбленими на Рис. 2.2) необхідних для формування прямокутної матриці. Перший і останній СЕ у другому рядку на Рис. 2.2 ілюструють процедуру. Решта СЕ в цьому рядку вже знаходиться в прямокутній формі.

В якості введення до того, як структурні елементи використовуються в морфології, розглянемо Рис. 2.3. На малюнках 2.3 (а) і (б) показано просту множину і структурний елемент. Як згадувалося в попередньому абзаці, реалізація комп'ютера вимагає, щоб множина A була перетворена також у прямокутний масив шляхом додавання фонових елементів. Фонова межа зроблена досить великою, щоб вмістити весь структурний елемент, коли його центр знаходиться на кордоні оригінальної множини. В цьому випадку, структурний елемент має розмір 3x3 з початком в центрі, так що достатньо одноелементного кордону, що оточує усю множину, як показано на Рис. 2.3 (в). Як на Рис. 2.2, структурний елемент заповнений з найменшою можливою кількістю фонових елементів, необхідних, щоб перетворити його в прямокутний масив (рис. 9,3 (г)).

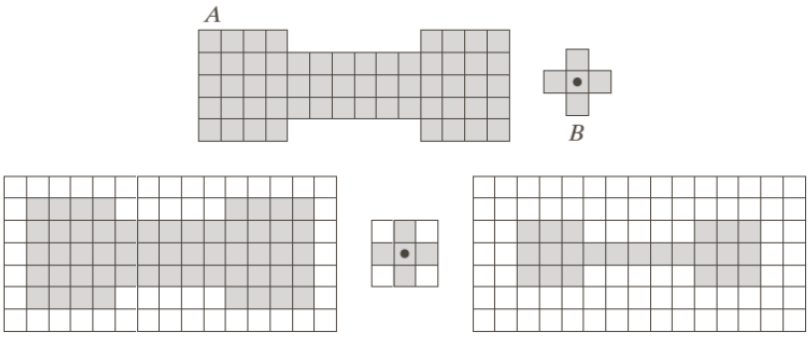


Рис. 2.3 (а,б,в,г,д)

Припустимо, що ми визначимо операцію на множині A за допомогою структурного елементу B, таким чином: створити нову множину, пробігаючи B понад A таким чином, що центр B відвідує кожен елемент A. У кожному місці проходження B, якщо B повністю міститься в A, помітити локацію як члена нової множини (показано сірим кольором); інакше помітити її як не члена нової множини (не зафарбовано)). Малюнок 9.3 (д) показує результат цієї операції. Ми бачимо, що, коли центр B знаходиться на прикордонному елементі A, частина B, перестає міститься в A, таким чином виключаючи місця, на яких B центрирован в якості можливого елемента для нового набору. Кінцевим результатом є те, що межа множини розмивається, як показано на рис. 9.3 (д). Коли ми використовуємо термінологію, такі як «структурирующий елемент міститься в наборі," ми маємо на увазі конкретно, що елементи A і B повністю перекривають один одного. Іншими словами, хоча ми показали, A і B, як масиви, що містять як заштриховані так і не заштриховані елементи, тільки заштриховані елементи обох множин розглядаються у визначенні того, міститься чи ні B в A.

2.2. ЕРОЗІЯ І ДИЛАТАЦІЯ

Ерозія і ділятація мають основоположне значення для морфологічної обробки. Справді, багато які з морфологічних алгоритмів, засновані на цих двох примітивних операціях.

З A і B у якості множин в Z^2, ерозія A на B позначається $A\Theta B$, визначається як

$$A\Theta B = \{z \mid (B)_z \subseteq A\} \tag{2.1}$$

Іншими словами це рівняння вказує на те, що ерозія A через B це множина всіх точок z таких, що B відтрансльоване на z міститься в A. В подальшому множина B, як передбачається, є структурним елементом. Рівняння (2.1) є математичною формулюванням прикладу на фіг. 2.3 (д). Оскільки твердження, що B має міститися в A рівносильно твердженню, що B не має спільних елементів з фоном, можна висловити ерозію в наступній еквівалентній формі:

$$A\Theta B = \{z \mid (B)_z \bigcap A^c = \varnothing\} \tag{2.2}$$

На Рис. 2.4 показаний приклад ерозії. Елементи A і B заштриховані, а фон білий. Тверда межа на Рис. 2.4 (в) є кордоном, за яким додаткові зміщення центру B, призведуть до того, що структурний елемент перестане повністю міститься в A. Таким чином, геометричне місце точок (розташувань центру B) в межах цього кордону являє собою ерозію A по B. Ерозію показано заштрихованою на Рис. 2.4 (в). Межу множини A показано пунктиром на рис 2.4 (в) і (д) тільки в якості посилання: вона не є частиною операції ерозії зображення. Рис. 2.4 (г) показує подовжений структурний елемент, а на рис 2.4 (д) показує ерозію A цим елементом. Треба звернути увагу, що вихіда множина еродіровала в лінію.

Рівняння (2.1) і (2.2) не єдині визначення ерозії. Проте, ці рівняння мають явну перевагу перед іншими формулюваннями в тому, що вони є більш інтуїтивно зрозумілими коли структурний елемент B розглядається як просторова маска.

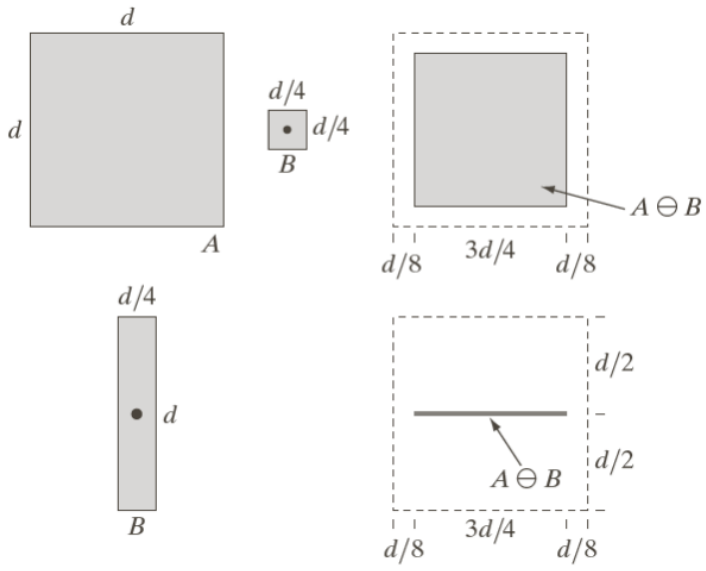


Рис. 2.4 (а),(б),(в),(г),(д)

З A і B у якості множин в Z^2, ерозія A на B позначається $A \oplus B$, визначається як

$$A \oplus B = \{z \mid (B)_z \bigcap A \neq \varnothing\} \tag{2.3}$$

Це рівняння засноване на відображенні B відносно її центру, і зміщення цього відображення на z. Таким чином, дилатація A на B це множина всіх зсувів z таких, що що B і A перекриваються, щонайменше, на один елемент. На підставі цієї інтерпретації, рівняння (2.3) можна записати у еквівалентному вигляді:

$$A \oplus B = \{z \mid [(B)_z \bigcap A] \subseteq A\} \tag{2.4}$$

Як і раніше, ми припускаємо, що B є структурним елементом і A це множина (зображення), яке повинно бути ділатоване.

Рівняння (2.3) і (2.4) не є єдиним визначенням дилатації використовуваним в даний час. Проте, попередні визначення мають явну перевагу перед іншими, в тому, що вони є більш інтуїтивними, коли структурним елемент B розглядаються в якості маски згортки. Основний процес відображення (обертання) B відносно його центру, а потім послідовне переміщення його так, що він рухається над множиною (зображенням) A аналогічно виконанню просторовій згортці.

На відміну від ерозії, яка є операцією скорочення або витончення, дилатація «збільшує» або «потовщує» об'єкти в бінарному зображенні. Специфічний спосіб й ступінь цього потовщення контролюється формою структурного елементу, який використовується. Малюнок 2.5 (а) показує той же самий набір, який використовується на Рис. 2.4, а на Рис. 2.5 (б) показаний структурний елемент (в даному випадку $B = \hat{B}$, тому що СЕ симетричний щодо його центру). Пунктирною лінією на Рис. 2.5 (в) показана оригінальна множина, а суцільна лінія показує межу, за якою будь-які подальші переміщення центру B, по z дадуть порожній перетин A і B. Таким чином, всі точки на та всередині цього кордону становлять дилатацію A на B. На Мал 2.5 (г) показаний структурний елемент, призначений для досягнення більшого ступеню дилатації вертикально, ніж горизонтально, а на Рис. 9.6 (д) показано дилатація, що досягнута з цим елементом.

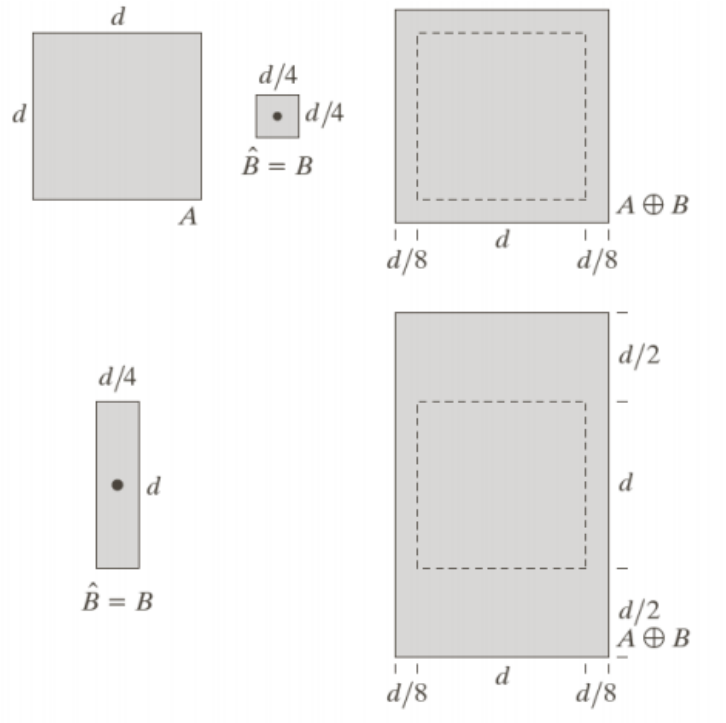


Рис. 2.5 (а),(б),(в),(г),(д)

Ерозія і дилатація є подвійними один відносно одного щодо доповнення та відображення множини. Таким чином,

$$(A\Theta B)^c = A^c \oplus \hat{B} \tag{2.5}$$

$$(A \oplus B)^c = A^c \Theta \hat{B} \tag{2.6}$$

Рівняння (2.5) вказує на те, що ерозія A на B є доповненням дилатації A^c на $\hat{B}$, і навпаки. Властивість подвійності особливо корисна, коли структурний елемент є симетричним щодо його центру (як це часто буває), так що $B = \hat{B}$. Тоді ерозію A на B можна отримати просто проведенням дилатації його фону (тобто, A^c) з тим же структурним елементом і доповнюючи результат. Те ж саме відноситься до формули. (2.6).

2.3. ВІДМИКАННЯ І ЗАМИКАННЯ

Як ми вже бачили, ділатація розширює компоненти зображень, а ерозія зменшує їх. У цьому розділі ми розглянемо дві інших важливих морфологічних операції: відмикання та замикання. Відмикання в цілому розгладжує контур об'єкта, ламає вузькі перешийки, і усуває тонкі виступи. Замикання також згладжує фрагменти контурів, але, на відміну від відкриття, воно зазвичай зливає вузькі розриви і тонкі довгі затоки, усуває дрібні отвори, і заповнює прогалини в контурі.

Відмикання множини A структурним елементом B позначається $A \circ B$ і визначається як

$$A \circ B = (A\Theta B) \oplus B \tag{2.7}$$

Таким чином, відмикання A на B є ерозією A на B, з подальшою ділацією результату за допомогою B.

Аналогічно, замикання множини A структурним елементом B позначається $A \odot B$ і визначається як:

$$A \odot B = (A \oplus B)\Theta B \tag{2.8}$$

Таким чином, замикання A через B це просто розширення A на B, з подальшою ерозією результаті до B.

Мал 2.6 ілюструє операції розмикання і замикання. Малюнок 2.6 (а) показує множину A, і на Рис. 2.6 (б) показані різні позиції структуруючого елемента під час процесу ерозії. Після завершення, цей процес привів до багатозв'язної фігури на Рис. 2.6 (в).

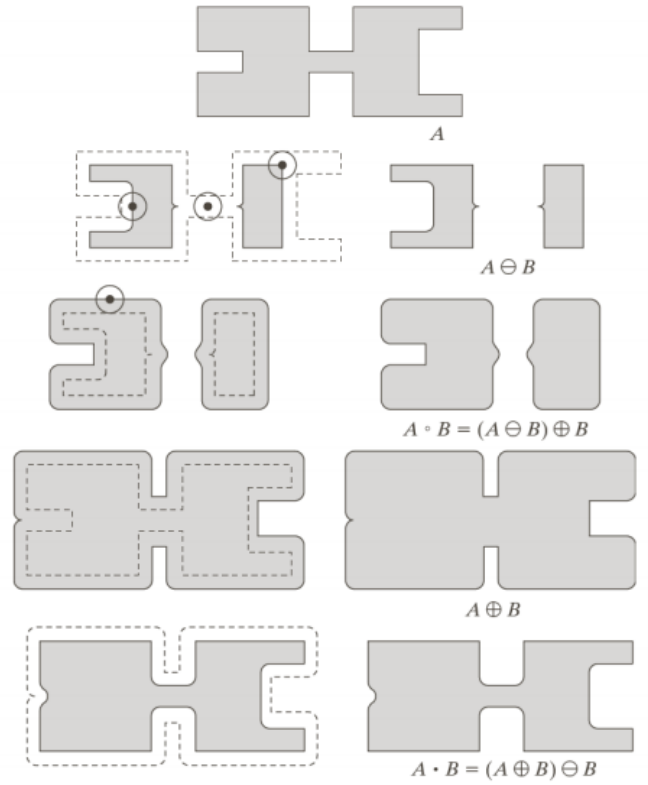


Рис. 2.6 (а),(б),(в),(г),(д),(е),(ж),(з),(і)

На Мал 2.6 (г) показаний процес дилатації розмитого набору, а на Рис. 2.6 (д) показано кінцевий результат відмикання. Треба звернути увагу, що зовнішні кути, були закруглені, а внутрішні кути не були порушені.

Аналогічно, на Рис. 2.6 (е) –(і) показані результати замикання A з тим же структурним елементом. Треба відзначити, що внутрішні кути були закруглені, в той час як зовнішні кути залишилися без змін.

2.4. ПЕРЕТВОРЕННЯ HIT OR MISS

Морфологічне перетворення Hit or miss є основним інструментом для виявлення форми на зображенні. Введемо цю концепцію за допомогою Рис. 2.7, який показує множину A, що складається з трьох форм (підмножин), що позначаються C, D і E. Штрихуванням на Рис. 2.7 (а) до (в) вказані вихідна множина, в той час як штрихуванням на Рис. 2.7 (г) і (д) вказано результат морфологічних операцій. Мета полягає в тому, щоб знайти розташування однієї з форм, скажімо, D.

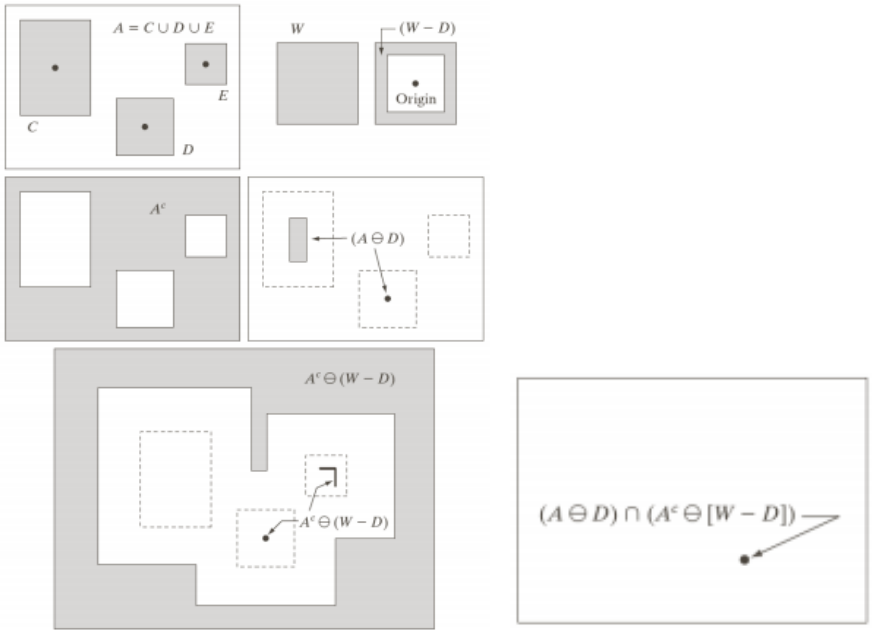


Рис. 2.7 (а),(б),(в),(г),(д),(е)

Нехай центр кожної форми розташований в його центрі ваги. Нехай D укладений в невеликому вікні, W. Локальний фон D по відношенню до W визначається як різниця множин $(W - D)$, як показано на Рис. 2.7 (б). Малюнок 2.7 (в) показує додаток A, який знадобиться подалі. Рис. 2.7 (г) показує ерозію A по D (пунктирні лінії включені для ілюстрації).

Малюнок 2.7 (д) показує ерозію доповнення до місцевого фонової множини $(W - D)$. Зовнішня заштрихована область на рис. 2.7 (д) є частиною ерозії. З Рис. 2.7 (д) і (е), зрозуміло, що множина місць розташування, для яких D точно поміщається всередину A є перетином

ерозії A по D і ерозії A^c по $(W-D)$, як показано на рис. 9,12 (е). Іншими словами, якщо B позначає множину, що складається з D і її фону, відповідність (або набір відповідностей) з B в A позначається $A \otimes B$ і визначається як:

$$A \otimes B = (A \Theta D) \bigcap [A^c \Theta (W - D)] \tag{2.9}$$

Можна дещо узагальнити цей запис, положив $B = (B_1, B_2)$, де B_1 - це множина, створена із елементів B, що асоціюються із об'єктом, і B_2 - це множина елементів B, що асоціюються з фоном ($B_1 = D$, $B_2 = W - D$). Тоді

$$A \otimes B = (A \Theta B_1) \bigcap (A^c \Theta B_2) \tag{2.10}$$

Таким чином, множина $A \otimes B$ містить усі точки, в яких одночасно B_1 знаходить відповідність («hit») в A, а B_2 знаходить відповідність в A^c. Використовуючи (2.7) можна записати:

$$A \otimes B = (A \Theta B_1) - (A^c \oplus B_2) \tag{2.11}$$

Абияка з трьох вищевказаних формул визначають операцію перетворення «Hit or miss».

2.5. ДЕЯКІ БАЗОВІ МОРФОЛОГІЧНІ АЛГОРИТМИ

При роботі з бінарними зображеннями, одне з головних застосувань морфології полягає у витягу компонентів зображення, які є дуже корисними в поданні та описі форми. Зокрема, ми розглянемо морфологічні алгоритми для вилучення кордонів, компонентів зв'язності, опуклої оболонки і каркаса регіону.

2.5.1. ВИДІЛЕННЯ ГРАНИЦІ

Границя множини A, яку обозначимо $\beta(A)$, може бути отримана, якщо спочатку провести ерозію зображення на B, а потім отримати різницю між вихідною множиною A та її ерозією. Таким чином,

$$\beta(A) = A - (A \Theta B), \tag{2.12}$$

де B - це відповідний структурний елемент.

На Рис. 2.8 (б) показано результат виділення границь для досить простого бінарного зображення з Рис. 2.8 (а).



Рис. 2.8 (а),(б)

2.5.2. ЗАПОВНЕННЯ ДИРОК

Дирка може бути визначеною як фоновий регіон, оточений межою із пікселів відмінного кольору. Нехай A позначає множину, чиї елементи є 8-зв'язними границями, кожна границя оточує регіон фонового кольору (тобто дирку). Маючи на вході точку у кожній з дирок, задача – заповнити усі дирки одиницями.

Почнемо з формування масиву X_0 такого ж розміру як масив A і який складається із нулей за виключенням тих місць, які відповідають до заданих точек у кожній дирці. Значення у цих місцях ми встановлюємо у одиницю. Наступна процедура заповнює усі дирки одиницями:

$$X_k = (X_{k-1} \oplus B) \bigcap A^c, \quad k = 1,2,3,... \tag{2.13}$$

де B - симетричний структурний елемент.

Алгоритм припиняє роботу коли $X_k = X_{k-1}$. Множина X_k тоді містить усі заповнені дирки. Об'єднання множин X_k та A містить усі заповнені дирки та їх границі.

На рис. 2.9 (б) приведено результат роботи алгоритму заповнення дирок із вихідним зображенням з рис. 2.9 (а) після однієї ітерації, а на рис. 2.9 (в) – остаточний результат роботи алгоритму.



Рис. 2.9 (а),(б),(в)

2.5.3. ВИДІЛЕННЯ ЗВ'ЯЗНИХ КОМПОНЕНТ

Виділення зв'язних компонент з бінарного зображення є одною з основних задач у системах автоматичного розпізнавання зображень.

Нехай A позначає множину, чиї елементи є один або кілька зв'язних компонент. Зформуємо масив X_0 такого ж розміру як масив A і який складається із нулей за виключенням тих точок, які відповідають до заданих точек у кожній зв'язній компоненті. Значення у цих місцях ми встановлюємо у одиницю. Задачею є, маючи масив X_0, знайти усі зв'язані компоненти. Наступна процедура виконує цю задачу:

$$X_k = (X_{k-1} \oplus B) \bigcap A, \quad k = 1,2,3,... \tag{2.14}$$

де B - структурний елемент.

Алгоритм припиняє роботу коли $X_k = X_{k-1}$. Множина X_k тоді містить усі зв'язані компоненти. На рис. 2.10 (в) представлено результат дії алгоритму на зображення з рис. 2.10(б), яке, в свою чергу, є результатом дії стандартної операції Threshold на зображення з рис. 2.10(а).

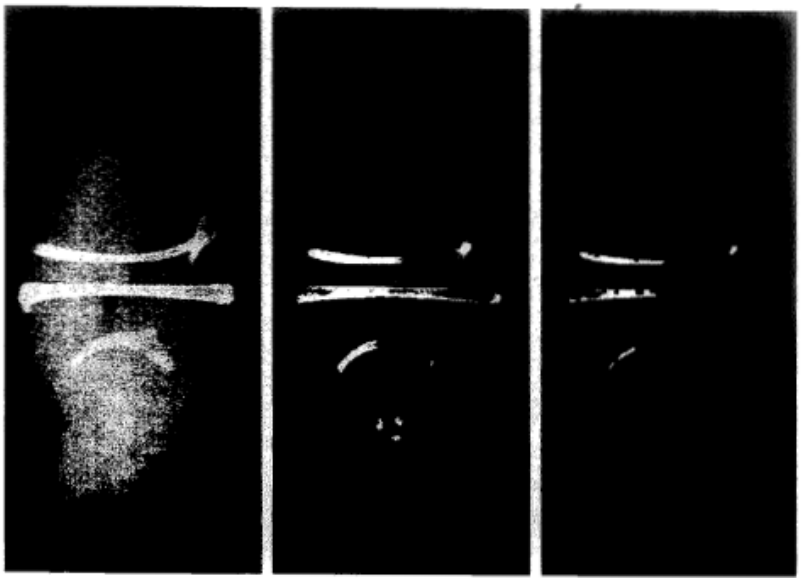


Рис. 2.10 (а),(б),(в)

2.5.4. ПОБУДОВА ОПУКЛОЇ ОБОЛОНКИ.

Множина A зветься опуклою якщо абиякий відрізок, з'єднуючий абиякі дві точки з A лежить цілком всередині A. Опукла оболонка H довільної множини S є найменшою опуклою множиною, що містить S. Різниця множин $H-S$ зветься опуклою різницею множини S. Опукла оболонка та опукла різниця є дуже корисними при описі об'єктів. Представимо простий морфологічний алгоритм отримання опуклої оболонки, $C(A)$ множини A.

Нехай B^i, $i=1,2,3,4$ представляють собой чотири структурних елемента з рис. 2.11. Алгоритм складається з реалізації рівняння

$$X_k^i = (X_{k-1} \odot B^i) \bigcup A, \quad i=1,2,3,4; \quad k=1,2,3,... \tag{2.15}$$

де $X_0^i = A$. Коли алгоритм зходиться (тобто $X_k^i = X_{k-1}^i$), положимо $D^i = X_k^i$. Тоді опукла оболонка A може бути отримана наступним чином:

$$C(A) = \bigcup_{i=1}^{4} D^i \tag{2.16}$$

Іншими словами, метод складається з ітеративного застосування перетворення «Hit or miss» до множини A із структурним елементом B.

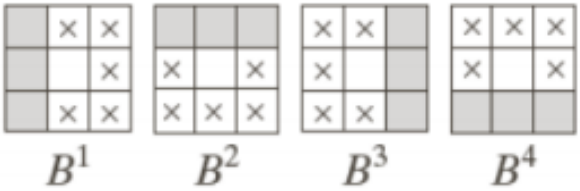


Рис. 2.11

Рис. 2.12 (б)-(д) ілюструє застосування кожного з чотирьох структурних елементів B^i до вихідної множини на рис. 2.12 (а). Рис. 2.12 (е) демонструє об'єднання X_k^i, представляючи собою опуклу оболонку множини A.

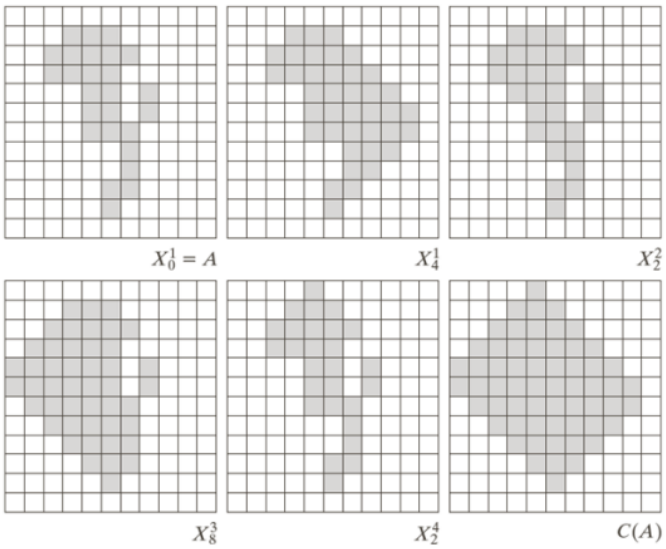


Рис. 2.12 (а),(б),(в),(г),(д),(е)

3. КОМПЮТЕРНА СИСТЕМА ДЛЯ МОРФОЛОГІЧНОЇ ОБРОБКИ ТА СТРУКТУРУВАННЯ ЕЛЕМЕНТІВ ЗОБРАЖЕННЯ.

Метою цієї дипломної роботи було створення програмної системи, що дозволяє виконувати різноманітні морфологічні операції із різноманітними структурними елементами на зображенні, що обирає користувач. У зв'язку з цим, розташування елементів управління на формі плагіну було спроектовано таким чином, що вікно плагіну розділено на три частини (Рис. 3.1).

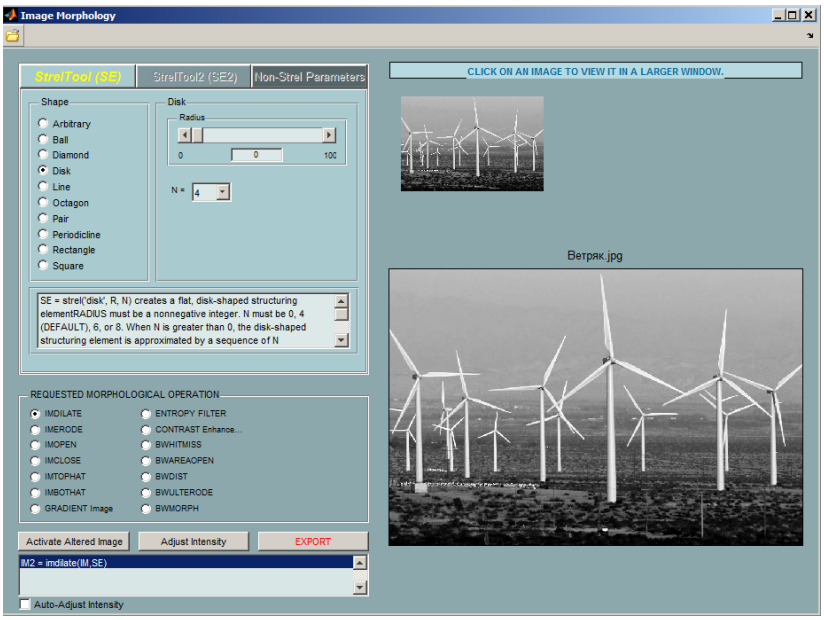


Рис. 3.1

Права частина форми містить вихідне зображення у вигляді мініатюри для попереднього перегляду та зображення після цифрової обробки. Обидва зображення (на Рис. 3.1 вони однакові тому, що ніякої цифрової обробки ще не проводилося) є клікабельними, тобто після клацання миші на абиякому з цих зображень відкривається вікно із збільшеним відповідним зображенням (Рис. 3.2)



Рис. 3.2

Клацання правою кнопкою миші на абиякому з цих двох збільшених зображень призводить до автоматичного зберігання відповідного зображення у поточному оточенні MATLAB, що гарантує простий доступ програмам, що використовують Image Morphology як плагін, до обробленого зображення.

Ліва верхня частина форми містить налаштовуємі параметри, що стосуються форми, розміру та параметричної частини структурних елементів, а також параметри, які не відносяться до операцій із структурними елементами (які більшою мірою потрібні виключно у роботі з однобітними зображеннями).

Перший із доступних для зміни параметрів структурного елемента – це його форма. Для вибору предоставлені як стандартні форми структурних елементів, так і форми, що визначаються користувачем. Системою підтримуються наступні стандартні форми структурних елементів:

- Ball - створюється неплоский, кулястий структурний елемент (насправді він має форму еліпсоїда), радіус якого в площині X-Y є R і висота якого є H. Налаштовуємимі параметрами цього структурного

елемента є радіус, висота, а також необхідність (та якість у випадку необхідності) апроксимації структурного елемента набором періодичних ліній.

- Diamond – створюється плоский, ромбоподібний структурний елемент (Рис. 3.3). Єдиним налаштовуємим параметром цього структурного елементу є відстань від центру структурного елементу до кінцевих точок алмазу. Цей параметр повинен бути невід'ємним цілим числом.

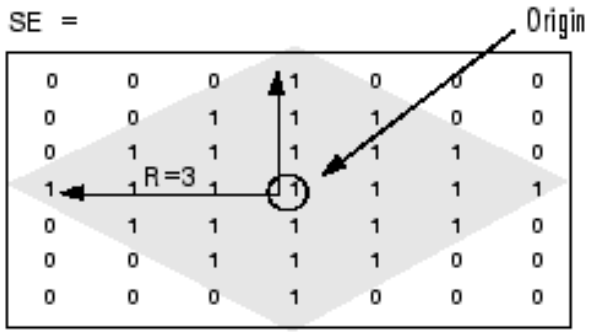


Рис. 3.3

- Disk – створюється плоский структурний елемент у формі диска (Рис. 3.4). Параметрами цього елемента є радіус диска (невід’ємне ціле число) та необхідність/якість апроксимації набором періодичних ліній.

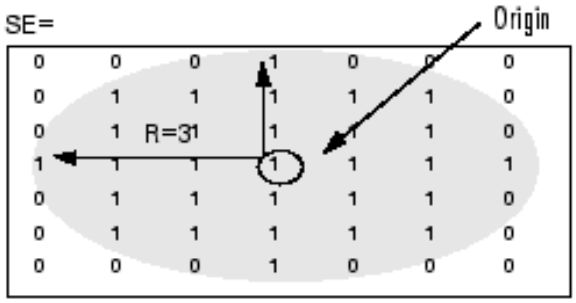


Рис. 3.4

- Line – створюється плоский лінійний структурний елемент (Рис. 3.5). Перший з двох налаштовуємих параметрів вказує дліну лінії, а другий визначає кут (в градусах) лінії, який виміряно в напрямку проти годинникової стрілки від горизонтальної осі.

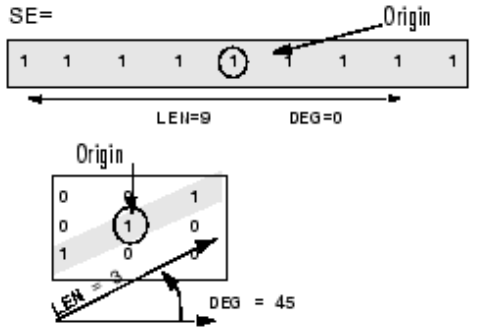


Рис. 3.5

- Octagon – створюється плоский октогональний структурний елемент (Рис. 3.6). Єдиний налаштовуємий параметр визначає відстань від центру структурного елементу до сторін восьмикутника, виміряний вздовж горизонтальної та вертикальної осей. Це повинно бути невід'ємне ціле число кратне 3.

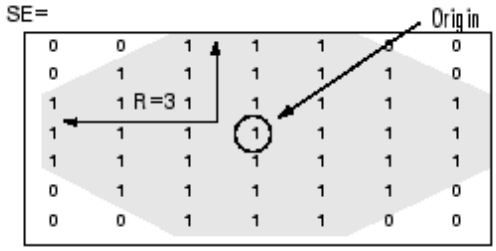


Рис. 3.6

- Pair – створюється плоский структурний елемент, який містить два піделементи (Рис. 3.7). Один піделемент розташований у центрі структурного елементу. Розташування другого члена визначається вектором зміщення. Зміщення повинно бути двохелементним вектором цілих чисел.

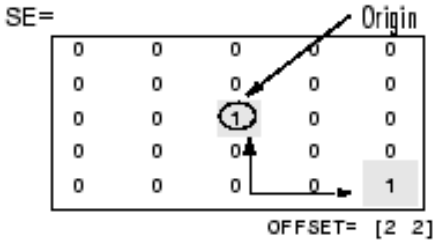


Рис. 3.7

- Periodic Line – створюється плоский структурний елемент, який містить непарну кількість членів (Рис. 3.8). Перший параметр P

визначає кількість членів (яка розраховується як 2*P+1), а другий, V, – це двохелементний вектор, що містить цілочисельні значення зміщення по рядкам і стовпцям. Один з членів структурного елементу розташований в центрі, інші члени розташовані в 1*V, -1*V, 2*V, -2*V, ..., P*B, -P*V.

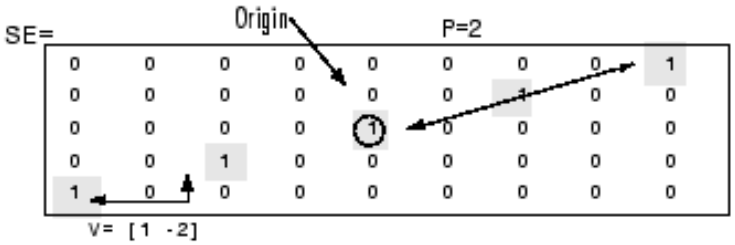


Рис. 3.8

- Rectangle – створюється плоский структурний елемент прямокутної форми (Рис. 3.9). Єдиний параметр визначає розмір прямокутника. Він повинен бути двохелементним вектором невід'ємних цілих чисел. Перший елемент є числом рядків в околицях структурного елементу; другий елемент являє кількість стовпців.

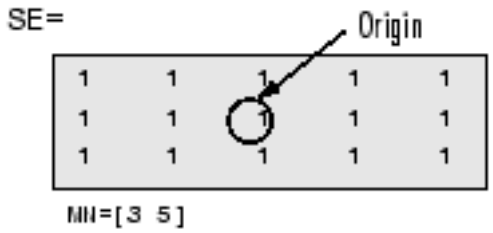


Рис. 3.9

- Square – створюється плоский квадратний структурний елемент (Рис. 3.10). Єдиний параметр визначає розмір сторони квадрата у пікселях.

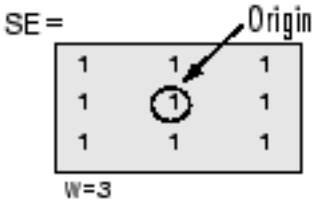


Рис. 3.10

Окрім базових форм структурних елементів, користувач має можливість створити структурний елемент сам (Рис. 3.11). Єдиним параметром при створенні структурного елемента є матриця із нулей та одиниць, яка визначає структурний елемент. Центром структурного елемента у даному випадку рахується центр матриці.

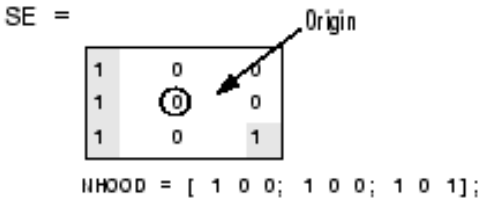


Рис. 3.11

Параметри, які не відносяться до работи із структурними елементами і які потрібні при обработці двокольорових зображень, будуть розглянуті пізніше, під час опису відповідних функцій обробки.

Ліва нижня частина форми містить доступні морфологічні операції, а також засоби для автоматичного налаштування інтесивності зображення та зберігання обробленого зображення.

Перша морфологічна операція, що реалізована в системі Image Morphology – діляція. На Рис. 3.12 приведено результат застосування діляції із структурним елементом «Ball» радіусом 10 пікселів до зображення з Рис. 3.2



Рис. 3.12

Наступна реалізована у системі морфологічна операція – ерозія. На Рис 3.13 приведено результат обробки зображення із структурним елементом Disk радіусом 6 пікселів.



Рис. 3.13

Розмикання у системі Image Morphology реалізовано із використанням вбудованої функції MATLAB imopen. На Рис. 3.14 приведено результат обробки вихідного зображення з Рис. 3.2 із структурним елементом Line довжиною 10 пікселів і кутом 45 градусів.



Рис. 3.14

Результат замикання зображення із структурним елементом Rectangle приведено на Рис. 3.15. Розмір структурного елементу – 8 рядків та 12 стовбців.



Рис. 3.15

Операція Imtophat виконує морфологічну фільтрацію «top hat» на відтінках сірого або двоїчного вхідного зображення. Imtophat обчислює морфологічне відкриття зображення (за допомогою imopen), а потім віднімає результат з вихідного зображення. На рис. 3.16 приведено результат застосування цієї операції до вихідного зображення з Рис 3.2. Структурним елементом був обраний Octagon із відстанню від центра до сторін 15 пікселів.

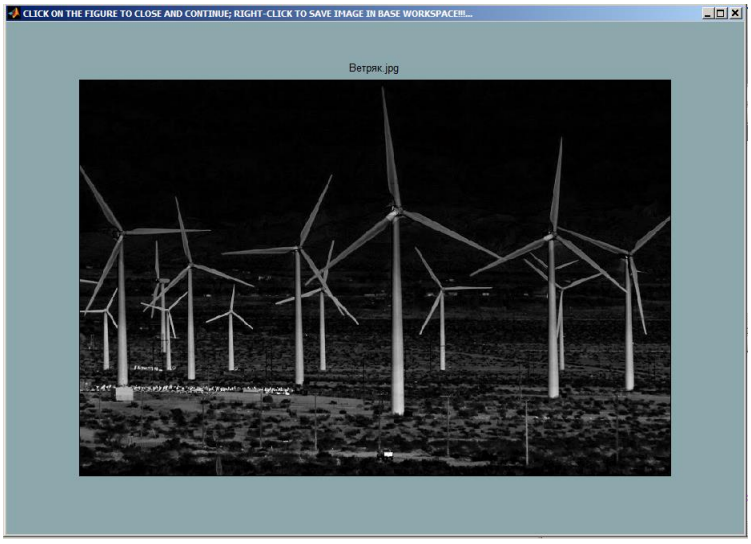


Рис. 3.16

Операція Imbothat реалізує фільтрацію «bottom hat», яка служить зазвичай для покращення чіткості зображення та збільшення його констрастності. Результат застосування цього фільтру на вихідному

зображенні з Рис. 3.2 із використанням структурного елементу «Ball» із радусом 12 пікселів та висотою 5 пікселів приведено на рис. 3.17



Рис. 3.17

Операція Entropy filter повертає масив, де кожен вихідний піксель містить значення ентропії околиці 9 на 9 пікселів навколо відповідного пікселя у вихідному зображенні. Результатом застосування цього фільтру є також зображення і на Рис. 3.18 приведено відфільтроване зображення з Рис. 3.2, нормалізоване по максимальній інтесивності.

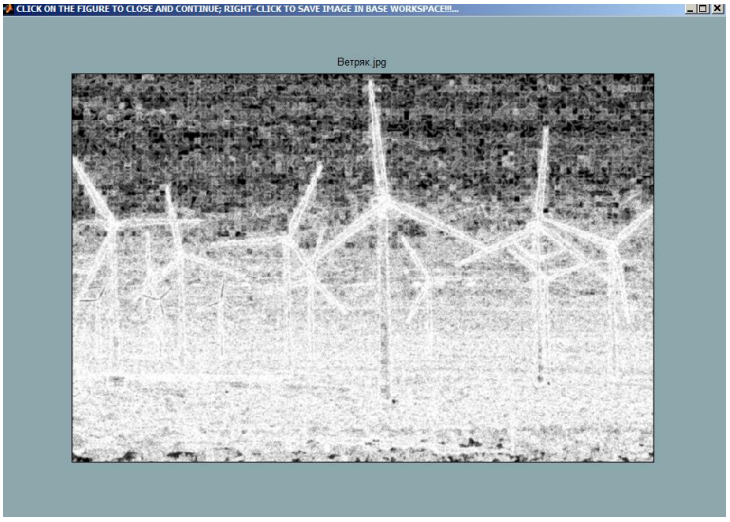


Рис. 3.18

Операція Contrast Enhance покращує контрастність зображення. Результат досягається за рахунок виконання наступної послідовності операцій: До вихідного зображення додається вихідне зображення, що оброблене фільтром Top Hat, а із результату віднімається вихідне

зображення, що обработане фільтром Bottom Hat. Для отримання результату, показаного на Рис. 3.19 був використаний струтурний елемент Disk радіусом 2 пікселя.



Рис. 3.19

Решта морфологічних операцій, що підтримуються системой Image Morphology, застосуються до однобітних зображень, таких як зображення на Рис. 3.20.

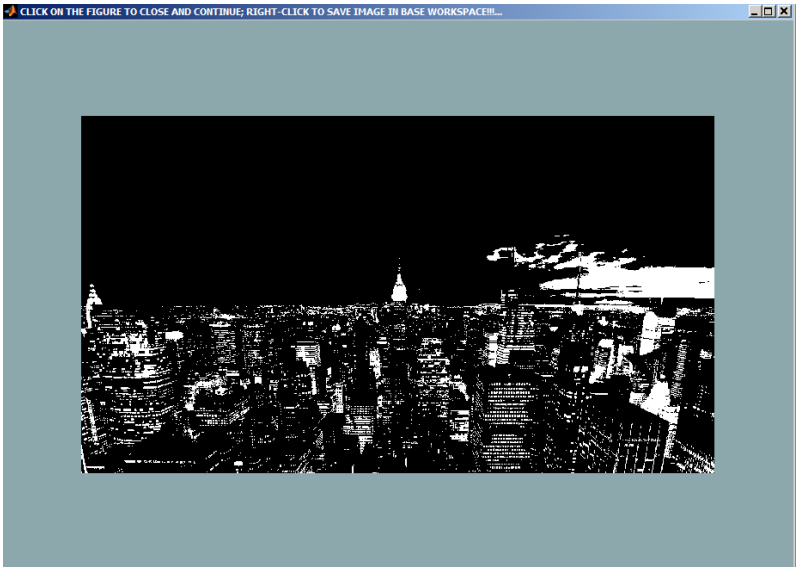


Рис. 3.20

Операція BWHitMiss реалізує фільтр Hit and Miss для бінарного зображення. Для застосування цього фільтра необхідно обрати два структурних елементи, і на Рис. 3.21 приведено результат обробки зображення з Рис. 3.20 із першим структурним елементом Ball з радіусом

10 пікселів і другим структурним елементом Diamond з відстаню до центру 3 пікселя.



Рис. 3.21

Операція BWAreaOpen видаляє з бінарного зображення усі об’єкти розміром менше ніж задане число пікселів. Для виконання цієї операції у системі ImageMorphology необхідна настройка параметрів, які не мають відношення до структурних елементів (Рис. 3.22)

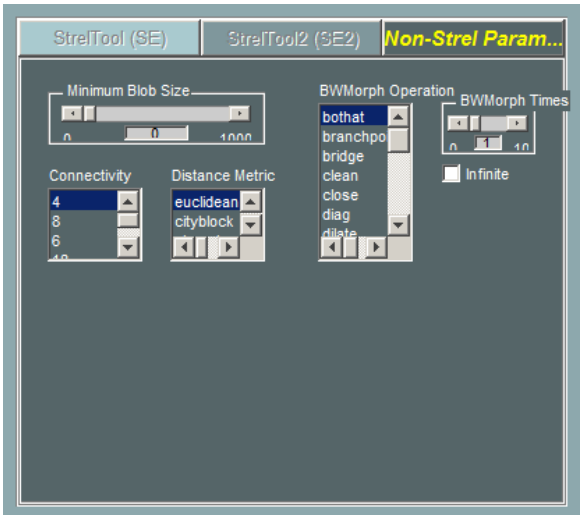


Рис. 3.22

Для обробки зображення з рис. 3.20 буі іибраний розмір видаляємих об’єктів 50 пікселів. Результат застосування фільтра приведений на Рис. 3.23.

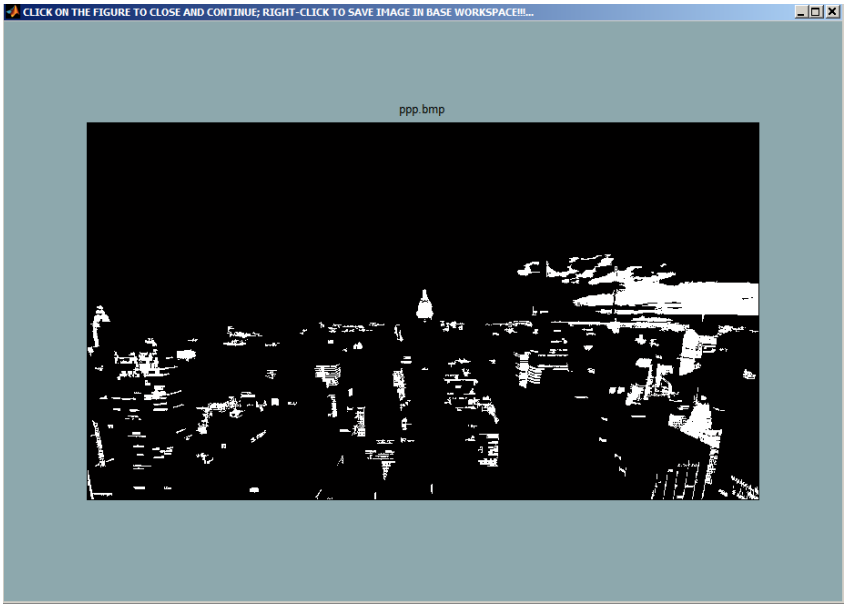


Рис. 3.23

Операція BWDist виконує перетворення відстані (Euclidean Distance Transform). Перетворення відстанні – це оператор, який зазвичай застосовуються тільки до бінарних зображень. Результатом перетворення є напівтонове зображення, яке виглядає подібно вхідному зображенню, за винятком того, що інтенсивність сірих точок всередині переднього плану регіонів змінюються, щоб показати відстань до найближчого кордону з кожної точки. Робота цього оператору проілюстрована на Рис. 3.24(б). Об’єктом перетворення було зображення з Рис. 3.24(а). В якості метрики відстані була обрана Евклідова метрика.

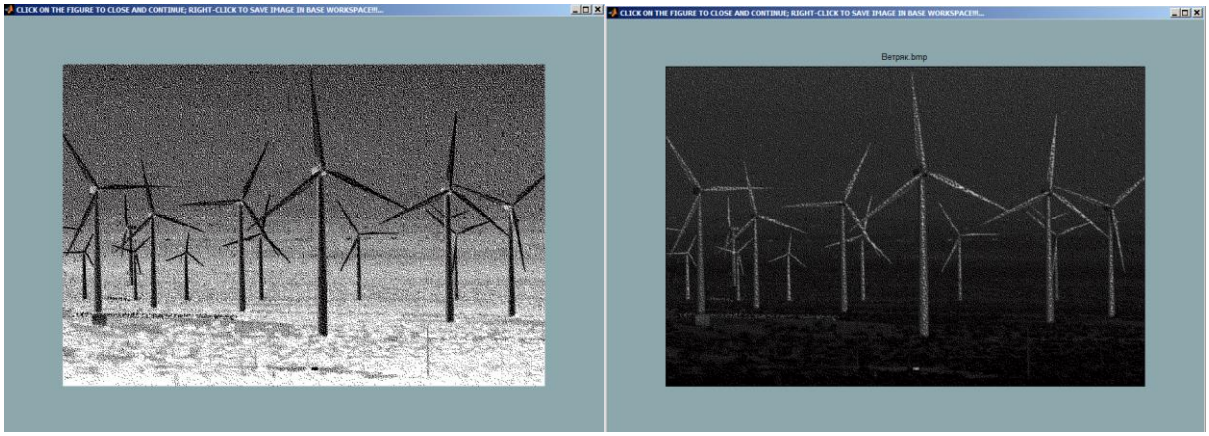


Рис. 3.24 (а),(б)

Операція BWUltErode обчислює остаточну ерозію чорно-білого зображення. Остаточна ерозія – це множина регіональних максимумів

(зв’язних наборів пікселів із однаковою інтенсивністю) перетворення відстані від доповнення зображення. На Рис. 3.25(а) та (б) показано вихідне зображення та результат дії оператора остаточної ерозії на нього.

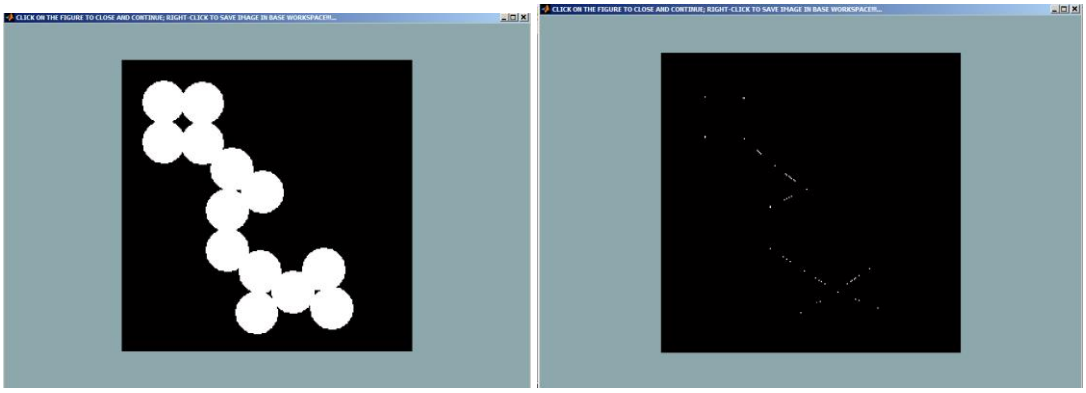


Рис. 3.25 (а),(б)

Остання морфологічна операція, що підтримується системою Image Morphology, є складовою. Ця операція у системі зветься BWMorph і в залежності від її параметрів вона може виконувати різні морфологічні операції на однобітних зображеннях. Список підтримуваних операцій виглядає таким чином:

- bothat – виконує морфологічну операцію Bottom Hat, повертаючи вихідне зображення мінус морфологічне замикання вихідного зображення.
- branchpoints – знаходить точки розгалуження зображення. Наприклад, зображення

  0 0 1 0 0 — 0 0 0 0 0
  0 0 1 0 0 — 0 0 0 0 0
  1 1 1 1 1 — перетворюється у — 0 0 1 0 0
  0 0 1 0 0 — 0 0 0 0 0
  0 0 1 0 0 — 0 0 0 0 0

- bridge – з’єднує нез’єднані пікселі. Іншими словами, встановлює значення пікселя, який має значення нуль в одиницю, якщо він має двох ненульових сусідів, які не є з’єднаними. Наприклад, зображення

  1 0 0 — 1 0 0

1 0 1	перетворюється у	1 1 1
0 0 1		0 0 1

- clean – видаляє ізольовані пікселі. Наприклад, зображення

0 0 0		0 0 0
0 1 0	перетворюється у	0 0 0
0 0 0		0 0 0

- close – виконує морфологічне замикання зображення (послідовно виконує діляцію та ерозію зображення).
- diag – використовує діагональну заливку для виключення 8-зв’язності фону. Наприклад, зображення

0 1 0		0 1 0
1 0 0	перетворюється у	1 1 0
0 0 0		0 0 0

- dilat – виконує морфологічну діляцію із використанням структурного елементу [1 1 1].
- endpoints – знаходить кінцеві точки скелету. Наприклад, зображення

1 0 0 0		1 0 0 0
0 1 0 0	перетворюється у	0 0 0 0
0 0 1 0		0 0 1 0
0 0 0 0		0 0 0 0

- erode – виконую морфологічну ерозію зображення із використанням структурного елементу [1 1 1].
- fill – заповнює ізольовані внутрішні пікселі. Наприклад, зображення

1 1 1		1 1 1
1 0 1	перетворюється у	1 1 1
1 1 1		1 1 1

- hbreak – видаляє H-зв’язні пікселі. Наприклад, зображення

1 1 1		1 1 1
0 1 0	перетворюється у	0 0 0
1 1 1		1 1 1

- majority – виставляє піксель у одиницю якщо 5 із його 8 сусідів є одиницями. В іншому випадку виставляє піксель в нуль.
- open – виконує морфологічне відмикання зображення.
- remove – видаляє внутрішні пікселі. Ця операція встановлює піксель в нуль, якщо всі його 4 сусіда мають значення 1.
- shrink – видаляє пікселі таким чином, що об’єкти без дирок перетворюються у точки, а об’єкти з дирками – зтягуються до кільція навколо дирки напівшляху між диркою та зовнішнім кордоном. Ця операція зберігає число Ейлера для об’єкту.
- skel – видаляє пікселі на границях об’єкту, але не дозволяє об’єкту розпастися на частини. Залишившиєся пікселі формують скелет об’єкту. Ця операція також зберігає число Ейлера для об’єкту.
- spur – видаляє «висячі» пікселі. Наприклад, зображення

0 0 0 0 0		0 0 0 0 0
0 0 0 0 0		0 0 0 0 0
0 0 1 0 0	перетворюється у	0 0 0 0 0
0 1 0 0 0		0 1 0 0 0
1 1 0 0 0		1 1 0 0 0

- thicken – потовщує об’єкт за рахунок додавання пікселів за зовнішньою межою об’єкту до тих пір, поки це не робить незв’язані до того об’єкти зв’язаними. Це перетворення зберігає число Ейлера.
- thin – потоншує об’єкт до лінії. Видаляє пікселі таким чином, що об’єкт без дирок стягується до мінімально зв’язаної лінії, а об’єкт з диркою – до кільця напівшляху між диркою та зовнішнім кордонгом. Ця операція зберігає число Ейлера.
- tophat – виконує морфологічну операцію «Top Hat», повертаючи зображення мінус морфологічне відмикання зображення.

4. ВИСНОВКИ.

Мета та задачі, поставлені в цієї роботі були виконані, а саме:

- було проведенно аналіз та обрані методи цифрової обробки зображень;
- був розроблений алгоритм програми;
- був розроблений інтерфейс програми;
- був реалізований програмний продукт «Image Morphology»;
- Програмний продукт був протестований.

За останні десять років цифрова обробка зображень стала одним із провідних напрямків, який активно розвивається та застосовується практично у всіх областях досліджень.

Розроблена у цій роботі система «Image Morphology» дозволяє використовувати майже всі відомі методи та алгоритми обробки зображень. Користувач може використовувати цю систему і як автономну програму, і як плагін для своєї розробки.

Використання даної системи істотно прискорить і спростить обробку та візуалізацію зображень в різних областях науки.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. http://matlab.exponenta.ru/imageprocess/book2/80.php
2. Image Processing Toolbox 6 User’s Guide. - United States of America.: United States of America. The MathWorks 2007.
3. Hanselman D., Littlefield B. R. [2001]. Mastering MATLAB 6, Prentice Hall, Upper Saddle River, NJ.
4. Jansson P. A., ed. [1997]. Deconvolution of Images and Spectra, Academic Press, NY. Kim C. E. Sklansky J. [1982]. "Digital and Cellular Convexity,"Pattern Recog., vol. 15, no. 5
5. Telescope Images and Spectra,"in Deconvolution of Images and Spectra, P.A. Jansson, ed., Academic Press
6. Gonzalez R. C., Woods R. E. [2002]. Digital Image Processing, 2nd ed., Prentice Hall, Upper Saddle River, NJ.
7. Дьяконов В. Я. MATLAB: Учебный курс. СПб.: Питер, 2001.
8. Дьяконов В. Я MATLAB 6: Учебный курс. СПб.: Питер, 2001.
9. Дьяконов В. Я, Абраменкова И. В., Круглое В. В. MATLAB 5 М.: Нолидж, 2001.
10. Рудаков Я. И., Сафонов В. И. Обработка сигналов и изображении MATLAB 5x. М.: Диалог-МИФИ, 2000.
11. Роджерс Д. Алгоритмические основы машинной графики. М.: Ми$$ 1989.
12. Роджерс Д., Адаме Дж. Математические основы машинной графики.М.: Мир, 2001.
13. Прэтптп У. Цифровая обработка изображений: В 2 т. М.: Мир.
14. Шлихт Г. Ю. Цифровая обработка цветных изображений. М.: Эко1997.
15. Быстрые алгоритмы в цифровой обработке изображений / Под Т. С Хуанга. М.: Радио и связь, 1984.

16. Kulkarni Arun D. Computer Vision and Fuzzy-Neural Systems. Prentice Hall, 2001. www.prenhall.com.
17. Filho Oge Marques & Neto Hugo Vieira. Processamento Digital de Imagens (Digital Image Processing). Brasport, 1999.
18. Нестерук В.Ф. Преобразование оптических изображений и оценка их качества // Успехи научной фотографии. – М.: Наука. – 1985. – Т. 23. – С. 93 – 102.
19. Ярославский Л.П. Введение в цифровую обработку изображений. – М.: Сов. радио, 1979. – 312 с.
20. Gonzalez R.C., Wintz P. Digital Image Processing. – Addison–Wesley. Reading. Massachusetts. – 1987. – 505 p.
21. Wallis R. An approach to the space variant restoration and enhancement of images // Proc. Symp. on Current mathematical problems in image Science. Naval Postgraduate Schol, Monterey, CA. – 1976.