

ІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»
ФАКУЛЬТЕТ ІНФОРМАЦІЙНО-КОМП'ЮТЕРНИХ ТЕХНОЛОГІЙ
КАФЕДРА БІОМЕДИЧНОЇ ІНЖЕНЕРІЇ ТА ТЕЛЕКОМУНІКАЦІЙ

УДК 616-073.757.1

«Допущено до захисту»

Завідувач кафедри біомедичної
інженерії та телекомунікацій
к.т.н., доц. Нікітчук Т.М.

« 14 » грудня 2021 р.

КВАЛІФІКАЦІЙНА РОБОТА

на тему Дослідження методів фільтрації рентгенівських зображень

за спеціальністю 163 «Біомедична інженерія»
освітній ступінь «магістр»

Виконавець роботи
студент 2 курсу, гр.БІм-20-1

10.12.21 р. 
(дата, підпис)

Кравченко Є.В.

Керівник роботи
к.т.н., доц.

10.12.2021 р. 
(дата, підпис)

Коломієць Р.О.

Житомир
2021

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»
ФАКУЛЬТЕТ ІНФОРМАЦІЙНО-КОМП'ЮТЕРНИХ ТЕХНОЛОГІЙ
КАФЕДРА БІОМЕДИЧНОЇ ІНЖЕНЕРІЇ ТА ТЕЛЕКОМУНІКАЦІЙ

Пояснювальна записка
до кваліфікаційної магістерської роботи

на тему Дослідження методів фільтрації рентгенівських зображень

Виконав: студент 2 курсу,
групи БІм-20-1 спеціальності
163 « Біомедична інженерія »

Виконав Кравченко Є.В.

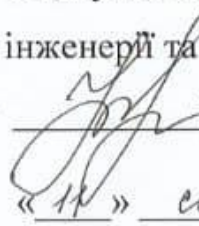
Керівник Коломієць Р.О

Рецензент Ципоренко В.В.

Житомир
2021

«Затверджено»

Завідувач кафедри біомедичної
інженерії та телекомунікацій

 Т.М. Нікітчук

« 11 » січня 2021 р.

Завдання на кваліфікаційну магістерську роботу

Кравченко Євгенія Вікторівна
(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження методів фільтрації рентгенівських зображень

затверджено наказом по університету від « 08 » листопада 2021 р. № 545/с

2. Мета дослідження: визначення найкращого методу фільтрації рентгенівських зображень для подальшої інтерпретації

3. Вихідні дані (характеристики об'єкта, умов дослідження та ін.
Тип зображення: напівтонове, рентгенівське; Кількість каналів: 256;
Тип шумів: гаусів; фіксер-шум. Типи фільтрів: морфологічний, вейвлет.

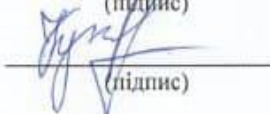
4. Основні задачі дослідження: 1) дослідити вплив шумів на початковий зображення; 2) провести аналіз роботи алгоритмів фільтрації рентгенівських зображень; 3) визначити найкращий метод фільтрації рентгенівських зображень

5. Термін подання роботи до захисту грудень 2021 р.

6. Дата видачі завдання « 11 » січня 2021 р.

Науковий керівник

Завдання прийняв
до виконання


(підпис)

(підпис)

Коломієць Р.О.
(прізвище, ініціали)
Кравченко Є.В.
(прізвище, ініціали)

АНОТАЦІЯ

Дана магістерська робота присвячена дослідженню методів фільтрації рентгенівських зображень. Основною областю засновування дослідження повинні стати програми для перегляду та обробки рентгенівських зображень.

В роботі було проаналізовано існуюче програмне забезпечення для перегляду та обробки рентгенівських зображень.

Для дослідження було створено алгоритм та програму. Промодельовані шуми на зображеннях та досліджено три методи фільтрації. Оцінка методів фільтрації зроблена за максимальною похибкою. При дослідження найкращі результати за максимальною похибкою показав вейвлет-фільтр.

Ключові слова: рентгенівські зображення, шуми на зображеннях, фільтрація зображень, вейвлет-фільтр.

ANNOTATION

This master's thesis is devoted to the study of X-ray image filtering methods. The main area of foundation of the study should be programs for viewing and processing X-ray images.

The existing software for viewing and processing X-ray images was analyzed. An algorithm and a program were created for the research. The noise in the images was simulated and three filtering methods were investigated. The evaluation of filtration methods is based on the maximum error. In the study, the best results on the maximum error showed the wavelet filter.

Keywords: X-ray images, noise in images, image filtering, wavelet filter.

РЕФЕРАТ

Дана магістерська робота присвячена дослідженню методів фільтрації рентгенівських зображень. Основною областю засновування дослідження повинні стати програми для перегляду та обробки рентгенівських зображень.

Робота складається з вступу, трьох розділів з висновками, загальних висновків, переліку використаних джерел (24 джерел), та двох додатків. Робота містить 3 таблиці, 6 формул, 35 рисунків, основний матеріал викладений на 56 сторінці. Повний обсяг роботи становить 67 сторінки.

У вступі обґрунтовано актуальність розробки та сформульовано мету, об'єкт та предмет розробки і поставлені задачі.

У першому розділі проведено огляд та аналіз існуючих програм для перегляду та обробки рентгенівських зображень та розглянуті основні методи обробки рентгенівських зображень.

У другому розділі детально розглянуті мови програмування для фільтрації рентгенівських зображень. Детально проаналізовані інструменти для обробки зображень в різних мовах програмування. Перераховані переваги та недоліки мов програмування C#, Python та Matlab. Обрано для програмування мову Matlab.

У третьому розділі проведено розробку алгоритму та програми для дослідження фільтрації рентгенівських зображень. Згідно розробленої програми проведено дослідження фільтрації рентгенівських зображень.

Оцінка фільтрації проведена за максимальною похибкою. Було досліджено медіанний, усереднюючий та вейвлет-фільтр. При дослідженні найкращі результати за максимальною похибкою показав вейвлет-фільтр.

У висновках коротко наведено виконання всіх поставлених задач.

У додатках наведено програму для дослідження фільтрації рентгенівських зображення.

ЗМІСТ

ВСТУП.....	7
1 ОГЛЯД ІСНУЮЧИХ РІШЕНЬ З ОБРОБКИ РЕНТГЕНІВСЬКИХ ЗОБРАЖЕНЬ.....	9
1.1 Методи обробки рентгенівських зображень.....	9
1.2 Існуючі програми для обробки рентгенівських зображень	20
Висновок до першого розділу.....	25
2 ВИБІР МОВИ ПРОГРАМУВАННЯ ДЛЯ ФІЛЬТРАЦІЇ РЕНТГЕНІВСЬКИХ ЗОБРАЖЕНЬ	26
2.1 Мови програмування для фільтрації рентгенівських зображень.....	26
2.2 Вибір мови програмування.....	38
Висновок до другого розділу.....	42
3 ДОСЛІДЖЕННЯ МЕТОДІВ ФІЛЬТРАЦІЇ РЕНТГЕНІВСЬКИХ ЗОБРАЖЕНЬ.....	43
3.1 Аналіз методів фільтрації рентгенівських зображень.....	43
3.2 Дослідження методів фільтрації рентгенівських зображень	47
Висновок до третього розділу.....	56
ВИСНОВОК.....	57
ДОДАТКИ.....	58
Додаток А Програма для дослідження фільтрації рентгенівських зображень.....	59
Додаток Б Обробка зображень.....	60

ВСТУП

Актуальність теми. Останнім часом одним із актуальних напрямів розвитку комп'ютерних технологій у медицині стає обробка цифрових зображень: покращення якості зображення, відновлення пошкоджених зображень, розпізнавання окремих елементів. Розпізнавання патологічних процесів є одним із найважливіших завдань обробки та аналізу рентгенівських зображень.

Відповідно, виникла ідея застосувати комп'ютерні технології до області розпізнавання зображень патологічних процесів там, де лікарі займаються обробкою та аналізу зображень, а саме до аналізу знімків і, для точніше, рентгенівських знімків.

Рентгеноскопія застосовується для діагностики широкого спектра захворювань та ушкоджень: ушкодження легень (пневмонія, ракові захворювання), переломи та інші ушкодження кісток, частина діагностики роботи системи травлення та багато іншого.

Рентгенографія – технічно простий і найдоступніший для населення метод попередньої діагностики захворювання, у тому числі відстеження захворювання на ранній стадії. Даний тип дослідження широко поширений по всьому світу та всупереч наявності інших, більш складних та ефективних способів діагностики, він залишається пріоритетним у медичній практиці навіть у добре оснащених клініках та медичних центрах. Незважаючи на всі переваги даного методу, аналіз та виявлення ознак захворювання за флюорографічними знімками є дуже складним завданням, що вимагає участі кількох висококваліфікованих фахівців. Складність завдання полягає, як правило, у низькій роздільній здатності зображень і наявності різноманітних “шумів” (засвічене зображення, наявність сторонніх предметів тощо.).

Важливо, що у діагностиці частини цих захворювань рентгенівський знімок та його інтерпретація є превалюючим інструментом у постановці діагнозу.

Інтерпретація знімка, своєю чергою, робиться лікарем-рентгенологом з урахуванням візуального аналізу зображень.

Для того щоб якісно поставити діагноз лікарю-рентгенологу потрібно щоб зображення було гарної якості. Для цього отримане рентгенівське зображення потрібно буде попередньо обробити. При обробці та аналізі рентгенівських зображень виділяють такі основні етапи: фільтрація; попередня обробка; сегментація; виділення контурів; розпізнавання; діагностика.

Метою даної роботи є визначення найкращого методу фільтрації рентгенівських зображень для подальшої інтерпретації знімка лікарем-рентгенологом щоб встановити діагноз.

Об'єктом досліджень є рентгенівські зображення для подальшої інтерпретації знімка лікарем-рентгенологом щоб встановити діагноз.

Предметом досліджень є методи фільтрації рентгенівських зображень для подальшої інтерпретації знімка лікарем-рентгенологом щоб встановити діагноз.

Задачі для досягнення мети роботи наступні:

1. Проаналізувати існуючі шуми на рентгенівських зображеннях;
2. Проаналізувати існуючі програми для обробки рентгенівських зображеннях;
3. Проаналізувати існуючі та обрати мову програмування для дослідження фільтрації рентгенівських зображень;
4. Дослідити існуючі методи фільтрації зображень;
5. Обрати оптимальний метод для фільтрації рентгенівських зображень.

Структура та обсяг роботи. Дана робота складається з анотації, вступу, 3 основних розділів, висновків, переліку використаних літературних джерел 24 та 2 додатки. Робота викладена на 59 сторінках загального обсягу та 65 основного тексту.

1. ОГЛЯД ІСНУЮЧИХ РІШЕНЬ З ОБРОБКИ РЕНТГЕНІВСЬКИХ ЗОБРАЖЕНЬ

1.1 Методи обробки рентгенівських зображень

При обробці та аналізі рентгенівських зображень виділяють такі основні етапи:

- фільтрація;
- попередня обробка;
- сегментація;
- виділення контурів;
- розпізнавання;
- діагностика.

Рентгенографія – технічно простий і найдоступніший для населення метод попередньої діагностики захворювання, у тому числі відстеження захворювання на ранній стадії. Даний тип дослідження широко поширений по всьому світу та всупереч наявності інших, більш складних та ефективних способів діагностики, він залишається пріоритетним у медичній практиці навіть у добре оснащених клініках та медичних центрах. Незважаючи на всі переваги даного методу, аналіз та виявлення ознак захворювання за флюорографічними знімками є дуже складним завданням, що вимагає участі кількох висококваліфікованих фахівців. Складність завдання полягає, як правило, у низькій роздільній здатності зображень і наявності різноманітних “шумів” (засвічене зображення, наявність сторонніх предметів тощо.).

Крім шумів, додаткові труднощі вносить наявність на знімках природних анатомічних структур (наприклад, кісток, інших органів, що перекривають), які, часто, приховують аномалії, що містяться в легеневій тканині. Виділення та наступна фільтрація подібного роду структур, із збереженням інформації про легеневу чи кісткову тканину є важливо, що в

свою чергу використовуються для рентгенографічного дослідження і може значно підвищити точність виявлення аномалій. [1]

Відповідно потрібно розглянути види шумів, які притаманні рентгенівським знімкам.

Відповідно, шуми бувають різних видів. Найбільш адекватними з точки зору використання у практичних завданнях є моделі адитивного Гаусового, імпульсного та мультиплікативного шуму.

Адитивний Гаусів шум характеризується додаванням до кожного пікселю зображення значень відповідного нормального розподілу з нульовим середнім значенням. Такий шум зазвичай вводиться на етапі формування цифрових зображень.

Гаусів шум є найпростішим з математичної точки зору. Спектральні складові цього шуму рівномірно розподілені по всьому діапазону задіяних частот. Прикладами білого шуму є шум водоспаду чи ефірні перешкоди. У природі та техніці «чисто» білий шум (тобто білий шум, що має однакову спектральну потужність на всіх частотах) не зустрічається (через такий сигнал мав би нескінченну потужність), проте під категорію білих шумів потрапляють будь-які шуми, спектральна щільність яких однакова (або слабо відрізняється) у аналізованому діапазоні частот. Функція густини розподілу Гаусового шуму випадкової величини z має вигляд (1.1):

$$p(z) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(z-\mu)^2}{2\sigma^2}}, \quad (1.1)$$

де z (стосовно тематики даної роботи) є значення яскравості,

μ – середнє значення випадкової величини z ,

σ – її середньоквадратичне відхилення.

Графік функції представлений на наведеному рис.1.1 Зображення з гаусовим шумом представлено на рис.1.2

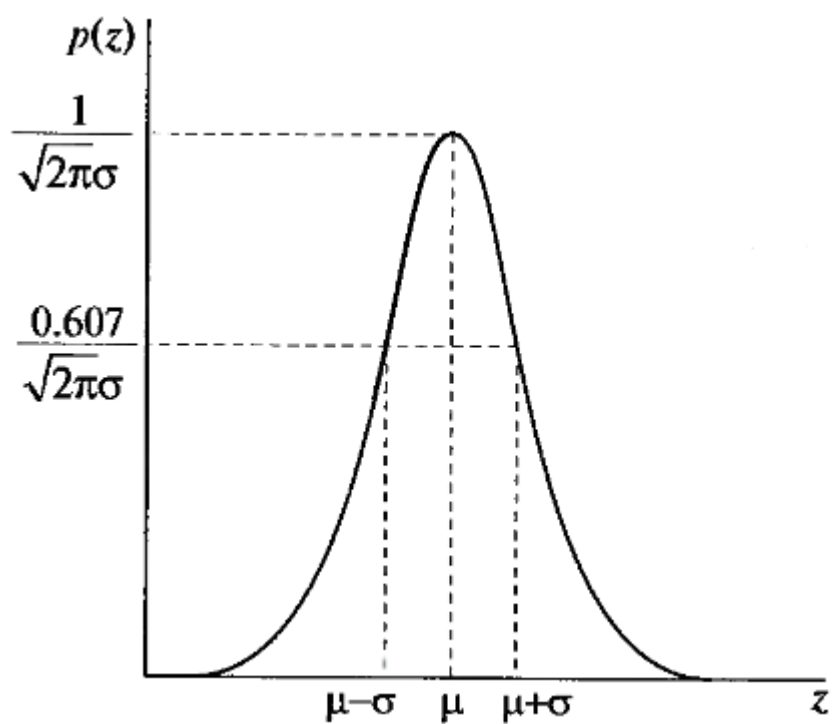


Рисунок 1.1 – Графік щільності розподілу Гаусового шуму



Рисунок 1.2 – Рентгенівське зображення з Гаусовим шумом

Момент другого порядку гаусівського білого шуму некорельований, а момент першого порядку постійний, що стосується кореляції послідовних сигналів у часі [2].

Імпульсний шум характеризується заміною частини пікселів на зображенні значеннями фіксованої чи випадкової величини. Така модель шуму пов'язана, наприклад, з помилками під час передачі зображень.

Імпульсний шум також відомий, як шум «солі та перцю» (salt-and-pepper), який випадково змінює деякі значення пікселів, являє собою чорно-білий світловий і темний шум, створюваний датчиками зображення, каналами передачі та обробкою декодування.

Через обрізання зображень часто виникає шум «солі та перцю». Рентгенівське зображення з імпульсним шумом (представлене на рис. 1.3).



Рисунок 1.3 – Рентгенівське зображення з імпульсним шумом

Мультиплікативний шум зазвичай викликається недосконалими каналами. Зв'язок між ними та сигналом – це множення. Якщо сигнал знаходиться в ньому, його не буде, якщо немає сигналу.

Рентгенівське зображення з мультиплікативним шумом представлено на рис. 1.4 [3]



Рисунок 1.4 – Рентгенівське зображення з мультиплікативним шумом

Від результатів фільтрації та попередньої обробки безпосередньо залежить ефективність наступних етапів обробки зображень [4, 5]. Етап фільтрації необхідний для зменшення різноманітних завад. Існують різні типи фільтрів: низькочастотні, високочастотні, медіани, адаптивні та інші види цифрових фільтрів. У цифровій обробці зображень широко використовується лінійна фільтрація. Вона виходить з використання швидких алгоритмів згортки. Статичні маски фільтрів який завжди гарантують прийнятний результат, так як лінійні фільтри призводять до згладжування перепадів яскравості, а це у свою чергу ускладнює завдання виділення меж. Нелінійна

фільтрація має ряд переваг у порівнянні з лінійною: менше спотворює перепади яскравості, що дає можливість точніше знаходити межі об'єктів та прибирає імпульсні завади.

Ще один вид фільтрації – адаптивна фільтрація. Даний тип фільтрації має ряд переваг, таких як: локальна фільтрація, зміна локальної маски фільтра та розміру апертури фільтра .

Найчастіше використовуються усереднюючий фільтр і медіанний фільтр.

Усереднюючий фільтр замінює значення яскравості поточної точки відеозображення на середню яскравість, обчислену на основі значень яскравості 8 сусідніх точок і початкової яскравості даної точки формула (1.2) та (рис. 1.5).

$$\bar{Y} = \frac{1}{9} \sum_{i=-1}^1 \sum_{j=-1}^1 Y_{ij} \quad (1.2)$$

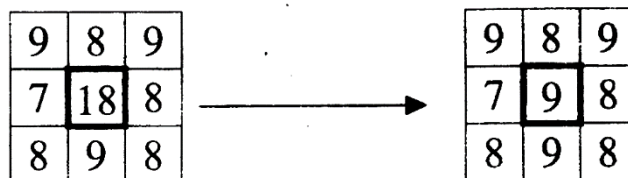


Рисунок 1.5 – Приклад роботи усереднюючого фільтра

Усереднюючий фільтр є самим простим фільтром і не потребує виконання великого об'єму обчислень. Недоліками даного фільтра є згладжування ступеневих перепадів яскравості, що відповідають контурам об'єктів. Крім того, точки, що по яскравості значно відрізняються від інших, будуть вносити суттєвий вклад в результат фільтрації. Тому усереднюючий фільтр видаляє значну частину шумів з відеозображення, але при цьому знижується чіткість

відеозображення, згладжуються контури об'єктів і можуть загубитися дрібні деталі відеозображення.

Медіанний фільтр є одним з варіантів нелінійного рангового фільтра. Медіанний фільтр являє собою вікно, що переміщується по відеозображенню і охоплює непарне число елементів (точок). Дія цього фільтра зводиться до заміни значення в точці зображення медіану значень яскравості в околиці цієї точки. При обчисленні медіани значення в самій точці також враховується (рис. 1.5).

Медіаною дискретної послідовності з N елементів при непарному N називається елемент, для якого існує $(N-1)/2$ елементів, менших або рівних йому по величині, і $(N-1)/2$ елементів, більших або рівних йому по величині.

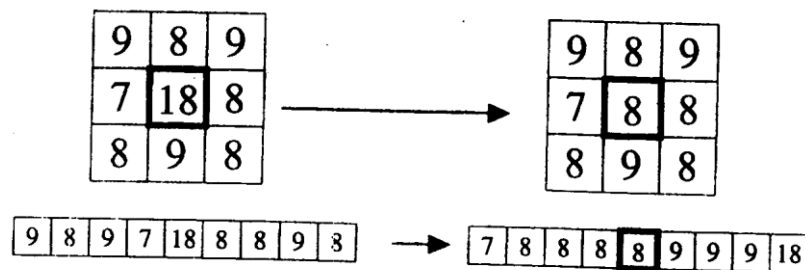


Рисунок 1.6 – Приклад роботи медіанного фільтра

Медіанний фільтр забезпечує видалення шуму, але в окремих випадках може викликати небажане зниження рівня сигналу, що швидко змінюється. Медіанний фільтр не впливає на степеневі функції, що є його корисною властивістю, однак він не пропускає імпульсні сигнали, тривалість яких складає менше половини ширини вікна фільтра.

Широка популярність цих фільтрів обґрунтована тим, що вони чудово пристосовані до пригнічення випадкових шумів, і при цьому призводять до найменшого розмивання, порівняно з іншими фільтрами. Медіанний фільтр більш ефективно видаляє імпульсні завади, ніж шуми типу “білого” шуму з

нормальним законом розподілу. Медіанний фільтр слід вважати евристичним методом обробки відеозображень.

Усереднюючий і медіанний фільтр є ефективними тільки по відношенню до дробового шуму, який проявляється на відеозображенні у вигляді імпульсних завад, тобто чорних і білих точок. Якщо викривлення на відеозображенні мають інший вид, наприклад, це є “білий” шум з нормальним законом розподілу, то більш ефективним вважається спектральний підхід до фільтрації відеозображень. [3]

Для усунення інших шумів краще працюють вейвлет-фільтри, що засновані на вейвлет-функціях.

Вейвлет (англ. wavelet – невелика хвиля, бриж; також сплеск, рідше – вейвлет) – математична функція, що дозволяє аналізувати різні частотні компоненти даних. Графік функції виглядає як хвилеподібні коливання з амплітудою, що зменшується до нуля далеко від початку координат. Однак це окреме визначення – у загальному випадку аналіз сигналів проводиться у площині вейвлет-коефіцієнтів (масштаб – час – рівень) (Scale-Time-Amplitude). Вейвлет-коефіцієнти визначаються інтегральним перетворенням сигналу. Отримані вейвлет-спектрограми принципово від звичайних спектрів Фур'є тим, що дають чітку прив'язку спектра різних особливостей сигналів до часу.

Існує кілька підходів до визначення вейвлета: через масштабний фільтр, масштабну функцію, вейвлет-функцію. Вейвлети можуть бути ортогональними, напівортогональними, біортогональними. Вейвлетні функції можуть бути симетричними, асиметричними та несиметричними, з компактною областю визначення та не мають такої, а також мати різний ступінь гладкості.

Приклади вейвлетів [6]:

- вейвлет Хаара;
- вейвлети Добеші;
- вейвлети Гауса;

- вейвлет Меєра;
- вейвлети Морле;
- вейвлет Пауля;
- вейвлет МНат («Мексиканська капелюх»);
- вейвлети Койфмана – койфлети;
- вейвлет Шеннона.

Для фільтрації рентгенівських зображень доречно використати вейвлети Хаара та Добеши.

Вейвлет Хаара – один із перших та найпростіших вейвлетів. Вейвлети Хаара ортогональні, мають компактний носій, добре локалізовані в просторі, але не є гладкими.

Перетворення Хаара використовується для стиснення вхідних сигналів, компресії зображень, переважно кольорових і чорно-білих з плавними переходами. Цей вид фільтрації відомий досить давно і безпосередньо виходить з ідеї використання когерентності областей.

Ідеальний для картинок рентгенівських знімків.

В математиці вейвлет Хаара є послідовністю масштабованих «прямокутних» функцій, які разом утворюють сімейство вейвлетів або основу. Вейвлет-аналіз аналогічний аналізу Фур'є у цьому, що дозволяє представити цільову функцію на інтервалі в термінах ортонормованого базису. Послідовність Хаара тепер визнана першою відомою основою вейвлетів і широко використовується як навчальний приклад.

Вейвлет Хаара також є найпростішим із можливих вейвлетів. Технічний недолік вейвлета Хаара полягає в тому, що він не безперервний і, отже, не диференційований.

Послідовність Хаара була запропонована в 1909 Альфредом Хааром. Хаар використовував ці функції, щоб дати приклад ортонормованої системи для простору функцій, що квадратично інтегруються, на одиничному інтервалі $[0, 1]$. Вивчення вейвлетів і терміна «вейвлет» з'явилося набагато пізніше. Як окремий випадок вейвлета Добеши, вейвлет Хаара також відомий як Db1.

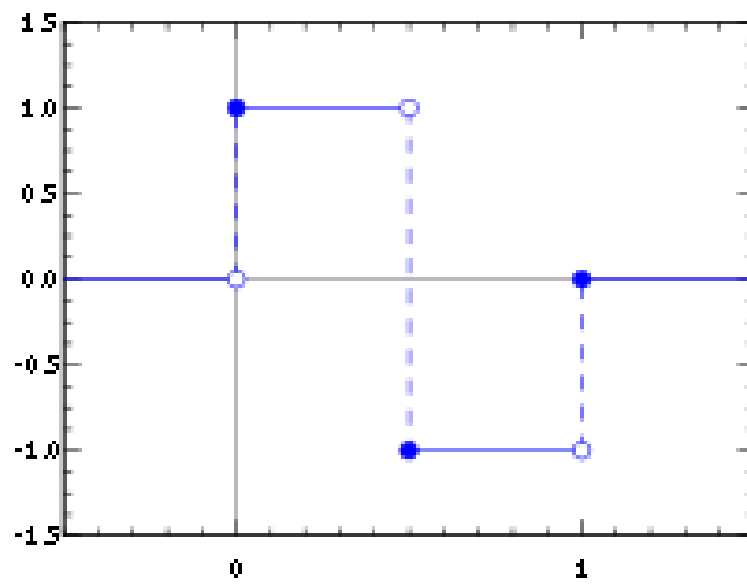


Рисунок 1.7 – Вейвлет Хаара

Породжуюча вейвлет-функція $\psi(x)$ з нульовим значенням інтеграла

$\int_{-\infty}^{\infty} \psi(x) dx = 0$, що визначає деталі сигналу, задається таким чином:

$$\psi(x) = \begin{cases} 1, & 0 \leq x < 1/2 \\ -1, & 1/2 \leq x < 1 \\ 0, & x \notin [0, 1) \end{cases} \quad (1.3.)$$

Масштабуюча функція $\psi(x)$ з одиничним значенням інтеграла

$\int_{-\infty}^{\infty} \psi(x) dx = 1$, що визначає грубе наближення (апроксимацію) сигналу,

постійна:

$$\psi(x) = \begin{cases} 1, & 0 \leq x < 1 \\ 0, & x \notin [0,1) \end{cases} . \quad (1.4)$$

Вейвлети Добеші (англ. Daubechies wavelet) – сімейство ортогональних вейвлетів з компактним носієм, що обчислюється ітераційним шляхом. Названі на честь математика зі США, яка першою побудувала дане сімейство, Інгрід Добеші.



Рисунок 1.8 – Вейвлет Добеші другого порядку

Для побудови вейвлетів скористаємось рівнянням розтягування та вейвлет-рівнянням:

$$\varphi(x) = \sqrt{2} \sum_k h_k \varphi(2t - k), \quad (1.5)$$

$$\psi(x) = \sqrt{2} \sum_k g_k \varphi(2t - k). \quad (1.6)$$

Компактність носія функцій φ та ψ може бути досягнута, якщо буде вибрано кінцеве число $h_n \neq 0$ таким чином, щоб було досягнуто ортогональності та гладкості вейвлету, або щоб виконувалася умова моментів [7].

Методи, що використовуються на етапі попередньої обробки, залежать від завдань досліджень і вони досить різноманітні. Вони можуть включати виділення найбільш інформативних фрагментів, їх збільшення, зміна просторового дозволу, зміна контрастного дозволу.

Одні з основних дій, які проводяться на етапі попередньої обробки, це зміна контрастності та яскравості зображення. Методи зміни контрастності та яскравості поділяються на лінійні, нелінійні та адаптивні. При застосуванні відповідних масок можна з'єднати два етапи (етап фільтрації та етап попередньої обробки) для забезпечення швидкодії. Етап попередньої обробки також передбачає геометричні операції над зображенням. До них відносяться методи повороту зображення, збільшення та зменшення зображення [8].

Так, як використовуються рентгенівські зображення, а вони в чорно-білому кольорі та найчастіше мають шуми, то основна обробка над такими зображеннями що буде використовуватися – це фільтрація.

1.2 Існуючі програми для обробки рентгенівських зображень

Одна з самих популярних програм є Radiant. Вона оптимально працює на наступних Windows XP (Service Pack 3), Windows Vista, Windows 7, Windows 8 і 8,1, Windows 10. Дана програма дуже проста у використанні так, як максимально інтуїтивна. Великим плюсом цієї програми є те, що вона перекладена російською мовою від розробника. Невеликий бонус простоти використання полягає в тому, що користувачеві не потрібно завантажувати файли з дисків на комп'ютер, а програма зробить це за вас, автоматично відкривши ваші дослідження з диска. Ця програма не вимагатиме від вас

додаткових програм, таких як JAVA або NET, що значно полегшує процес перегляду DICOM - файлів.

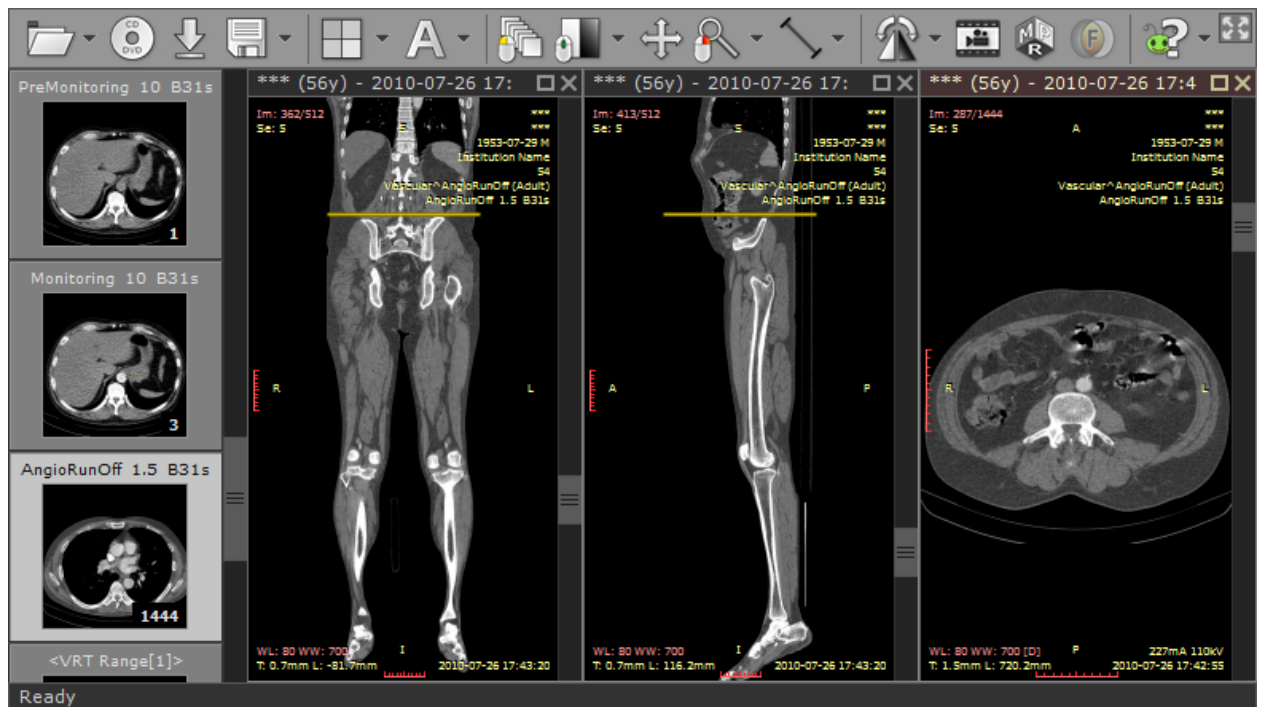


Рисунок 1.9 – Інтерфейс програми Radiant

Ця програма підтримує DICOM файли наступних досліджень:

- 1) цифрова рентгенографія та мамографія.
- 2) МРТ – магнітно-резонансна томографія.
- 3) КТ – комп'ютерна томографія.
- 4) УЗД – ультразвукове дослідження.
- 5) ЦА – цифрова ангіографія.
- 6) ПЕТ/КТ – позитронно-емісійна томографія.

Програма RadiAnt дозволяє працювати з однаковою гарною швидкістю і на комп'ютерах з оперативною пам'яттю 512 мегабайт, а також на комп'ютерах з оперативною пам'яттю 4 гігабайти та вище. Звичайно, на комп'ютерах потужніших у користувача більше можливостей для використання всього потенціалу цієї програми.

У програмі користувач може виконати такі дії:

- 1) Зміна яскравості та контрастності.

- 2) Збільшення чи зменшення об'єкта.
- 3) При оцінці КТ можливість вибору вікна візуалізації (легеневе, м'яко-тканне, кісткове).
- 4) Поворот, розворот, дзеркальне відображення сканів.
- 5) Здійснити виміри довжини, товщини, ширини, об'єму.
- 6) Вимірювання щільності тканин у одиницях Хаунсфілда при КТ дослідженні.

При використанні цієї програми є можливість зберегти DICOM зображення як відео у форматі WMV або зображення у форматі JPEG. Також можна скопіювати зображення в буфер обміну, а далі використовувати зображення у презентації або заключенні, що зручно при підготовці доповіді з презентацією та для більш інформативного висновку із зображенням. [9]

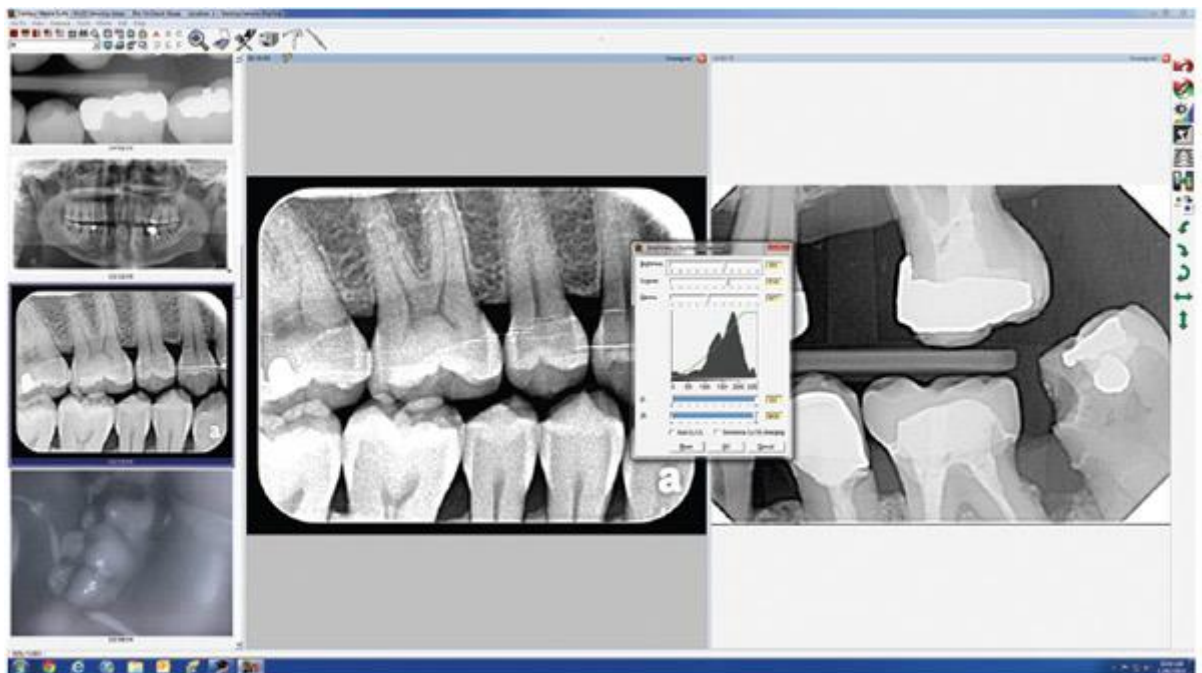


Рисунок 1.10 – Інтерфейс програми MediaSuite

Графічний редактор MediaSuite дозволяє:

- отримувати зображення з різноманітного обладнання (радіовізіографи, ортопантомографи, сканери датчиків, інтраоральні камери);

- зберігати отримані зображення в єдиній базі пацієнтів з прив'язкою до конкретного зуба, щелепи, ротової порожнини;
- імпортувати будь-які графічні файли;
- проводити обробку рентгенівських знімків і фотографій (усі основні функції програми для роботи з радіовізіографом);
- проводити вимірювання на рентгенівських знімках, у тому числі розрахунок телерентгенограм.

Основні інструменти для обробки зображень у програмі MediaSuite:

- Яскравість, контрастність та гамма-фільтр;
- Позитив/негатив;
- Насиченість, прозорість;
- Вирівнювання, тиснення, колір, різкість, згладжування та шумозаглушення;
- Виділення областей однорідної щільності тканини.

Вимірювання та коментарі.

- Можливість калібрування вимірювань;
- Вимірювання відстаней за допомогою ламаної лінії;
- Вимірювання кутів;
- Текстові нотатки з можливістю зміни кольору та шрифту тексту;
- Малювання поверх зображення;
- Відображення/приховування нотаток;
- І багато іншого:

Обрізка:

- Поворот та відображення;
- Режим "Псевдо-3D";
- Фільтр покращення різкості зображення ("нечітка маска");
- Можливість у будь-який час повернення до вихідного зображення.

Vidar Dicom Viewer – вже з назви програми зрозуміло, що цей продукт дозволить працювати зі специфічним форматом зображень, які

використовуються з медичною метою, стандартом DICOM. Однак, незважаючи на такий скромний статус переглядача включеного в ім'я програми, Dicom Viewer дозволить не тільки переглядати зображення, але також відкрити диск MPT або КТ дослідження, побудувати мультипланарну та 3D-реконструкцію.

Зокрема, даний продукт має можливості перегляду зображень, деякі функції обробки DICOM-файлів, дозволить виконати прості операції пов'язані з вимірюванням, також доступні операції пов'язані з імпортом або експортом таких зображень (у вільній версії є обмеження, пов'язані із записом досліджень на диски CD/DVD), виведення файлів на друк (є обмеження з виведенням на лазерні принтери в безкоштовній версії).

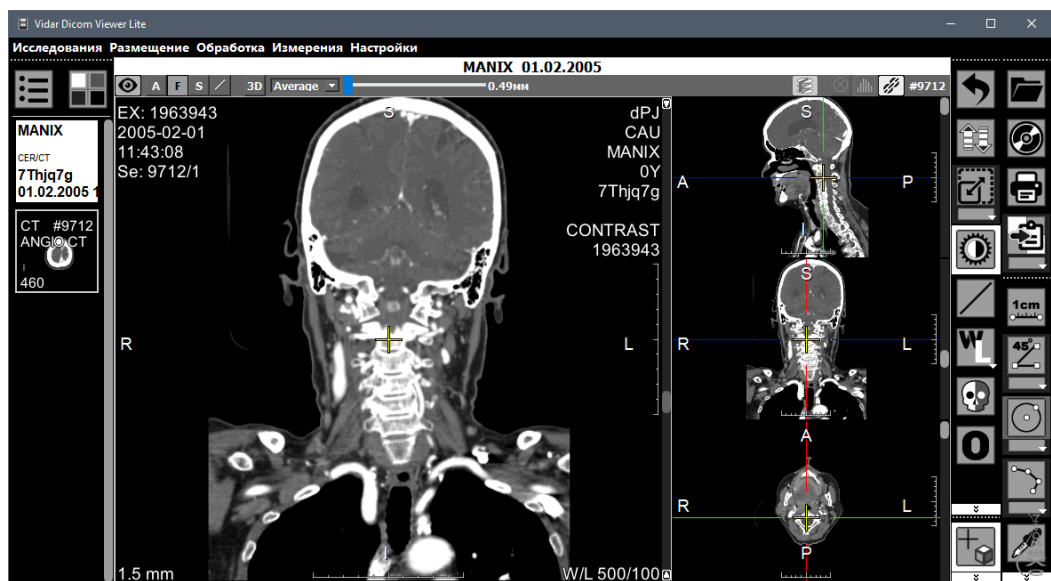


Рисунок 1.11 – Інтерфейс програми Vidar Dicom Viewer

Vidar Dicom Viewer встановлений на комп'ютері багатьох відомих медичних закладів, відомих не тільки в нашій країні. Даний програмним забезпеченням протягом багатьох років користуються багато відомих лікарів променевої діагностики та клініцисти. За заявою авторів цього чудового інструменту, він повністю здатний задовольнити вимоги до подібного ПЗ, оскільки має:

- усі необхідні функції для аналізу медичних знімків;

- МПР та 3D-реконструкції, відтворення фільмів;
- Не високі системні вимоги до комп'ютера дозволяють запуснути програму на малопотужних машинах;
- Підтримує роботу у всій лінійці операційної системи Windows;
- Інтуїтивне, зрозуміле розташування русифікованих елементів керування.

Деякі галузі медицини, де можливе використання цього програмного забезпечення: комп'ютерна томографія, ультразвукова діагностика, рентгенографія, цифрова ендоскопія, МРТ, мамографія та ангіографія. [11]

Висновки до першого розділу

В першому розділі були розглянуті методи обробки рентгенівських зображень. З'ясували, що основними методами обробки рентгенівських зображень є фільтрація. Розглянули медіаний, усереднюючий та вейвлет-фільтри. Також розглянули основні види шумів на рентгенівських знімках. Це білий шум, дробовий шум та мультипликативний шум.

Були розглянуті та проаналізовані існуючі програми для перегляду та обробки рентгенівських знімків – програми Vidar Dicom Viewer, MediaSuite та Radiant.

2. ВИБІР МОВИ ПРОГРАМУВАННЯ ДЛЯ ФІЛЬТРАЦІЇ РЕНТГЕНІВСЬКИХ ЗОБРАЖЕНЬ

2.1 Мови програмування для фільтрації рентгенівських зображень

Вибір мови програмування в даному випадку базується на можливостях обробки цифрових зображень.

Написання програми для фільтрації рентгенівських зображень можливо застосувати такі мови програмування:

- Python;
- C#;
- Matlab.

Розглянемо детально кожен мову програмування.

Python – це скриптова мова, що активно розвивається, яку використовують для вирішення великого обсягу найрізноманітніших проблем і завдань. Python стане в нагоді у створенні комп'ютерних та мобільних додатків, його застосовують у роботі з великим обсягом інформації, при розробці web-сайтів та інших різноманітних проектів, використовують у машинному навчанні. Ця мова програмування використовують великі відомі корпорації, такі як Spotify та Amazon (наприклад, для аналізу даних та створення алгоритму рекомендацій), YouTube, Instagram і навіть Walt Disney. Таким чином, Python знайшов своє місце в різних галузях – з його допомогою можна вирішити багато завдань різної складності.

Розробка мови Python розпочалася наприкінці вісімдесятих років двадцятого сторіччя. Для розподіленої операційної системи «Амоеба» знадобилася скриптова мова, що розширюється, і співробітник голландського інституту Гвідо ван Россум почав писати таку мову у вільний час. Вже в 1991 році Гвідо опублікував перший код. Назва мови, незважаючи на співзвучність з назвою сімейства неотруйних змій, походить від іншого. Розробник назвав мову на честь відомого британського гумористичного телевізійного шоу

сімдесятих – "Літаючий цирк Монті Пайтона". Серед користувачів Python часто називають просто "Пітон".

Python застосовують у різних сферах. Розробники з усього світу полюбили цю мову програмування за багато можливостей.

1. Об'єктно-орієнтоване програмування (ООП). Реалізація ООП в Пайтон хоч і специфічна порівняно з іншими об'єктно-орієнтованими мовами, але водночас непогано продумана.

2. Узагальнене та функціональне програмування.

3. Модулі та пакети. Програмне забезпечення (ПЗ) на Пітоні оформляється у вигляді модулів, які можуть бути зібрані в пакети.

4. Інтроспекція. Це можливість запросити тип та структуру об'єкта під час виконання програми. У Пітоні для будь-якого об'єкта можна дізнатися всю інформацію про його внутрішню структуру.

5. Обробка винятків, ітератори та генератори.

6. Управління контекстом виконання.

7. Декоратори, регулярні вирази

Переваги Python наступні:

1. Мова характеризується логічним синтаксисом, унаслідок чого вихідний код програм, написаних «на пітоні», легко читається та сприймається.

2. Ще одна з переваг цієї мови програмування – її умовна легкість. Він вважається найбільш підходящим для фахівців-початківців: розробляти нескладні програми можна навчитися вже через пару-трійку днів вивчення.

3. Велика інтернет-спільнота. Якщо розробник стикається з питаннями та труднощами, він завжди може запитати поради у колег, що значно прискорює вирішення проблем.

4. Гнучкість та масштабованість. Пайтон дозволяє розробникам адаптувати високорівневу логіку програми, що дозволяє легко розширювати складні програми у міру потреби.

5. Розробка «на пітоні» йде швидше, ніж на більшості інших мов.

6. Python є мовою програмування, що інтерпретується. Це означає, що до запуску він є звичайним текстовим файлом. Відповідно програмувати можна майже на всіх платформах.

Недоліки Python наступні:

1. Швидкість роботи. Високопродуктивні проекти на чистому Пайтоні написати буде важко. Для цього потрібно вдаватися до інших мов.

2. Безпека, яка забезпечується моделлю пам'яті мови Python, зводить нанівець більшість можливих процесорних оптимізацій.

Це два головних мінуси мови програмування Python: його мала швидкодія та недостатні можливості статичного аналізу коду.

Пітон можна використовувати у більшості сфер, приклади яких наведено нижче:

1) Програмні програми. За допомогою цієї мови програмування можна розробляти програми на комп'ютері.

2) Мобільні додатки. Мобільна розробка «на пітоні» має попит у всьому світі – для програмування серверної частини таких додатків найчастіше вибирають Python.

3) Ігри. Багато комп'ютерних ігор цілком або частково розроблено на Пайтон. Наприклад, він використовувався під час створення популярної гри World of Tanks.

4) Вбудовані системи. На Пітон можна створювати вбудовані системи для різних пристроїв. Наприклад, для керування банкоматами.

5) Скрипти. На Python можна написати плагіни та скрипти до вже реалізованих програм для створення нових модулів та автоматизації процесів.

6) Машинне навчання (Machine Learning). З його допомогою пишуть алгоритми для Machine Learning програм, а також багато аналітичних програм.

7) Тестування. Python активно використовується для автоматизації тестування. [12]

Обробка зображень у Python здійснюється завдяки таких бібліотек:

1) Scikit-image – це Python-пакет з відкритим кодом, який працює з масивами NumPy. Він реалізує алгоритми та утиліти для використання в дослідницьких, освітніх та промислових додатках. Це дуже проста та зрозуміла бібліотека навіть для новачків в екосистемі Python. Ця бібліотека містить високоякісний та рецензований код, написаний активною спільнотою добровольців. Використовується для фільтрації та виділення контурів зображень.

NumPy – це одна з основних Python-бібліотек із підтримкою масивів. Зображення є стандартним масивом NumPy, що містить пікселі точок даних. Таким чином, під час виконання основних NumPy-операцій (зрізи, маски, вибагливе індексування) можна змінювати піксельні значення зображення.

Використовується для маскуванню зображень.

2) SciPy – це такий самий важливий науковий модуль у Python, як і NumPy. Він підходить для вирішення основних завдань з обробки та іншої роботи із зображеннями. Зокрема, у підмодулі `scipy.ndimage` доступні функції, що працюють у n-мірних масивах NumPy. Поточний пакет включає функції для лінійної та нелінійної фільтрації, бінарної морфології, інтерполяції B-сплайнами та вимірювань об'єктів. Використовується для завдання шумів на зображенні.

3) PIL (Python Imaging Library) – це безкоштовна Python-бібліотека для відкриття, роботи та збереження різних форматів зображень. На жаль, її розробка остаточно зупинилася, а останнє оновлення вийшло в 2009. На щастя, є Pillow – форк PIL, що активно розвивається, з простою установкою. Він працює на всіх основних операційних системах та підтримує Python 3. Бібліотека містить базовий функціонал для обробки зображень, включаючи точкові операції, фільтри з набором вбудованих ядер згортки та перетворення кольорового простору. Використовується для покращення зображень та їх фільтрації.

4) OpenCV (Open Source Computer Vision Library) – одна з найпопулярніших бібліотек для програм з комп'ютерного зору. OpenCV-

Python – це Python-версія інтерфейсу для OpenCV. Наявність коду на C/C++ у бекенді гарантує швидкість бібліотеки, а Python-обгортка у фронтенді забезпечує легкість налаштування та розгортання. Завдяки цьому OpenCV-Python є чудовим рішенням для високонавантажених обчислювальних програм з комп'ютерного зору.

Використовується для накладання та поєднання двох зображень.

5) SimpleCV – це ще один фреймворк з відкритим кодом для створення програм комп'ютерного зору. З ним у вас з'являється доступ до кількох потужних бібліотек комп'ютерного зору (наприклад OpenCV) без необхідності вивчення глибини кольору, файлових форматів, колірних просторів. Крива навчання куди менше, ніж у OpenCV, і, як мовиться в їхньому слогані, «комп'ютерний зір стає простіше». Парочка плюсів SimpleCV:

Функціональна сумісність із камерами, відеофайлами, зображеннями та відеопотоками.

Використовується для виділення контурів, перетворення двоградаційне зображення та створення 3D моделі.

6) Mahotas також є Python-бібліотекою для комп'ютерного зору та обробки зображень. Вона містить стандартні функції з обробки зображень (фільтри та морфологічні операції), а також сучасні можливості комп'ютерного зору для обчислення ознак (виявлення особливих точок та локальні дескриптори). Швидкість розробки забезпечується Python-інтерфейсом, а плюсом для швидкості є алгоритми на C++. Mahotas – це швидка бібліотека з мінімальним кодом і залежностями.

Використовується для фільтрації та морфологічних операцій з зображеннями.

7) ITK або Insight Segmentation and Registration Toolkit – це крос-платформна система з відкритим кодом, що надає розширений набір інструментів для аналізу зображень. Сюди відноситься і SimpleITK – спрощений шар, "надбудований" поверх ITK. Цей шар полегшує роботу з бібліотекою при швидкому прототипуванні, навчанні та мовах, що

інтерпретуються. SimpleITK – це набір інструментів для аналізу зображень з великою кількістю компонентів, що підтримують загальну фільтрацію, сегментацію та реєстрацію зображень. Сам SimpleITK написаний на C++, але доступний багатьом мовам програмування, включаючи Python.

8) Jupyter Notebook показує використання SimpleITK в освітніх та дослідних цілях. Він також демонструє можливості SimpleITK щодо інтерактивного аналізу зображень з використанням мов програмування Python та R.

9) Pymagick – обгортка на базі Python для бібліотеки GraphicsMagick. Систему GraphicsMagick іноді називають швейцарським ножом для обробки зображень. Вона пропонує колекцію ефективних інструментів та бібліотек, що підтримують читання, запис та операції із зображеннями у більш ніж 88 основних форматах, включаючи DPX, GIF, JPEG, JPEG-2000, PNG, PDF, PNM та TIFF.

Pymagick присвячений цілий репозиторій Github. Там ви знайдете інструкції з встановлення та основні вимоги. Також є докладний посібник користувача.

Використовується для виділення контурів та масштабування зображень.

10) PyCairo є набір прив'язок Python-коду для графічної бібліотеки Cairo. Cairo – це 2D-бібліотека для відображення векторної графіки. Векторна графіка цікава тим, що не втрачає чіткості при зміні розмірів або трансформації. PyCairo – це набір прив'язок для Cairo, за допомогою яких можна викликати команди Cairo з Python.

Детальна інформація щодо встановлення та роботи доступна в GitHub-репозиторії PyCairo. Є ще вступний посібник з коротким описом PyCairo [13,14].

Також обробку зображень можна виконати, написавши програму на C#. C# – мова програмування, що поєднує об'єктно-орієнтовані та контекстно-орієнтовані концепції. Розроблений в 1998-2001 роках групою інженерів під керівництвом Андерса Хейлсберга в компанії Microsoft як основна мова

розробки додатків для платформи Microsoft .NET. Компілятор з C# входить до стандартної установки самої .NET, тому програми на ньому можна створювати і компілювати навіть без інструментальних засобів на кшталт Visual Studio.

C# відноситься до сім'ї мов з C-подібним синтаксисом, їх синтаксис найбільш близький до C++ і Java. Мова має строгу статичну типізацію, підтримує поліморфізм, навантаження операторів, покажчики на функції-члени класів, атрибути, події, властивості, винятки, коментарі у форматі XML. Перейнявши багато від своїх попередників – мов C++, Delphi, Modula і Smalltalk – C#, спираючись на практику їх використання, виключає деякі моделі, що зарекомендували себе як проблематичні при розробці програмних систем: так C# не підтримує множинне успадкування класів (на відміну від C++) або виведення типів (на відміну Haskell) [15,16].

C# – це об'єктно-і компонентно-орієнтована мова програмування. C# надає мовні конструкції безпосередньої підтримки такої концепції роботи. Завдяки цьому C# підходить для створення та застосування програмних компонентів. З моменту створення мова C# збагатилася функціями для підтримки нових робочих навантажень та сучасними рекомендаціями щодо розробки ПЗ. C# – це об'єктно-орієнтована мова. Ви визначаєте типи та їх поведінку.

Програми C# виконуються в .NET, віртуальній системі виконання, що викликає загальномовне середовище виконання (CLR) та набір бібліотек класів. Середовище CLR – це реалізація загальномовної інфраструктури мови (CLI), що є міжнародним стандартом від корпорації Майкрософт. CLI є основою для створення середовищ виконання та розробки, в яких мови та бібліотеки прозоро працюють одна з одною.

Вихідний код, написаний мовою C#, компілюється в проміжну мову (IL), який відповідає специфікаціям CLI. Код мовою IL та ресурси, у тому числі растрові зображення та рядки, зберігаються у збірці, зазвичай з розширенням .dll. Складання містить маніфест з інформацією про типи, версії, мову та регіональні параметри для цієї збірки.

Під час виконання програми C# збірка завантажується у середовище CLR. Середовище CLR виконує JIT-компіляцію з коду IL в інструкції машинної мови. Середовище CLR також виконує інші операції, наприклад, автоматичне складання сміття, обробку виключень та управління ресурсами. Код, виконуваний серед CLR, іноді називається керованим кодом. "Некерований код" перетворюється на машинну мову, призначену для конкретної платформи.

Забезпечення взаємодії між мовами є основною особливістю .NET. Код IL, створений компілятором C# відповідає специфікації загальних типів (CTS). Код IL, створений з коду C#, може взаємодіяти з кодом, створеним з версій .NET для мов F#, Visual Basic, C++. Існує більше 20 інших мов, сумісних із CTS. Одна збірка може містити кілька модулів, написаних різними мовами .NET, і всі типи можуть посилатися один на одного, якби вони були написані однією мовою.

Крім служб часу виконання .NET також включає розширені бібліотеки. Ці бібліотеки підтримують багато різних робочих навантажень. Вони впорядковані за просторами імен, які надають різні корисні можливості: від операцій файлового введення та виведення до керування рядками та синтаксичного аналізу XML, від платформ веб-додатків до елементів керування Windows Forms. Зазвичай програму C# активно використовують бібліотеку класів .NET для вирішення типових завдань.[17]

Бібліотека зображень Visual Studio містить зображення програм, які відображаються в Microsoft Visual Studio, Microsoft Windows, Office та іншому програмному забезпеченні Microsoft. Набір із понад 1000 зображень можна використовувати для створення програм, зовнішній вигляд яких поєднується з програмним забезпеченням Microsoft.

Бібліотека зображень складається з п'яти категорій: загальні елементи, дії, нотатки, значки та об'єкти. Для типів спільних елементів та піктограм у бібліотеці також включені файли відомостей у форматі PDF. Вони містять інформацію про правильне використання цих зображень у програмах. [18]

Список бібліотек C# для роботи з зображенням наступний:

- 1) ImageResizer – додайте команди у зображення URLs, щоб отримати змінені версії за мілісекунди. Зміна розміру, редагування тощо. зображень у режимі реального часу.
- 2) ImageProcessor – бібліотека з відкритим вихідним кодом .NET для керування зображеннями on-the-fly.
- 3) DynamicImage – високопродуктивна бібліотека обробки зображень з відкритим вихідним кодом для ASP.NET.
- 4) MetadataExtractor – витягує Exif, IPTC, XMP, ICC та інші метадані із файлів зображень.
- 5) Emgu CV – крос-платформна оболонка .NET для бібліотеки OpenCV.
- 6) DotImaging – мінімалістична .NET портативна платформа для візуалізації зображень
- 7) Magick.NET – .NET оболонка для бібліотеки ImageMagick.
- 8) OpenCvSharp – крос-платформна оболонка OpenCV для фреймворку .NET. [19]

Основна бібліотека для роботи з зображенням в C# ImageMagick. ImageMagick – це потужна бібліотека для роботи із зображеннями, яка підтримує понад 100 основних форматів файлів (не включаючи підформати). З Magick.NET можна використовувати ImageMagick у своєму додатку C#/VB.NET/.NET Core без встановлення ImageMagick на свій сервер або комп'ютер.

Бібліотека Magick.NET має відмінність у x86 або x64 розрядності.

У бібліотеці Magick.NET велика кількість методів для роботи із зображеннями розглянемо деякі з них [20]:

- MedianFilter – медіанний фільтр;
- Normalize – нормалізувати;
- ReduceNoise – зменшити рівень шуму;
- SelectiveBlur – вибіркове розмиття;
- Sharpen – різкість;

- Threshold – поріг;
- UnsharpMask – нерізка маска;
- WaveletDenoise – вейвлет Денуаз;
- Monochrome – моно хром.

Основним недоліком використання C# та Python для обробки зображень є складність в пошуках та встановленні бібліотек для роботи з зображеннями. Тому пропонується розглянути ще одну мову для написання програми для обробки зображень, а саме MATLAB.

MATLAB – це високорівнева мова технічних розрахунків, інтерактивне середовище розробки алгоритмів та сучасний інструмент аналізу даних. MATLAB у порівнянні з традиційними мовами програмування (C/C++, Java, Pascal, FORTRAN) дозволяє скоротити час вирішення типових завдань і значно спрощує розробку нових алгоритмів. MATLAB є основою всього сімейства продуктів MathWorks і є головним інструментом для вирішення широкого спектру наукових та прикладних завдань, в таких галузях як: моделювання об'єктів та розробка систем управління, проектування комунікаційних систем, обробка сигналів та зображень, вимірювання сигналів та тестування, фінансове моделювання, обчислювальна біологія та ін. Ядро MATLAB дозволяє максимально просто працювати з матрицями реальних, комплексних та аналітичних типів даних та зі структурами даних та таблицями пошуку. MATLAB містить вбудовані функції лінійної алгебри (LAPACK, BLAS), швидкого перетворення Фур'є (FFTW), функції для роботи з поліномами, функції базової статистики та чисельного розв'язання диференціальних рівнянь; розширені математичні бібліотеки для Intel MKL. Всі вбудовані функції ядра MATLAB розроблені та оптимізовані фахівцями та працюють швидше або так само, як їхній еквівалент на C/C++.

Мова MATLAB є високорівневою мовою програмування, що включає базовані на матрицях структури даних, широкий спектр функцій, інтегроване середовище розробки, об'єктно-орієнтовані можливості та інтерфейси до програм, написаних іншими мовами програмування. Програми, написані на

MATLAB, бувають двох типів – функції та скрипти. Функції мають вхідні та вихідні аргументи, а також власний робочий простір для зберігання проміжних результатів обчислень та змінних. Скрипти використовують загальний робочий простір. Як скрипти, і функції не інтерпретуються в машинний код і зберігаються як текстових файлів. Існує також можливість зберігати так звані *pre-parsed* програми - функції та скрипти, оброблені у вигляді, зручний для машинного виконання. У випадку такі програми виконуються швидше звичайних. Основною особливістю мови MATLAB є його широкі можливості для роботи з матрицями, які творці мови висловили в гасла думай векторно (англ. *Think vectorized*).

MATLAB надає зручні засоби розробки алгоритмів, включаючи високорівневі з використанням концепцій об'єктно-орієнтованого програмування. У ньому є всі необхідні засоби інтегрованого середовища розробки, включаючи налагоджувач та профайлер. Функції до роботи з цілими типами даних полегшують створення алгоритмів для мікроконтролерів та інших додатків, де це необхідно.

У складі пакету MATLAB є велика кількість функцій для побудови графіків, у тому числі тривимірних, візуального аналізу даних та створення анімованих роликів. Вбудоване середовище розробки дозволяє створювати графічні інтерфейси користувача з різними елементами керування, такими як кнопки, поля введення та інші. За допомогою компонента MATLAB Compiler ці графічні інтерфейси можуть бути перетворені на самостійні програми.

Пакет MATLAB включає різні інтерфейси для отримання доступу до зовнішніх підпрограм, написаних іншими мовами програмування, даними, клієнтами та серверами, що спілкуються через технології Component Object Model або Dynamic Data Exchange, а також периферійним пристроям, які взаємодіють безпосередньо з MATLAB. Багато з цих можливостей відомі під назвою MATLAB API.

Пакет MATLAB містить функції, які дозволяють йому отримувати доступ до інших програм середовища Windows, так само як і цим програмам

отримувати доступ до даних MATLAB, за допомогою технології динамічного обміну даними (DDE). Кожна програма, яка може бути DDE-сервером, має своє унікальне ідентифікаційне ім'я. Для MATLAB це ім'я – Matlab.

Інтерфейс MATLAB, що відноситься до загальних DLL дозволяє викликати функції, що знаходяться в звичайних бібліотеках, що динамічно підключаються, прямо з MATLAB. Ці функції повинні мати C-інтерфейс. Крім того, у MATLAB є можливість отримати доступ до його вбудованих функцій через C-інтерфейс, що дозволяє використовувати функції пакету у зовнішніх додатках, написаних на C. Ця технологія в MATLAB називається C Engine.

Для MATLAB є можливість створювати спеціальні набори інструментів (toolbox), що розширюють його функціональність. Набори інструментів є колекцією функцій, написаних мовою MATLAB для вирішення певного класу завдань. Компанія Mathworks постачає набори інструментів, які використовуються в багатьох галузях, включаючи наступні: Цифрова обробка сигналів, зображень і даних: зображень, проектування цифрових фільтрів та систем зв'язку.[21]

2.2 Вибір мови програмування

Розглянувши три мови програмування C#, Python та MATLAB прийшли до таких висновків: перевагами використання мов програмування є C# та Python швидкість роботи. Основним недоліком використання C# та Python для обробки зображень є складність в пошуках та встановленні бібліотек для роботи з зображеннями. В Matlab бібліотеки для обробки зображень встановлюються разом з самою програмою. Всі вбудовані функції ядра MATLAB розроблені та оптимізовані фахівцями та працюють швидше або так само, як їхній еквівалент на C/C++ або Python. До переваг MATLAB також можна віднести швидкість доступу, все кришталево чисте, імена функцій добре вибрані, достатня кількість потрібних функцій для роботи з зображеннями, відслідкування помилок в коді та швидке їх від лагодження. З python завжди потрібно працювати з google. Тому для написання програми для дослідження фільтрації на рентгенівських зображеннях пропонується обрати Matlab. [22]

Цифрова обробка сигналів, зображень і даних: Signal Processing Toolbox (з'явився в 1987 [12]), DSP System Toolbox, Image Processing Toolbox (з'явився в 1993), Wavelet Toolbox, Communications System Toolbox – набори функцій і об'єктів, дозволяють вирішувати широкий спектр завдань обробки сигналів, зображень, проектування цифрових фільтрів та систем зв'язку.

Можливості MATLAB для роботи з зображеннями:

- Імпорт зображень із різних форматів;
- Передобробка та фільтрація зображень;
- Сегментація зображення, виділення об'єктів на зображенні;
- Пошук ключових точок на зображенні;
- Обробка супутникових знімків, реєстрація зображень;
- Обробка зображень великого розміру;
- Прискорення обчислень, використання GPU ядер.

Image Processing Toolbox – надає повний набір стандартних алгоритмів та програм для обробки зображень, аналізу, візуалізації та розробки нових алгоритмів. Ви можете виконувати сегментацію зображень, зменшувати шум,

геометричні перетворення, виконувати реєстрацію зображень та обробляти тривимірні зображення.

Програми Image Processing Toolbox дозволяють автоматизувати загальні робочі процеси обробки зображень. В інтерактивному режимі сегментувати зображення, порівнювати методи реєстрації зображень, пакетно обробляти великі набори даних. Функції візуалізації та програми дозволяють вам переглядати зображення, відео 3D набори зображень, налаштовувати контрастність, будувати гістограми, та керувати областями інтересу (ROI).

Можна прискорювати свої алгоритми, запускаючи їх на багатоядерних процесорах та GPU. Більшість функцій з Image Processing Toolbox підтримують генерацію C/C++ коду для десктопних систем, що вбудовуються.

Основні засоби бібліотеки Image Processing Toolbox

- Побудова фільтрів, фільтрація та відновлення зображень;
- Збільшення зображень;
- Аналіз та статистична обробка зображень;
- Виділення областей інтересів, геометричні та морфологічні операції;
- Маніпуляції з квітами;
- Двовимірні перетворення;
- Блок обробки;
- Засіб візуалізації;
- Запис/читання графічних файлів. [23]

Для фільтрації рентгенівських знімків будуть використовуватись фільтри зображень: медіаний, усереднюючий та вейвлет.

За вейвлет фільтри в Matlab відповідає бібліотека Wavelet Toolbox.

Wavelet Toolbox надає програми та функції для шумоподавлення у сигналах та зображення. Видалення шуму за допомогою вейвлетів та пакетів вейвлетів дозволяє зберігати ознаки у ваших даних, які часто видаляються або згладжуються іншими методами шумоподавлення.

Програма Wavelet Signal Denoiser дозволяє візуалізувати та автоматично видаляти шум з тимчасових рядів даних.

Wavelet Toolbox підтримує різні стратегії вибору порогового значення, які можна застосувати до даних та використовувати для порівняння результатів. Шум у сигналі не завжди однорідний у часі, тому можна застосовувати залежні від інтервалу порогові значення для шумоподавлення даних із непостійною дисперсією.

Можна видаляти шум із наборів сигналів за допомогою вейвлетів, використовуючи кореляції між окремими сигналами. Також можна кластеризувати групи сигналів шляхом фільтрування несуттєвих деталей з використанням розріджених вейвлет-уявлень.

Також можливе дослідження за допомогою вейвлет-меню wavemenu Wavelet Toolbox GUI (Graphical User Interface) досліджувати за допомогою вейвлет-меню wavemenu Wavelet Toolbox GUI (Graphical User Interface) досліджувати за допомогою вейвлет-меню wavemenu Wavelet Toolbox GUI (Graphical User Interface), інтерфейс головного вікна (представлений на рис.2.1).

Фільтрація зображень за допомогою вейвлет-фільтра в wavemenu (представлено на рис.2.2).

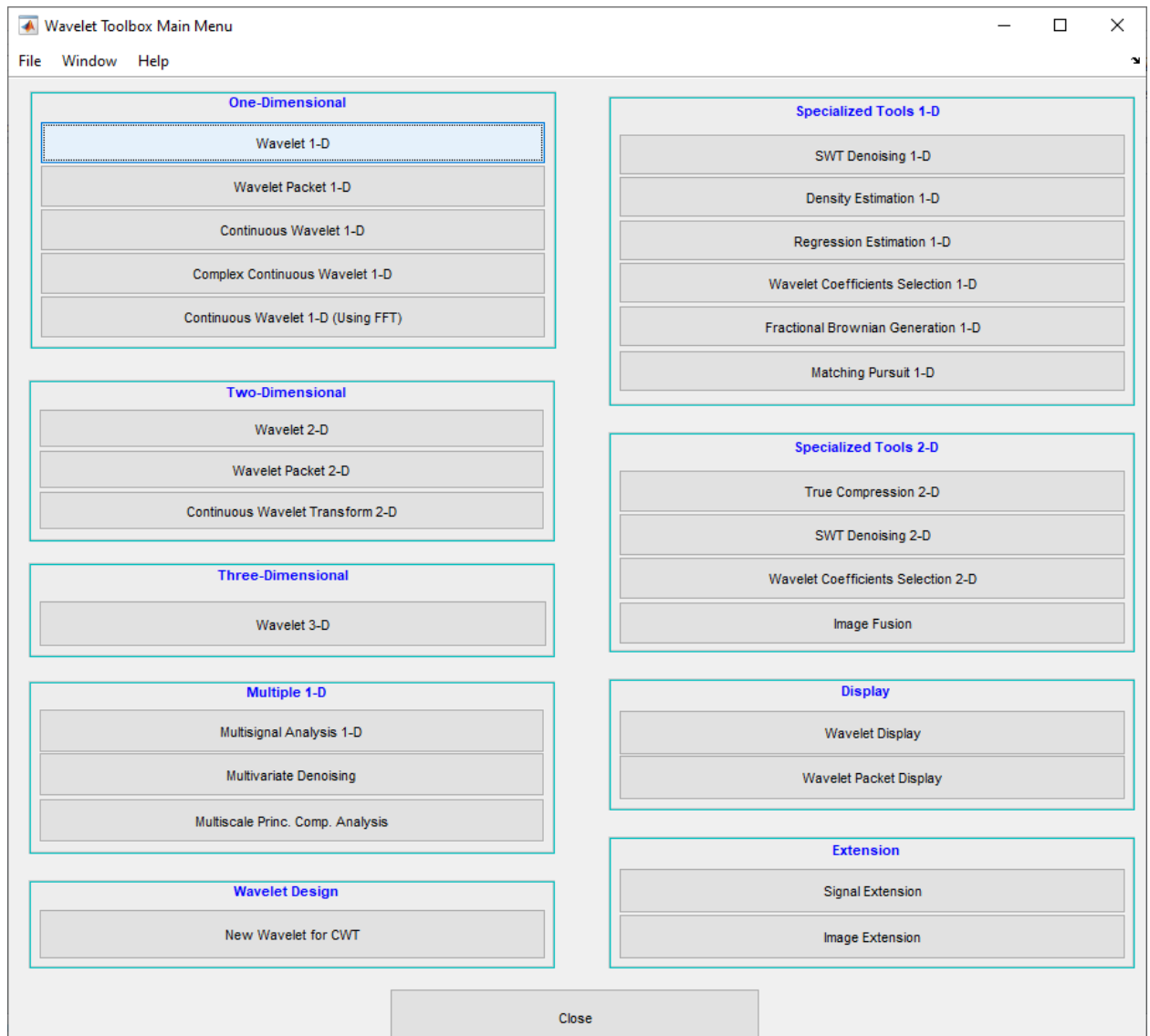


Рисунок 2.1 – Інтерфейс Wavelet Toolbox Menu

За допомогою Wavelet Toolbox Menu можна стискати та фільтрувати зображення різними вейвлет фільтрами, а також знаходити різницю та помилки при відновленому зображенні.

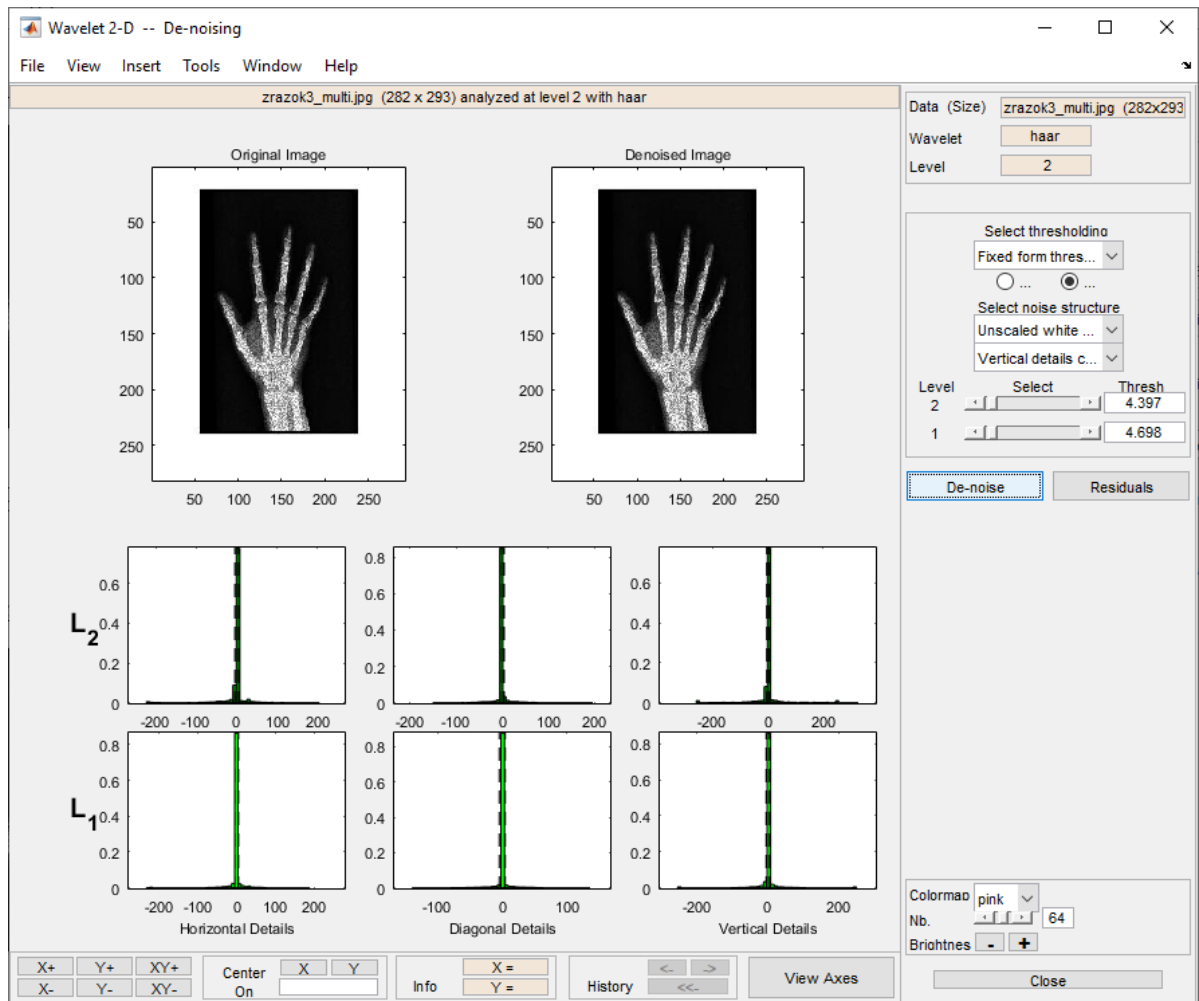


Рисунок 2.2 – Фільтрація рентгенівського знімка в Wavelet Toolbox Menu

Висновки до другого розділу

В другому розділі були проаналізовані три мови програмування: C#, Python та MATLAB. Прийшли до висновків: переваги використання мов програмування C# та Python є швидкість роботи. Основним недоліком використання C# та Python для обробки зображень є складність в пошуках та встановленні бібліотек для роботи з зображеннями. Тому для написання програми для дослідження фільтрації на рентгенівських зображеннях пропонується обрати Matlab. В Matlab бібліотеки для обробки зображень встановлюються разом з самою програмою, також можна віднести швидкість доступу завдяки завдання добре вибраних імен функцій, достатня кількість потрібних функцій для роботи з зображеннями, від слідкування помилок в коді та швидке їх від лагодження.

3.ДОСЛІДЖЕННЯ МЕТОДІВ ФІЛЬТРАЦІЇ РЕНТГЕНІВСЬКИХ ЗОБРАЖЕНЬ

3.1 Аналіз методів фільтрації рентгенівських зображень

Для рентгенівських зображеннях характерні мультиплікативний, білий та дробові шуми. Для усунення цих шумів вириховуються фільтрація.

Медіанний фільтр більш ефективно видаляє імпульсні завади, ніж шуми типу “білого” шуму з нормальним законом розподілу. Медіанний фільтр слід вважати евристичним методом обробки відеозображень.

Усереднюючий і медіанний фільтр є ефективними тільки по відношенню до дробового шуму, який проявляється на відеозображенні у вигляді імпульсних завад, тобто чорних і білих точок. Якщо викривлення на відеозображенні мають інший вид, наприклад, це є “білий” шум з нормальним законом розподілу, то більш ефективним вважається спектральний підхід до фільтрації відеозображень.

Для усунення інших шумів та завад на зображенні пропонується використовувати вейвлет-фільтрацію.

В Matlab функції задані білого шумів наступні:

```
Tsh='gaussian';           % ТИП ШУМУ НА ЗОБРАЖЕННІ
% 'gaussian' - "БІЛИЙ" ШУМ З НОРМАЛЬНИМ РОЗПОДІЛОМ
M=0;                     % СЕРЕДНЄ ЗНАЧЕННЯ ШУМУ
V=0.03;                  % ДИСПЕРСІЯ ШУМУ
```

Змоделюємо додавання білого шуму до зображення з дисперсією 0,01 (рис.3.2). Візьмемо для прикладу перелому променевої кістки на рентген знімку (рис.3.1).

В Matlab функції задані дробового шумів наступні:

```
% Tsh='salt & pepper';    % ТИП ШУМУ НА ЗОБРАЖЕННІ
% 'salt & pepper' - ШУМ У ВИГЛЯДІ БІЛИХ І ЧОРНИХ ТОЧОК
% D=0.05;                 % ЩІЛЬНІСТЬ ШУМУ НА ЗОБРАЖЕННІ
```

Змоделюємо додавання дробового шуму до зображення з щільністю 0,01 (рис.3.3).

В Matlab функції задані мультиплікативного шумів наступні:

```
% Tsh='speckle';           % ТИП ШУМУ НА ЗОБРАЖЕННІ
% 'speckle' - МУЛЬТИПЛІКАТИВНИЙ ШУМ
% V=0.01;                 % ДИСПЕРСІЯ ШУМУ
```

Змоделюємо додавання мультиплікативного шуму до зображення з дисперсією 0,05 (рис.3.4).



Рисунок 3.1– Зображення рентген знімка перелому променевої кістки



Рисунок 3.2 – Зображення рентген знімка перелому променевої кістки з білим шумом з дисперсією 0,01



Рисунок 3.3 – Зображення рентген знімка перелому променевої кістки з білим шумом з щільністю 0,01



Рисунок 3.4 – Зображення рентген знімка перелому променевої кістки з білим шумом з дисперсією 0,05

Для фільтрації в Matlab будемо використовувати:

- Медіаний фільтр;
- Усереднюючий фільтр;
- Вейвлет-фільтр.

Медіаний фільтр в Matlab задається наступним чином:

```
МЕДІАННИЙ ФІЛЬТР
% РОЗМІР МАТРИЦІ ДЛЯ ВИЗНАЧЕННЯ СУСІДНИХ ТОЧОК
% Mfilter=[3 3];
% RestoreImage=medfilt2(NoiseImage,Mfilter);
```

Усереднюючий фільтр в Matlab задається наступним чином:

```
Tfilter='average'; % ТИП ФІЛЬТРА
                  % 'average' - УСЕРЕДНЮЮЧИЙ ФІЛЬТР
Hsize=3;          % РОЗМІР КВАДРАТНОЇ МАСКИ ФІЛЬТРА
Filter=fspecial(Tfilter,Hsize); % СТВОРЕННЯ МАСКИ ФІЛЬТРА
```

Вейвлет-перетворення для усунення шуму в Matlab задається наступним чином:

```
[c,s]=wavedec2(t1,2,'sym4');  
a1=wrcoef2('a',c,s,'sym4');
```

Враховуючи те, що потрібно завантажити зображення, відфільтрувати оцінити фільтр, вивести зображення та похибки. Так як зображення отримані якісні, тому змодельюємо шуми на зображенні, які були розглянуті вище рис.3.2-3.4(білий, сіль та перець, мультиплікативний). Алгоритм роботи програми (представлений на рис.3.5).

Алгоритм роботи наступний: початок відповідає за роботу програми на початку; зчитування початкового зображення команда для зчитування вхідного зображення з файлу; формування шуму функція для завдання щільності або дисперсії та виду шуму; формування фільтра функція для завдання маски та виду фільтра; перетворення початкового зображення в напівтонове функція для перетворення початкового зображення в напівотонове; додавання шуму до зображення функція, яка додає потрібний шум до напівтонового зображення; фільтрування зображення функція, яка виконує фільтрування згідно заданого фільтра; обчислення похибок обраховує середнє значення похибки та максимальне значення похибки. Похибка розраховується як різниця між початковим зображенням та відфільтрованим.

3.2 Дослідження методів фільтрації рентгенівських зображень

Зробимо дослідження методів фільтрації рентгенівських знімків різними методами та оберемо оптимальний метод.

Для зображення рентген знімка перелому променевої кістки (зразок 1) з білим шумом з дисперсією 0,01 промодельюємо медіанний фільтр рис.3.6 , усереднюючий фільтр з маскою фільтра, що дорівнює 3 (рис.3.7), вейвлет-фільтр (рис.3.8).



Рисунок 3.5 – Алгоритм роботи програми оцінки метода фільтрування



Рисунок 3.6 – Медіаний фільтр з маскою 3 (зразок 1)



Рисунок 3.7 – Усереднюючий фільтр з маскою 3 (зразок 1)



Рисунок 3.8 – Вейвлет-фільтр (зразок 1)

Для зображення рентген знімка перелому плеча (зразок 2) з білим шумом з дисперсією 0,01 (рис.3.9) промодельюємо медіанний фільтр рис.3.10 , усереднюючий фільтр з маскою фільтра, що дорівнює 3 (рис.3.11), вейвлет-фільтр (рис.3.12).



Рисунок 3.9 – Рентген знімка перелому плеча (зразок 2) з білим шумом



Рисунок 3.10 – Медіаний фільтр з маскою 3 (зразок 2)



Рисунок 3.11 – Усереднюючий фільтр з маскою 3 (зразок 2)



Рисунок 3.12 – Вейвлет-фільтр (зразок 2)

Для зображення рентген знімка пальців рук (зразок 3) з білим шумом з дисперсією 0,01 (рис.3.13) промодельюємо медіанний фільтр рис.3.14, усереднюючий фільтр з маскою фільтра, що дорівнює 3 (рис.3.11), вейвлет-фільтр (рис.3.15).



Рисунок 3.13 – Рентген знімка пальців рук (зразок 3) з білим шумом



Рисунок 3.14 – Медіаний фільтр з маскою 3 (зразок 3)



Рисунок 3.11 – Усереднюючий фільтр з маскою 3 (зразок 3)



Рисунок 3.12 – Вейвлет-фільтр (зразок 3)

Значення максимальної похибки відновлення зображення для трьох зразків з білим шумом з дисперсією 0,01 для медіанного фільтра з маскою фільтра 3, усереднюючого фільтра з маскою фільтра 3 та вейвлет-фільтр (представлені в табл.3.1) .

Таблиця 3.1

Зображення	Максимальна похибка відновлення зображення за допомогою фільтра/д.т		
	Усереднюючий фільтр	Медіаний	Вейвлет
Зразок 1	139.00	122.00	77
Зразок 2	128.00	232.00	61.00
Зразок 3	90.00	107.00	99.00

Значення максимальної похибки відновлення зображення для трьох зразків з дробовим шумом («сіль та перець») з щільністю шуму 0,05 для медіанного фільтра з маскою фільтра 3, усереднюючого фільтра з маскою фільтра 3 та вейвлет-фільтр (представлені в табл. 3.2).

Таблиця 3.2

Зображення	Максимальна похибка відновлення зображення за допомогою фільтра/д.т		
	Усереднюючий фільтр	Медіаний	Вейвлет
Зразок 1	136.00	243.00	68.00
Зразок 2	129.00	235.00	79.00
Зразок 3	110.00	136.00	104.00

Значення максимальної похибки відновлення зображення для трьох зразків з мультиплікативним шумом з дисперсією 0,02 для медіанного фільтра з маскою фільтра 3, усереднюючого фільтра з маскою фільтра 3 та вейвлет-фільтр (представлені в табл. 3.3).

Таблиця 3.3

Зображення	Максимальна похибка відновлення зображення за допомогою фільтра/д.т		
	Усереднюючий фільтр	Медіаний	Вейвлет
Зразок 1	136.000	243.00	80.00
Зразок 2	120	232	71.00
Зразок 3	107.00	92.00	85.00

Значення максимальної похибки відновлення зображення для трьох зразків з різними шумами згідно таблиць 3.1-3.3 показують, що найменша похибка зображення за допомогою вейвлет-фільтра. Тому виходячи з досліджених даних пропонується в подальшому для фільтрації рентгенівських зображень використовувати вейвлет-фільтр [24].

Висновки до третього розділу

В третьому розділі було розглянута програмна реалізація в Matlab існуючих методів фільтрації. Був розроблений алгоритм та програма роботи для дослідження та аналізу методів фільтрації рентгенівських зображень.

Були промодельовані три зразка зображення з різними шумами та усунення цих шумів завдяки трьох різних фільтрів. В якості фільтрів було використано: усереднюючий фільтр, медіани фільтра та вейвлет-фільтр.

Оцінка фільтрації проводилася за максимальною похибкою відновлення зображення. Дослідження за максимальною похибкою показало, що найкраще для рентгенівських знімків використовувати вейвлет-фільтрацію.

ВИСНОВКИ

В кваліфікаційній магістерській роботі згідно теми та мети було проведено дослідження найкращого методу фільтрації рентгенівських зображень для подальшої інтерпретації знімка лікарем-рентгенологом для того, щоб встановити діагноз.

В кваліфікаційній роботі було зроблено наступне:

1. Проаналізовані існуючі види шумів на рентгенівських зображеннях. Встановлено, що найчастіше на рентгенівських зображеннях виникають білі, дробові та мультиплікативні шуми.
2. Проаналізовані існуючі програми для обробки рентгенівських зображеннях. Це програма Vidar Dicom Viewer, програма MediaSuite, програма Radiant.
3. Проаналізовані існуючі мови програмування для обробки зображень: C#, Python та Matlab. Проаналізували переваги та недоліки використання даних програм для обробки зображень та обрати мову програмування для дослідження фільтрації рентгенівських зображень. Для написання програми обробки рентгенівських зображень було обрано Matlab.
4. Досліджено існуючі методи фільтрації зображень. Було застосовано медіанний фільтр, усереднюючий фільтр та вейвлет-фільтр.
5. Зроблена оцінка методу фільтрації за максимальною похибкою. Отримані дані свідчать, що для фільтрації рентгенівських зображень потрібно обрати вейвлет-фільтр.

ДОДАТКИ

Додаток А

Програма для дослідження фільтрації рентгенівських зображень

```
% ДОСЛІДЖЕННЯ МЕТОДІВ ФІЛЬТРАЦІЇ ШУМІВ НА РЕНТГЕНІВСЬКИХ
% ЗОБРАЖЕННЯХ
ImageName='perelom_plecha_na_rentgene.png'; % ІМ'Я ФАЙЛА ЗОБРАЖЕННЯ
% ВИЗНАЧЕННЯ ШУМУ
Tsh='gaussian'; % ТИП ШУМУ НА ЗОБРАЖЕННІ
% 'gaussian' - "БІЛИЙ" ШУМ З НОРМАЛЬНИМ РОЗПОДІЛОМ
M=0; % СЕРЕДНЄ ЗНАЧЕННЯ ШУМУ
V=0.01; % ДИСПЕРСІЯ ШУМУ
Tsh='salt & pepper'; % ТИП ШУМУ НА ЗОБРАЖЕННІ
% 'salt & pepper' - ШУМ У ВИГЛЯДІ БІЛИХ І ЧОРНИХ ТОЧОК
D=0.05; % ЩІЛЬНІСТЬ ШУМУ НА ЗОБРАЖЕННІ
%Tsh='speckle'; % ТИП ШУМУ НА ЗОБРАЖЕННІ
% 'speckle' - МУЛЬТИПЛІКАТИВНИЙ ШУМ
% V=0.02; % ДИСПЕРСІЯ ШУМУ
% СТВОРЕННЯ ФІЛЬТРА
Tfilter='average'; % ТИП ФІЛЬТРА
% 'average' - УСЕРЕДНЮЮЧИЙ ФІЛЬТР
Hsize=3; % РОЗМІР КВАДРАТНОЇ МАСКИ ФІЛЬТРА
Filter=fspecial(Tfilter,Hsize); % СТВОРЕННЯ МАСКИ ФІЛЬТРА
% Tfilter='gaussian'; % ТИП ФІЛЬТРА
% 'gaussian' - ГАУСОВ ФІЛЬТР НИЖНІХ ЧАСТОТ
% РАДІУС МАСКИ ФІЛЬТРА, РОЗМІР КВАДРАТНОЇ МАСКИ Radius*2+1
% Radius=5;
% СТВОРЕННЯ МАСКИ ФІЛЬТРА
% Filter=fspecial(Tfilter,Radius);
% ЗАВАНТАЖЕННЯ ПОЧАТКОВОГО ЗОБРАЖЕННЯ
OrigImage=imread(ImageName);
if ndims(OrigImage) == 3
    OrigImage=rgb2gray(OrigImage);
end
% ДОДАВАННЯ ШУМУ ДО ЗОБРАЖЕННЯ
%NoiseImage = imnoise(OrigImage,Tsh,M,V);
% NoiseImage = imnoise(OrigImage,Tsh,D);
% NoiseImage = imnoise(OrigImage,Tsh,V);
% ФІЛЬТРАЦІЯ ЗОБРАЖЕННЯ
% УСЕРЕДНЮЮЧИЙ ФІЛЬТР З КВАДРАТНОЮ МАСКОЮ
% Hsize x Hsize ТОЧОК
% RestoreImage=imfilter(NoiseImage, Filter);
% МЕДІАННИЙ ФІЛЬТР
% РОЗМІР МАТРИЦІ ДЛЯ ВИЗНАЧЕННЯ СУСІДНИХ ТОЧОК
Mfilter=[3 3];
RestoreImage=medfilt2(NoiseImage,Mfilter);
%ВЕЙВЛЕТ ФІЛЬТР
% [c,s]=wavedec2(t1,2,'sym4');
%RestoreImage=wrcoef2('a',c,s,'sym4');
% ОБЧИСЛЕННЯ ВІДМІННОСТЕЙ МІЖ ВІДНОВЛЕННЯ І
% ПОЧАТКОВИМ ЗОБРАЖЕННЯМ
ErrorImage=uint8(abs(double(RestoreImage)-double(OrigImage)));
DeltaMean=double(mean2(ErrorImage));
DeltaMax=double(max(max(ErrorImage)));
% ВИВЕДЕННЯ РЕЗУЛЬТАТІВ
fprintf('МАКСИМАЛЬНЕ ЗНАЧЕННЯ ПОХИБКИ ВІДНОВЛЕННЯ %7.3f ДИСКРЕТНИХ РІВНЕЙ\n',
DeltaMax);
```

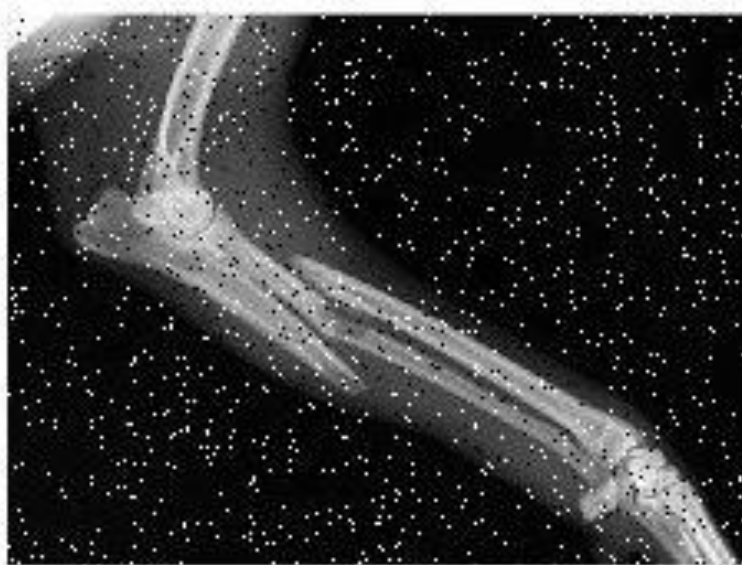
```
subplot(2,2,1); imshow(OrigImage);  
title('ПОЧАТКОВЕ ЗОБРАЖЕННЯ');  
subplot(2,2,2); imshow(NoiseImage);  
title('ЗОБРАЖЕННЯ З ШУМОМ');  
subplot(2,2,3); imshow(RestoreImage);  
title('ВІДНОВЛЕНЕ ЗОБРАЖЕННЯ');  
subplot(2,2,4); imshow(ErrorImage);  
title('ПОХИБКА ВІДНОВЛЕННЯ');  
figure; imshow(RestoreImage);  
title('ВІДНОВЛЕНЕ ЗОБРАЖЕННЯ');
```

Додаток Б

Обробка знімків

Обробка знімків трьох зразків дробовим шумом («сіль та перець») з дисперсією 0,05 для медіанного фільтра з маскою фільтра 3, усереднюючого фільтра з маскою фільтра 3 та вейвлет-фільтр.

Зображення з дробовим шумом (зразок 1)



Усереднюючий фільтр (зразок 1)



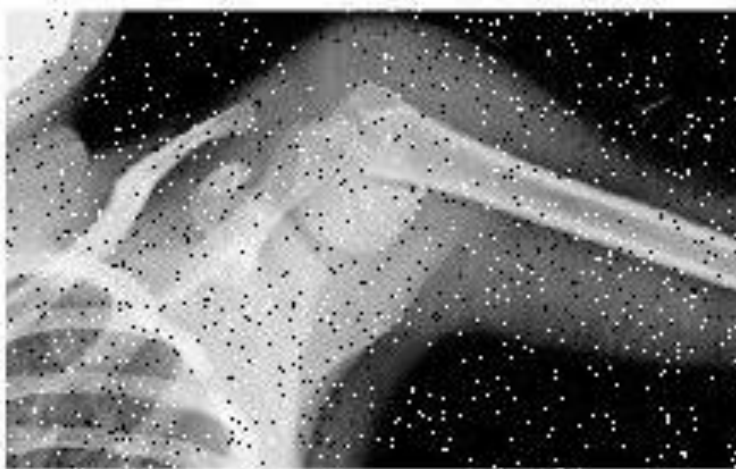
Медіаний фільтр (зразок 1)



Вейвлет-фільтр (зразок 1)



Зображення з дробовим шумом (зразок 2)



Усереднюючий фільтр (зразок 2)



Медіаний фільтр (зразок 2)



Вейвлет-фільтр (зразок 2)



Зображення з дробовим шумом (зразок 3)



Усереднюючий фільтр (зразок 3)



Медіаний фільтр (зразок 3)



Вейвлет-фільтр (зразок 3)



Значення максимальної похибки відновлення зображення для трьох зразків з мультиплікативним шумом з дисперсією 0,01 для медіанного фільтра з маскою фільтра 3, усереднюючого фільтра з маскою фільтра 3 та вейвлет-фільтр (представлені в табл. 3.2).