МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ В. Н. КАРАЗІНА
НАУКОВО-НАВЧАЛЬНИЙ ІНСТИТУТ
КОМП'ЮТЕРНОЇ ФІЗИКИ ТА ЕНЕРГЕТИКИ
Кафедра інформаційних технологій в фізико-енергетичних системах

ПОЯСНЮВАЛЬНА ЗАПИСКА

до дипломної роботи
за освітньо-кваліфікаційним рівнем бакалавр

«Дигіталізація процесів неруйнівного контролю»

Виконав: студент 4 курсу, групи НФ-43
Корнійчук М.Т.
спеціальність:
105 Прикладна фізика та наноматеріали
освітня програма: Прикладна фізика
енергетичних систем
Керівник: к.ф-м.н. Рогова С.Ю.

Рецензент:

Харків – 2020

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Харківський національний університет імені В.Н.Каразіна

Навчально-науковий інститут комп'ютерної фізики та енергетики
Кафедра інформаційних технологій у фізико-енергетичних системах
Рівень вищої освіти (освітньо-кваліфікаційний рівень) бакалавр
Спеціальність 105 Прикладна фізика та наноматеріали

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____Руслан СУХОВ

підпис

"____"_____2020 року

З А В Д А Н Н Я
НА ДИПЛОМНУ РОБОТУ

Корнійчук Микита Тимофійович

1.Тема роботи «Дигіталізація процесів неруйнівного контролю», керівник роботи: Рогова Світлана Юріївна, кандидат фізико-математичних наук, затверджені наказом по університету від "05" грудня 2019 року № 0210-05/2249

2. Строк подання студентом роботи 06.06.2020

3. Перелік питань, які потрібно розробити:

1) розглянути типи та задачі систем неруйнівного контролю;
2) дослідити ефективність запропонованого в дипломі методу покращення властивостей систем неруйнівного контролю;
3) розробити математичні алгоритми для реалізації запропонованого методу;
4) розробити програмне забезпечення програмно-апаратного комплексу;

5) провести тестування роботи програми та визначити якість запропонованих методів.

4. План роботи

№ з/п	Назви етапів роботи
1	Вибір теми, узгодження, а також затвердження її на кафедрі
2	Робота над літературою та іншими джерелами
3	Затвердження теми на кафедрі
4	Написання вступу до роботи та першого розділу
5	Написання другого розділу
6	Написання третього розділу
7	Завершення написання дипломної роботи (висновки та література)
8	Подача науковому керівнику
9	Відгук, рецензія, підготовка до захисту
10	Попередній захист
11	Захист

5.Дата видачі завдання

Студент _____ Микита КОРНІЙЧУК
підпис

Керівник роботи _____ Світлана РОГОВА
підпис

РЕФЕРАТ

Корнійчук Микита «Дигіталізація процесів неруйнівного контролю»

Кваліфікаційна робота бакалавра з прикладної фізики. – Х.: ХНУ імені В. Н.Каразіна, 2020. – 49 с.

У роботі досліджуються нові підходи до фільтрації та обробки цифрових рентгенограм і практична користь запропонованого алгоритму. Основна ідея полягає в використанні декількох цифрових зображень, використовуючи числові значення пікселів даних зображень, знаходиться середнє значення кожного відповідного пікселя і вже з усереднених пікселів відтворюється цифрове зображення.

Мета роботи – розробка і використання нових методів обробки цифрового зображення та реалізація цих методів за допомогою програмного забезпечення.

Об’єкт дослідження – методи обробки даних, за допомогою яких втрати корисної інформації будуть мінімальними.

Предмет дослідження – математичні методи покращення цифрового сигналу (у вигляді зображення).

У результаті виконання дипломної роботи було розроблено програмне забезпечення для фільтрації цифрової рентгенограми запропонованим алгоритмом підсумування.

КЛЮЧОВІ СЛОВА: ЦИФРОВА ОБРОБКА ЗОБРАЖЕНЬ, ЦИФРОВА РЕНТГЕНОГРАФІЯ, ФІЛЬТРАЦІЯ ЗОБРАЖЕНЬ, ЛІНІЙНЕ КОНТРАСТУВАННЯ ЗОБРАЖЕНЬ.

РЕФЕРАТ

Корнейчук Никита «Дигитализация процессов неразрушающего контроля»

Квалификационная работа бакалавра прикладной физики. - Х.: ХНУ имени В. Н.Каразина, 2020. - 49 с.

В работе исследуются новые подходы к фильтрации и обработки цифровых рентгенограмм и практическая польза предложенного алгоритма. Основная идея заключается в использовании нескольких цифровых изображений, используя числовые значения пикселей данных изображений, находится среднее значение каждого соответствующего пикселя и уже из усредненных пикселей воспроизводится цифровое изображение.

Цель работы разработка и использование новых методов обработки цифрового изображения, реализация этих методов с помощью программного обеспечения.

Объект исследования методы обработки данных, с помощью которых потери полезной информации будут минимальными.

Предмет исследования математические методы улучшения цифрового сигнала (в виде изображения).

В результате выполнения дипломной работы было разработано программное обеспечение для фильтрации цифровой рентгенограммы предложенному алгоритму суммирования.

КЛЮЧЕВЫЕ СЛОВА: ЦИФРОВАЯ ОБРАБОТКА ИЗОБРАЖЕНИЙ, ЦИФРОВАЯ РЕНТГЕНОГРАФИЯ, ФИЛЬТРАЦИЯ ИЗОБРАЖЕНИЙ, ЛИНЕЙНОЕ КОНТРАСТИНИРОВАНИЕ ИЗОБРАЖЕНИЙ.

ABSTRACT

Korneichuk Nikita "Digitalization of non-destructive testing".

Qualification work of the master of applied physics. – Kharkiv: V.N. Karazin Kharkiv national university, 2020. – 49 p.

The paper explores new approaches to filtering and processing digital radiographs and the practical benefits of the proposed algorithm. The main idea is to use several digital images, using the numerical values of the pixels of the image data, find the average value of each corresponding pixel and a digital image is already reproduced from the averaged pixels.

The purpose of this work ¬ development and use of new methods for processing digital images obtained using radiation monitoring and the implementation of these methods using software.

Object of study ¬ data processing methods by which the loss of useful information will be minimal.

Subject of research ¬ mathematical methods for improving a digital signal (as an image).

As a result of the thesis, software was developed to filter the digital radiographs of the proposed summation algorithm.

KEYWORDS: DIGITAL PROCESSING OF IMAGES, DIGITAL X-RAY ANALYSIS, FILTERING IMAGES, LINEAR CONTRASTING ENCHANCEMENT IMAGE.

ЗМІСТ

ВСТУП.....8
1. ТИПИ ТА ЗАДАЧІ СИСТЕМ НЕРУЙНІВНОГО КОНТРОЛЮ.....11
1.1.Задачі та типи неруйнівного контролю.....11
1.2.Класифікація та структура систем радіаційного неруйнівного контролю 11
1.3.Типи цифрової радіографії, її головні недоліки та переваги.....14
1.4.Постановка задачі.....17
2. ПОСТАНОВКА ЗАДАЧІ,НОВИЙ АЛГОРИТМ ПОКРАЩЕННЯ РАДІОГРАФІЧНОГО ЗОБРАЖЕННЯ. АЛЬТЕРНАТИВНІ МЕТОДИ ВИРІШЕННЯ ПРОБЛЕМИ.....18
2.1. Цифрові детектори.....18
2.2. Методи обробки та якісної оцінки цифрового знімку.....21
2.3.Альтернативні методи вирішення проблеми.....24
3. РЕАЛІЗАЦІЯ НОВОГО АЛГОРИТМУ ТА ТЕСТУВАННЯ НА ЦИФРОВИХ РЕТГЕНІВСЬКИХ ЗОБРАЖЕННЯХ РЕАЛЬНИХ ОБ’ЄКТІВ.....32
3.1. Інтерфейс програми.....32
3.2.Реалізація алгоритмів програми.....33
3.3.Порівняння результатів обробки цифрового зображення після запропонованого в дипломі алгоритму з алгоритмом медіанної фільтрації .37
3.4.Використання алгоритму для реального зображення.....43

ВСТУП

Актуальність теми. В останній час відбуваються якісні зміни в системах радіаційного неруйнівного контролю, а саме системах реального часу, серед яких можна виділити введення нових багатоелементних напівпровідникових детекторів зображень, отриманих за допомогою радіаційного неруйнівного контролю, а також інтенсивному росту та впровадженню цифрових технологій отримання, обробки, перетворення та аналізу зображень. Завдяки використанню такого типу детекторів і цифрових технологій рентгенівське випромінювання, яке проходить через досліджуваний об’єкт, за допомогою цифрових засобів перетворюється в масив даних (електричних сигналів), які потім зчитуються, обробляються і використовуються для моделювання просвічуваного об’єкта. Інформація, яка отримана за такою технологією містить в собі дані про його внутрішню структуру, матеріал, форму і може створюватись безпосередньо в момент просвічування, тобто в реальному часі. Такий метод радіаційного контролю називають цифровою радіографією або флеш-радіографією.

Отриманні, під час цифрової радіографії, дані про структуру досліджуваного об’єкту не завжди можуть бути однозначно оцінені фахівцями через появу дефектів при використовувані обладнання. Типи дефектів:

– цифровий шум на фотографії, основним джерелом яких виступають випадкові флуктуації числа поглинених фотонів ;

– геометричними або хроматичними аберації, або спотворення;

Проблеми такого типу приводять к постійному впровадженню великої кількості нових розробок та досліджень нових математичних алгоритмів.

У дипломі досліджуються нові методи покращення та збільшення корисної інформації у цифровому зображені. Основна ідея базується на використанні більшої кількості цифрових зображень, отриманих з використанням цифровою радіографії. В роботі використано метод лінійного контрастування, а також запропонований метод було порівняно з медіанною фільтрацією зображень. В

роботі було створено програмне забезпечення для наочної демонстрації ефективності описуваного в роботі методу та використано в реальних дослідженнях.

Для досягнення поставленої мети визначені задачі:

– розробити математичні алгоритми для реалізації покращення цифрових даних;

– розробити програмне забезпечення для використання запропонованих в роботі алгоритмів;

–провести тестування роботи програми та визначити якість запропонованих методів;

– дослідити ефективність запропонованого в дипломі методу покращення властивостей систем неруйнівного контролю;

Основна мета даної роботи – це розробка і використання нових методів обробки цифрового зображення отриманого за допомогою радіаційного контролю та реалізація цих методів за допомогою програмного забезпечення.

Об’єкт дослідження – методи обробки даних, за допомогою яких втрати корисної інформації будуть мінімальними.

Предмет дослідження – математичні методи покращення цифрового сигналу (у вигляді зображення).

Наукова новизна одержаних результатів.

– доведена корисність використовуваного алгоритму;

– розроблено програму для реалізації запропонованого в дипломі методу;

Практичне значення отриманих результатів:

– обґрунтовано можливість використання запропонованого методу покращення корисної інформації рентгенівського зображення;

– обґрунтовано актуальність використання цифрової радіографії;

Методи дослідження. Теоретична частина роботи базується на математичних алгоритмах обробки зображень та цифрових сигналів.

Структура і обсяг роботи. Дипломна робота бакалавра складається зі вступу, трьох розділів, висновків, списку використаних джерел та додатків.

1. ТИПИ ТА ЗАДАЧІ СИСТЕМ НЕРУЙНІВНОГО КОНТРОЛЮ

1.1.Задачі та типи неруйнівного контролю

На сьогоднішній день під неруйнівним контролем найчастіше розуміють аналіз надійності та основних робочих характеристик всього об'єкта або окремих його елементів, не пов'язаний з демонтуванням, розбиранням або порушенням його цілісної структури. Іншими словами, перевірка проводиться без руйнування об’єкта. Методами неруйнівного контролю вважаються [1 – 2,5]:

- магнітний;
- вихрострумовий;
- радіохвильової;
- оптичний;
- акустичний (ультразвуковий);
- радіаційний;
- тепловий;
- електричний;
- проникаючими речовинами;
- віброакустичний

1.2.Класифікація та структура систем радіаційного неруйнівного контролю

Серед усіх методів неруйнівного контролю [1] можна виділити радіаційний контроль з використанням рентгенівського та гамма-випромінювання. Радіаційний метод неруйнівного контролю перевершує об’ємом інформації про дефекти (форма, структура, розміри, місце розташування, матеріал) інші види неруйнівного контролю: магнітний, акустичний, вихрострумовий і ін.

Однак, радіаційний неруйнівний контроль має ряд суттєвих недоліків, серед яких [7]:

– неможливість отримати данні одразу після процесу, так як для отримання чіткого зображення експонований знімок об’єкта необхідно піддати тривалому процесу хіміко-фотографічної обробки (15 ... 30 хв);

– низький збіг думок фахівців при віднесенні проекції дефекту до однієї групи, через це об’єкт дуже часто декілька разів просвічують, що приводить до великих затрат плівки, на якій проявляється зображення;

– висока вартість рентгенівської плівки;

Радіографічний контроль не виявляє такі види дефектів [1]:

– якщо їх товщина в напрямку просвічування менш подвоєного значення абсолютної чутливості контролю;

– тріщини та дефекти виникаючі при зварюванні з розкриттям менше 0,1 мм, якщо товщина просвічує матеріалу до 40 мм, 0,2 мм - при товщині матеріалу від 40 до 100 мм, 0,3 мм - при товщині матеріалу від 100 до 150 мм;

– якщо зображення пустот і включень збігається на радіографічному знімку із зображенням сторонніх деталей, гострих кутів або різких перепадів товщини зварювальних елементів.

– тріщини, площина виявлення яких не збігається з напрямком просвічування;

Чутливість радіографічного контролю залежить від наступних факторів:

– геометричних умов радіографічного просвічування (значення фокусної плями рентгенівської трубки; відстань від фокусної плями трубки до деталі, від деталі до фіксуючої плівки);

– форми і розташування дефекту деталі відносно напрямку просвічування;

– жорсткості рентгенівського випромінювання, товщини і щільності матеріалу;

– характеристики фіксуючої плівки і коректності її обробки після експонування;

– використання підсилюючих екранів;

Системи радіаційного неруйнівного контролю складаються з трьох головних частин: джерела іонізуючого випромінювання, досліджуваного об'єкта та пристрою, що приймає випромінення. Системи радіаційного неруйнівного контролю залежно від типу приймаючого пристрою можна поділити наступним чином [3]: системи з проміжними носіями інформації і системи реального часу, використання таких систем знизить затрати часу, який було витрачено на процес хіміко-фотографічної обробки. Системи з проміжними носіями також називають системами із запам'ятовуванням цифрового зображення. В рентгенівських системах із запам'ятовуванням зображення радіаційне випромінювання виявляється і запам'ятовується на будь-якому носії (радіографічна плівка, електрорадіографічна пластина). Радіаційні системи неруйнівного контролю з проміжними носіями інформації широко застосовуються в промисловості і забезпечують високі відносну чутливість контролю. Однак такі системи мають суттєві недоліки: неможливість проводити контроль об'єктів в реальному часі, низька продуктивність і висока вартість.

Системи неруйнівного контролю реального часу без проміжних носіїв інформації дозволяють виконувати не тільки радіаційний контроль нерухомих об’єктів, а і рухомих об'єктів в реальному часі. До таких систем з проміжними носіями інформації можна віднести плівкова радіографія, електрорадіографія і комп'ютерна радіографія на основі запам'ятовуючих пластин. На сьогоднішній день радіографічні плівка залишається найбільш широко використовуваним детектором рентгенівського випромінювання з великим вмістом срібла (5 ... 10 г / м2). Зображення досліджуваного об'єкта формується на плівці, що складається з прозорої основи та нанесеним на неї желатином з суспензією кристалів галогенідусрібла. Після процесу просвічування проводиться хімічна обробка плівок і зчитування отриманої інформації, комп'ютерна обробка інформації, утилізація використаних плівок з метою повторного використання срібла. Серед важливих переваг плівкової радіографії слід назвати високу

відносну чутливість контролю (1%) в широкому діапазоні товщини досліджуваного об'єкту, високу роздільну здатність, велику площу робочого поля, можливість отримання в результаті контролю документа і т.д.

Радіаційні системи неруйнівного контролю з проміжними носіями інформації широко застосовуються в промисловості і забезпечують високі відносну чутливість контролю. Однак такі системи мають суттєві недоліки: неможливість проводити контроль об'єктів в реальному часі, низька продуктивність і висока вартість.

1.3.Типи цифрової радіографії, її головні недоліки та переваги

Всі сучасні дослідження в радіаційному неруйнівному контролі пов'язані зі зменшенням витрат внаслідок зменшення часів експозиції та зменшення вартості зберігання, зменшення необхідних робочих площ і виключення хімії з процесу обробки. Головними типами радіаційного неруйнівного контролю, які зосереджують у собі основне число таких технологій є:

1) Комп'ютерна радіографія з «фосфорними» пластинами багаторазового використання (насправді до складу запам'ятовуючих пластин хімічний елемент фосфор не входить [6]);

Системи комп'ютерної радіографії відрізняться за такими параметрами, як:

– Співвідношенню сигнал до шум;
– Просторова роздільна здатність;
– Розрядність оцифровки;
– По типу застосовуваної пластини;
– Вагою та габаритам.

Перевагами систем комп'ютерної радіографії перед плівковою технологією є:

– Можливість застосування в польових умовах;
– Відсутність процедури хімічної обробки;
– Можливість багаторазового використання пластини. Існують системи, де пластину можна використовувати до 25 000 разів;

– Висока чутливість;

– Висока просторова роздільність;

– Широкий діапазон можливих експозицій об'єкта контролю;

До обмежень даної технології можна віднести:

– Падіння чутливості контролю з ростом енергії рентгенівського випромінювання. Оптимальним діапазоном є енергії до 300 кеВ.

2) Цифрова радіографія з використанням цифрових детекторних систем цифрових детекторних систем.

Системи цифрової радіографії розрізняються:

– За типом виявляючого шару. Можуть бути використані напівпровідники або сцинтилятори.

– За типом виявляючої електроніки;

– По роздільної здатності – від десятків до сотень мікро метрів;

– По відношенню сигналу до шуму і по контрастності;

– За швидкістю роботи - від одного зображення за кілька хвилин до сотень зображень (кадрів) в секунду;

– За часом післясвітіння сцинтилятора;.

– По товщині матеріалу, яка може бути проконтрольована;

– Розрізняють плоскі панелі і лінійки. Їх розміри коливаються від одиниць до десятків сантиметрів;

– По розрядності цифрової здатності;

До обмежень цифрової радіографії можна віднести:

– Низьку просторову роздільну здатність більшості систем;

– Складність використання в польових умовах;

– Неможливість установки в важкодоступних місцях;

Цифрова радіографія була використана в першу чергу для заміни плівки в медицині. Цифрові детекторні системи, які були розроблені для цієї мети, дуже сильно вплинули на радіологічну техніку. Детектори такого типу дозволили розвинути нові, основані на комп'ютерних технологіях, додатки. Ці

технологічні та алгоритмічні розробки виявилися дуже ефективними також і для використання в неруйнівному контролі. Під цими технологіями мається на увазі нові технології калібрування цифрових детекторних систем, які дозволяють практично повністю позбутися їх власних шумів. Такі можливості відсутні у плівкових систем, систем комп'ютерної радіографії та запам'ятовуючих плівок. Через це цифрові детекторні системи мають найнижчий рівень шумів в радіографії і відкривають великі поле для нових досліджень, де потрібні надзвичайно високі чутливість і контрастність зображення в поєднанні з можливістю контролю слабких змін радіаційної товщини.

Надзвичайно важливим є також економічні переваги цифрових детекторних систем перед класичної плівковою технікою. Набагато більш швидкий процес обробки та інтерпретації при високій якості зображення призводить до суттєвого виграшу в часі в порівнянні з плівкою або іншими методами неруйнівного контролю.

Іноді, щоб краще розрізнити дефекти чи деталі досліджуваного об'єкту, необхідно збільшити контрастність або різкість рентгенівського знімка. Також, через випадкові флуктуації іонізуючого випромінювання та для більшої зручності перевіряючого, наявність дефектів, фахівця, необхідно використовувати фільтрацію зображення. Нажаль, серед усіх існуючих фільтрів (лінійних, нелінійних та комбінованих) [9 - 10] немає жодного, після використання якого, не виникнуть втрати корисної інформації [8]. Крім цього, поблизу горизонтальних країв з'являються розмитості, а поблизу точкових яскравих об'єктів з'являються помітні смуги і знижується загальна яскравість об'єктів. Ідеальним рішенням буде використання декількох рентгенівських знімків. Якщо в конкретній точці, усіх цифрових зображень, взяти числове значення пікселя та поділити на кількість цих зображень, ми отримаємо середнє значення пікселя. Таким чином локальні дефекти на одному знімку будуть скомпенсовані кількістю цифрових знімків без цього дефекту.

Контрастність знімка визначається різницею між значеннями оптичної щільності двох сусідніх ділянок знімка. Контрастність зображення визначається наступними факторами: контрастністю об'єкта і детектора (зазвичай радіографічної плівки). Контрастність об'єкта пропорційна різниці щільності ρ і атомного номера Z дефектних і бездефектних ділянок об’єкта і обернено пропорційна енергії випромінювання. Характеризується контрастність радіографічної плівки зміною щільності почорніння при використовуванні різних експозиційних доз випромінювання.

Різкість зображення на цифровому знімку характеризується стрибкоподібною зміною від однієї щільності почорніння до іншої на краях зображення. Чим вужче перехід від світлих ділянок до темних, тим більше розрізняються контури і тим більше різкість. Різкий знімок визначається добре виявленими контурами об'єкта і дефектів в матеріалі, що забезпечує високе виявлення цих самих дефектів. Більш ширший перехід від світлих ділянок до темних, призводить до більшої розмитості контурів і меншої різкості зображення, отже, гіршого виявлення дефектів.

1.4.Постановка задачі

Необхідно розробити програмний продукт, який буде зберігати, відкривати, контрастувати та реалізувати запропонований алгоритм фільтрації цифрового зображення. Для порівняння алгоритму підсумування та медіанної фільтрації розглянути та проаналізувати сучасні методи фільтрації зображення. Також, треба дослідити схему виникнення цифрового зображення при цифровій рентгенографії.

2. ПОСТАНОВКА ЗАДАЧІ,НОВИЙ АЛГОРИТМ ПОКРАЩЕННЯ РАДІОГРАФІЧНОГО ЗОБРАЖЕННЯ. АЛЬТЕРНАТИВНІ МЕТОДИ ВИРІШЕННЯ ПРОБЛЕМИ

2.1. Цифрові детектори

Цифрові детектори існують двох типів [11]:

1. Цифрові детектори непрямого перетворення рентгенівського випромінювання.
2. Цифрові детектори прямого перетворення рентгенівського випромінювання

Детектори непрямого перетворення (Рис.2.1.), найбільш часто використовуються на практиці.



Рис.2.1. Зовнішній вигляд плоскопанельного детектора.

Різниця між детекторами непрямого перетворення в тому, що рентгенівські кванти спочатку взаємодіють з сцинтилятором та утворюють фотони світла, а потім світло перетвориться або в електричний сигнал, як в твердотільних

плоскопанельних детекторах (Рис.2.2.), або в потік електронів в рентгенівському електронооптичному перетворювачі, який створює цифрове зображення на вихідному екрані.

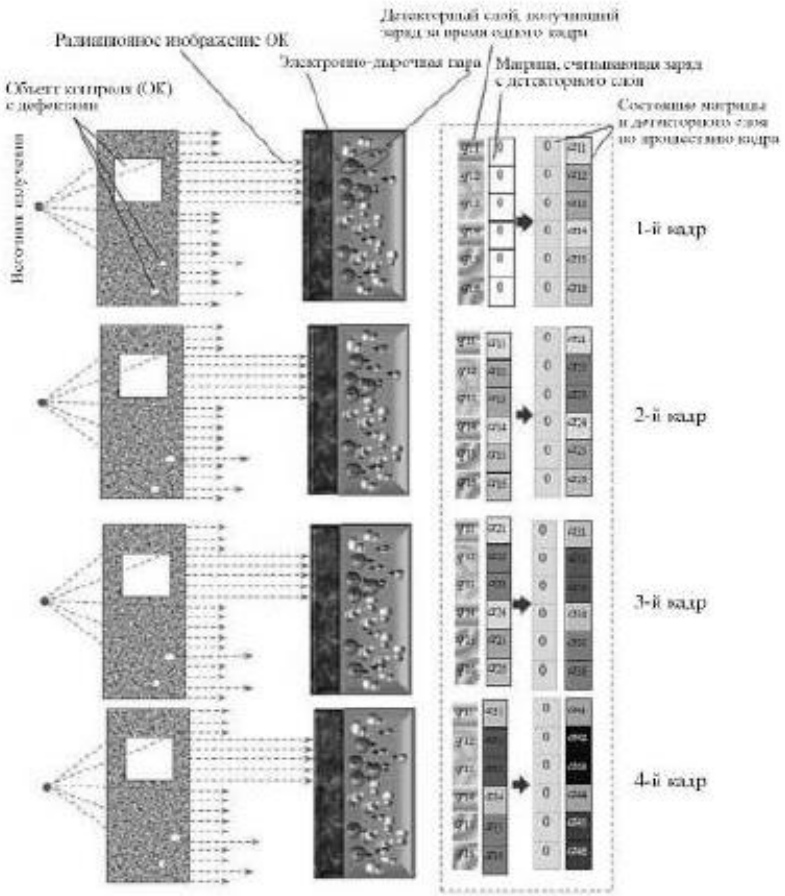


Рис.2.2. Процес формування цифрового зображення у плосокопанельном детекторі

В якості плоскопанельних детекторів непрямого перетворення найчастіше використовують панелі на основі аморфного кремнію. Їх робота основується на сцинтиляторах, які перетворюють рентгенівське випромінювання в видиме світло. Це світло конвертується в заряд сенсорами з аморфного кремнію. Чутливість детектора залежить від товщини приймаючого випромінювання покриття. Однак, через розсіювання світла, при збільшенні товщини поверхні погіршується просторова роздільна здатність детектора.

Ранні експерименти з аморфним кремнієм виявили високий рівень шуму панелей такого типу. Розвиток технологій разом з покращенням програмних можливостей, дозволив різко поліпшити відношення сигнал / шум. Якість зображень, що отримуються зараз на цих панелях, перевищує якість, що отримується на запам'ятовуючих пластинах, і наближається до якості цифрових зображень на панелях з аморфного селену.

Також, перевагою є те, що панелі з аморфного кремнію менш чутливі до властивостей навколишнього середовища, що робить можливим їх застосування в польових умовах.

У детекторах прямого перетворення при попаданні рентгенівського кванта на поверхню в товщі напівпровідника відразу генеруються електронно-діркові пари. Після чого під дією напруги виникає електричний ток.

Найбільш поширеними типами цифрових детекторних систем прямого перетворення є панелі на основі аморфного селену і монокристалічного телуриду кадмію. В першому випадку збірка тонкоплівкових транзисторів покривається аморфним селеном, цим самим рентгенівське випромінювання безпосередньо перетворюється в цифровий сигнал без використання сцинтиляторів або фосфору. Внаслідок цього відсутність впливу розсіювання і оптимальне відношення сигнал / шум, таким чином якість зображення досягає якості, як у середньозернистої плівки.

В плоскопанельних детекторах на основі селену є суттєвий недолік вузький діапазон робочої температури. Все через те, що при занадто низьких або високих температурах шар селену руйнується. Як для роботи, так і для зберігання детектора необхідно дотримувати температуру в діапазоні від 5 до 30 ° С. Ще одним суттєвим недоліком є те, що при високих енергіях селен може утворювати фантомні зображення.

Всі ці обмеження не дозволяють використовувати данні детектори в польових умовах. Детекторна система прямого перетворення на основі телуриду кадмію позбавлена ряду недоліків системи на основі аморфного селену. Вона може

працювати в набагато ширшому температурному діапазоні при енергіях рентгенівських квантів до 300 кеВ, володіючи при цьому в кілька разів більшою чутливістю.

Більш інформативна схема конструкції плоскопанельного детектора показана на Рис. 2.3.

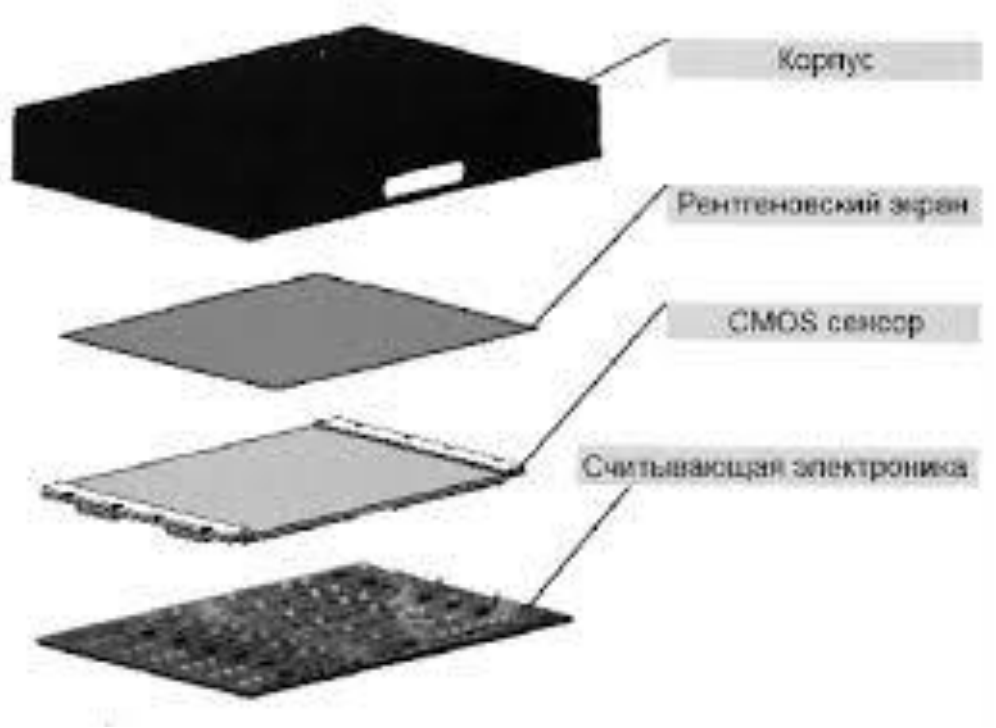


Рис.2.3. Детальна схема конструкції плоскопанельного детектора

2.2. Методи обробки та якісної оцінки цифрового знімку

В певний момент часу, який називають кадром, на вхід детекторного шару матриці поступають фотони рентгенівського випромінювання, енергія яких була зменшена після проходження досліджуваного об’єкту. Під час випромінювання в рентгеночутливих елементах виникає електричний сигнал, величина якого визначається кількістю отриманих фотонів. Весь масив електричних сигналів поступає на аналого-цифровий перетворювач, де перетворюється на цифровий масив. Після чого детектуючий шар стирає отримані дані і весь алгоритм повторюється. Отриманий другий кадр знову перетворюють в цифровий масив і

підсумовується з попереднім значенням. Постійне підсумовування зображень приводить к виникненню постійного зростання відношення сигналу до шуму, так як було встановлено, що сигнал зростає пропорційно кількості отриманих фотонів – N, а шум – $\sqrt{N}$. Отриманий цифровий масив даних, представлений у вигляді зображення, виводиться на монітор дефектоскописта.

Якість рентгенівського знімка визначається двома параметрами: чіткістю, контрастом та базову просторову здатність (показник роздільної здатності системи або роздільна здатність). Контраст рентгенівського зображення визначається досягнутою чутливістю, а чіткість – зворотна нерізкості зображення.

Роздільна здатність визначається числом роздільно помітних ліній, які перебувають на 1 мм детектора. Найбільш простим, як показано в дослідженні [12], і, згідно з результатами, які були одержані в практичних завданнях, найбільш зручним способом оцінки роздільної здатності є використання дводротового (дуплексного) еталона [13]. Еталон є 13 пар дротів різного діаметру. У кожній з пар дроту розташовуються один від одного на відстані, що дорівнює їх діаметру Рис.2.4. Пари дротів в залежності від діаметра пронумеровані від найбільших (1Д) до найтонших (13Д). Пари 1D - 3D виконуються з вольфраму, інші - з платини. Показник роздільної здатності визначається як номер останньої пари зволікань, яку можливо розрізнити, і дорівнює половині зареєстрованої нерізкості зображення або ефективному розміром пікселя. Залежно від радіаційної товщини об'єкта контролю і класу контролю встановлюють вимоги до мінімального значення SRb.

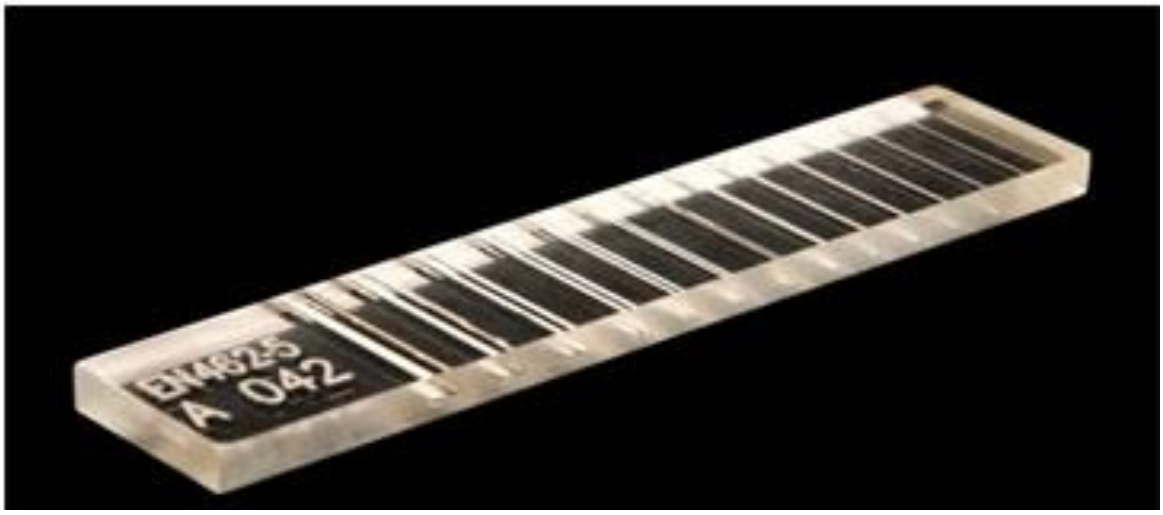


Рис.2.4. Дводротовий еталон

Іноді контраст зображення не дозволяє отримати повну інформацію про границі або форму об’єктів. Як правило, для цифрового представлення кожного пікселя зображення відводиться 1 байт пристрою, що запам'ятовує, то вхідний або вихідний сигнали можуть приймати одне з 2^8 (256) значень. Зазвичай, значення кольору знаходиться в діапазоні від 0 до 255, при цьому значення 0 візуалізації відповідає чорному кольору, а значення 255 - білому. Для покращення візуальної ідентифікації деталей цифрового знімку використовують контрастування [14]. В даному випадку будемо використовувати лінійне контрастування. При лінійному контрастуванні використовується лінійне попіксельне перетворення виду:

$$y = a * x + b$$

де y – вихідне значення яскравості, a, b – коефіцієнти, які визначають максимальне (y_{max}) та мінімальне (y_{min}) значення яскравості, x – початкове значення.

Побудуємо систему рівнянь y_{max} та y_{min}:

$$\begin{cases} y_{max} = a * x_{max} + b \\ y_{min} = a * x_{min} + b \end{cases}$$

де x_{max}– максимальне значення яскравості початкового зображення, а x_{min}– мінімальне.

Приведемо систему рівнянь до вигляду:

$$y = \frac{x - x_{min}}{x_{max} - x_{min}} * (y_{max} - y_{min}) + y_{min}$$

2.3.Альтернативні методи вирішення проблеми

При фільтрації, числова характеристики яскравості кожної точки цифрового зображення, замінюється іншим значенням яскравості, яке визнається в найменшій мірі спотвореним шумом [15]. Розрізняють частотну і просторову фільтрацію [15].

Частотні методи перетворень зображень [8-10,15-19] базуються на принципі Фур'є перетворення, основна ідея якого полягає в представлені вихідної функції у вигляді суми тригонометричних функцій різних частот, які були помножені на задані коефіцієнти. У разі, якщо вихідна функція періодична таке представлення називається рядом Фур'є. В іншому випадку, неперіодична функція, що має кінцеву площу під графіком, може бути виражена як інтеграл від тригонометричних функцій, помножених на деяку вагову функцію [15]. Такий випадок називається перетворенням Фур'є і в більшості практичних задач являється більш інформативним, ніж ряд Фур'є. Також слід зауважити, що важливою властивістю Фур'є-перетворення можна назвати те, що після здійснення над даною функцією перетворень її можна повернути до початкового стану. Таким чином, даний метод дозволяє обробляти частотну функцію в області, після чого повернутися до початкового вигляду без втрати інформації. Також перетворення Фур'є може застосовуватися для вирішення завдань фільтрації зображень. У практичних задачах реалізація частотних методів може бути аналогічна просторовим методам фільтрації [9-10,15-19].

Просторові методи покращення зображень застосуються до зображень, які представлені у вигляді двовимірних матриць. Основний принцип просторових

алгоритмів базується на використанні спеціальних операторів до кожного числового значення точки вихідного зображення. В якості операторів використовують прямокутні або квадратні матриці, які називають масками ядрами або вікнами [8-10,15-19]. Найчастіше маска представлена невеликим двовимірним масивом, а методи покращення зображення, які ґрунтуються на даному підході, часто називають обробкою по масці або фільтрацією по масці. На методі використання маски базуються лінійні, нелінійні (медіанні) та комбіновані (адаптивні) типи фільтрів.

При використані лінійної фільтрації вихідне значення пікселя задається сумою множення пікселів в області використання фільтра. У якості лінійного згладжуючого фільтра використовується усереднюючий фільтр з вихідним значенням, яке є середнім по усім крайнім пікселям маски фільтра. Такий фільтр використовується для видалення зернистості, яка була викликана імпульсним шумом. Загальна формула вихідного пікселя $g(x, y)$ усредняющего фільтра, який призначений для фільтрації зображення f з розмірами $M \times N$, має вигляд [15]:

$$G_{i,j} = \sum_{s=(m-1)/2}^{(m-1)/2} \sum_{t=(n-1)/2}^{(n-1)/2} W_{s,t} * E_{(i+s),(j+t)}$$

де $G_{i,j}$ – елемент матриці пікселів цифрового зображення;

$W_{s,t}$ – елемент масиву ядра згортки зображення;

$E_{i,j}$ – елемент матриці почткового зображення

Принцип використання нелінійних методів фільтрації подібний лінійним фільтрам. Операції або перетворення, що базуються на методі нелінійної фільтрації, залежать від числових значень елементів матриці цифрового зображення, які знаходяться в аналізованій масці [15].

Найпростішим і, в той же час, базовим методом нелінійної фільтрації є медійний фільтр. Принцип роботи даного фільтра полягає в обчисленні медіани значень елементів зображення аналізованого вікна. При медіанної фільтрації значення пікселів обчислюється як усереднене значення усіх точок відповідної

маски. Часто при вирішенні прикладних задач вилучення шуму медіанний фільтр є більш ефективним, а ніж звичайне усереднення, так як, при його використанні, виникають менші спотворення кордонів об'єктів обробленого зображення. Маски при медіанної фільтрації представляють собою двовимірне вікно з однаковим значенням висоти та ширини, при цьому його центр розташовується в поточній точці фільтрації (Рис.2.5) [8-10,15-19]. Метод двовимірного медіанного фільтра визначає формула:

$$G_{i,j} = med\left[E_{i+s,j+t}; (s,t) \in W\right]; i,j \in Z^2$$

де $G_{i,j}$-елемент матриці пікселів цифрового зображення після фільтрації,$E_{i,j}$ - елемент матриці початкового зображення, $W_{s,t}$-елемент масиву ядра згортки зображення.

223	186	114
204	161	106
219	194	138

106 114 138 161 **186** 194 204 219 223

Рис.2.5. Робоче вікно при використанні медіанної фільтрації.

При застосуванні медіанного і лінійного фільтрів не враховується факт того, що параметри зображення змінюються при переході від точки до точки. Адаптивні фільтри позбавлені цього недоліку через те, що їх властивості змінюються відповідно до змін значення характеристик зображення. Функціонал адаптивних фільтрів може бути ширшим, а ніж можливості лінійних і медіанний

фільтрів, на основі яких вони створені. Все це залежить, як від характеристик самого зображення, так і його стану спотворення шумом. При цьому збільшується складність алгоритмів і час роботи фільтра.

В основі адаптивної лінійної фільтрації покладено вінеровський фільтр, який є одним з типів лінійного фільтра для адаптивної локальної обробки зображень [18]. Якщо значення середньоквадратичного відхилення інтенсивностей пікселів в даній локальної області велике, то винеровский фільтр виконує невелике згладжування і, навпаки, при меншому відхиленні область згладжування більше. Цей підхід часто буває більш ефективним, ніж звичайна лінійна фільтрація. Перевага адаптивний фільтра ще полягає в тому, що він зберігає краю та інші високочастотні частини об'єктів зображення. Однак, вінеровський фільтр вимагає більшого часу для обчислень, ніж лінійний фільтр.

Середнє значення яскравості обчислюється для центрального пікселя маски W_{ij}, містить значення яскравості вихідного зображення в покритої маскою локальної області зображення за формулою:

$$\overline{w} = \frac{1}{m * n} * \sum_{i=1}^{m} \sum_{j=1}^{n} W_{ij}$$

Дисперсія маски дорівнює:

$$\boldsymbol{\delta^2 = \frac{1}{m * n} * \sum_{i=1}^{m} \sum_{j=1}^{n} (W_{ij} - \overline{w})^2}$$

В даному алгоритмі для кожного нового положення вікна маски заново обчислюються відповідні значення. Згладжування шуму оцінюється через середнє квадратичне відхилення:

$$\delta = \sqrt{\delta^2}$$

Адаптивний алгоритм на базі медіанного фільтра також було описано в [14]. При використанні медіанного фільтра значення пікселя завжди замінюється медіаною. Вихідне значення пікселя після медіанної фільтрації, при високій щільності імпульсного шуму помітно погіршується. Адаптація базується на тому, що спочатку піксель виявляється на імпульс, після чого теж саме відбувається і з медіаною. Заміна відбувається тільки в тому випадку, якщо піксель представлений імпульсом, а медіана - ні. Якщо медіана виявилася імпульсом, відбувається збільшення розмірів маски, що змінить значення медіани і, можливо, поліпшить результати оскільки тепер все обчислюється заново.

Збільшення розміру маски забезпечує покращення видалення шуму, але використання даного методу може спотворити контури або границі об’єктів. Тому розмір маски повинен слідувати за самим зображенням, в монотонних областях можна використовувати великі розміри маски, поблизу перепадів яскравості розмір вікна слід зменшувати. Це реалізовано в алгоритмі на основі обчислення дисперсії - локальної та всього зображення.

Також, розмиття зображення можна зменшити, вибираючи серед точок маски такі, яскравості яких мають незначні відмінності і обчислювати середнє по цим точкам. Такий фільтр називають адаптивним з вибором шаблону. Значення яскравості центрального елемента визначається як середнє значення пікселя в шаблоні і де дисперсія буде найменшою. Окрім зменшень втрат корисної інформації, така обробка дозволяє використовувати алгоритм фільтрації до різних типів шуму, що представляє особливий інтерес в разі, якщо тип шуму заздалегідь невідомий.

На Рис.2.5 наведені результати фільтрації при накладенні імпульсного шуму на цифрове зображення. На Рис.2.6 представлені результати фільтрації накладеного на цифрове зображення гауссовського білого шуму.

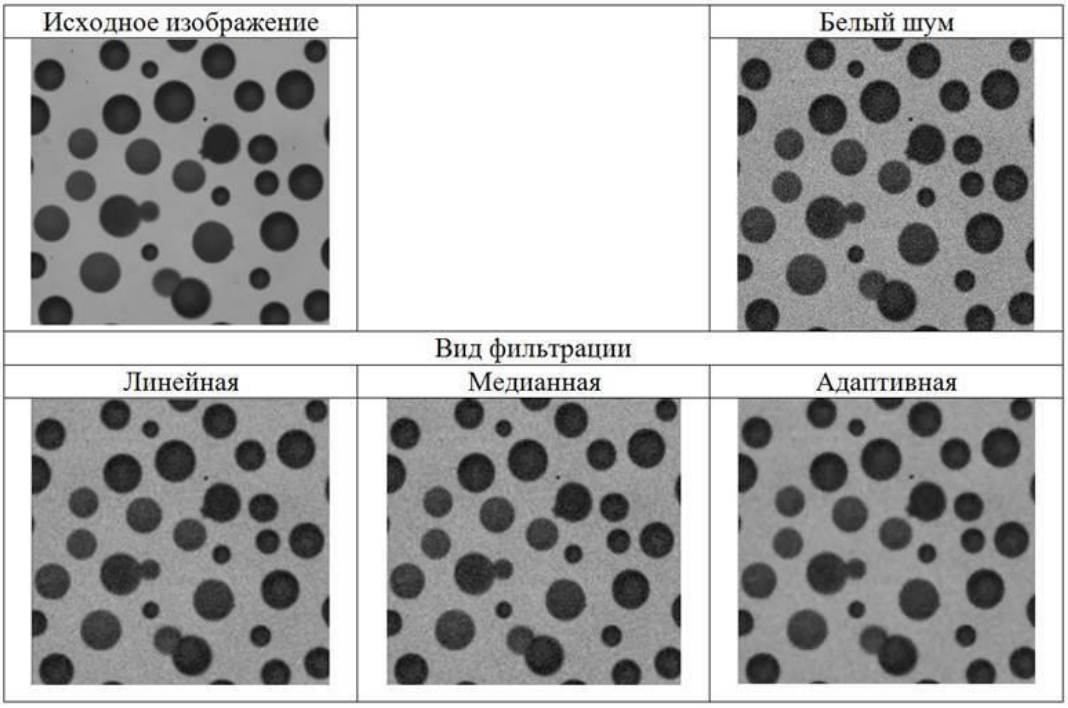


Рис.2.6. Приклад фільтрації різними фільтрами білого шуму. Адаптивний фільтр на базі вінеровського.

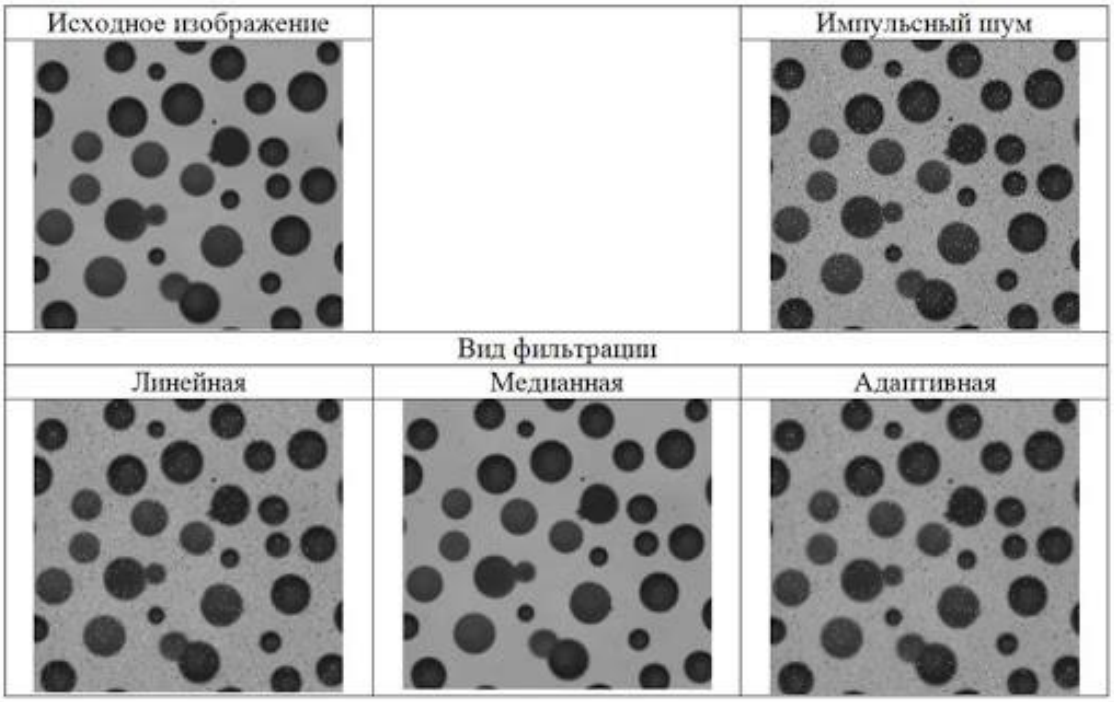


Рис.2.7. Приклад фільтрації різними фільтрами імпульсного шуму. Адаптивний фільтр на базі вінеровського.

На питання, який з усіх представлених типів фільтрації краще всього використовувати при різних типів шумів можна знайти відповідь у роботах [12, 13, 16] . В дипломі для порівняння з запропонованим методом поліпшення зображення буде використовуватися медіанна фільтрація. Також, для більшої наочності добутих результатів буде обчислюватись дисперсія зображення та математичне очікування.

Математичним очікуванням дискретної випадкової величини називається сума творів можливих значень випадкової величини на ймовірності цих значень. Якщо випадкова величина X характеризується кінцевим набором значень, то математичне сподівання $M\ (X)$ визначається за формулою:

$$M(X) = \sum_{i=1}^{n} x_i * p_i$$

де x_i-значення пікселя,p_i-ймовірність зустрічі i- го значення пікселя.

Математичне очікування для неперервної випадкової величини визначається рівністю:

$$M(X) = \int_{-\infty}^{+\infty} x * \varphi(\mathrm{x})\, dx$$

де $\varphi(\mathrm{x})$ – щільність ймовірності випадкової величини.

Дисперсією випадкової величини називають математичне очікування квадрата відхилення випадкової величини від її математичного очікування.

$$D(X)^2 = M([X - M(X)]^2)$$

2.4 Основна ідея алгоритму покращення зображення

Основною ідеєю диплому являється використання лише вихідних цифрових зображень, не використовуючи алгоритмів фільтрації, які приводять до втрати

корисної інформації. При цьому дисперсія повинна зменшуватись по наступній формулі:

$$\frac{D(x_{\text{вх.}})}{D(x_{\text{вх.}})} = N$$

де $D(x_{\text{вх.}})$ – дисперсія вхідного зображення

$D(x_{\text{вх.}})$ – дисперсія вихідного зображення

$N -$ кількість цифрових зображень.

Так як алгоритм буде порівнюватися з медіаною фільтрацією, окрім візуального порівняння необхідно вирахувати дисперсію обох зображень. Однак, через те що, на реальному чорно-білому знімку, зазвичай, велика кількість як білих так і чорних пікселів, тому значення дисперсії всього цифрового знімка буде не вірним. В таких випадках необхідно вибирати область зображення і вже в ньому обчислювати дисперсію. Через це в дипломі буде використовуватись створене цифрове зображення с одним значенням пікселя, після чого на нього накладається псевдовипадковий шум. Для цього було обрано псевдовипадкову функцію вихрю Мерсенна.

3. РЕАЛІЗАЦІЯ НОВОГО АЛГОРИТМУ ТА ТЕСТУВАННЯ НА ЦИФРОВИХ РЕТГЕНІВСЬКИХ ЗОБРАЖЕННЯХ РЕАЛЬНИХ ОБ'ЄКТІВ

3.1. Інтерфейс програми

Для реалізації програми була обрана середа розробки Microsoft Visual Studio Community 2019. За допомогою інструментів Windows Forms було створено основний діючий функціонал та інтерфейс (Рис.3.1.).

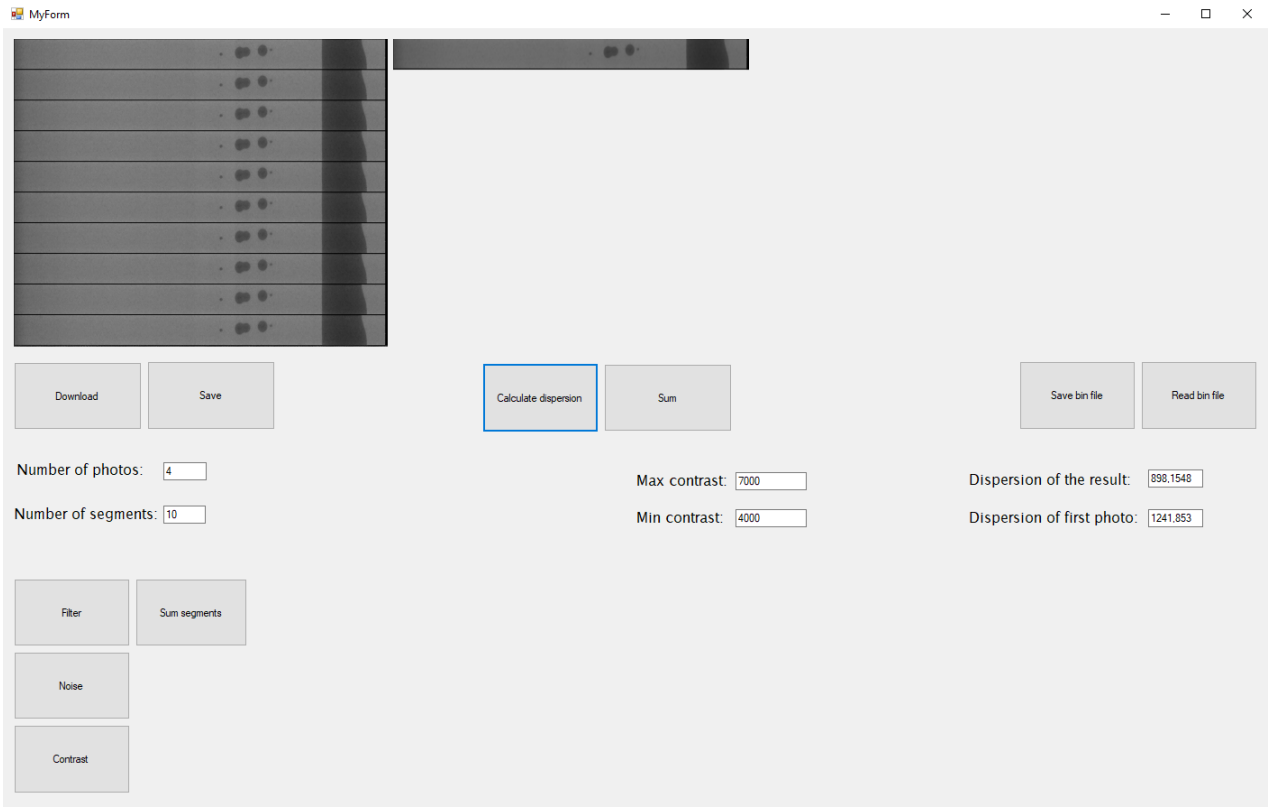


Рис.3.1. Інтерфейс програми

Наявні наступні кнопки:

– «Download»-відкриття цифрового зображення, які мають розширення «.jpg», «.png».
– «Save»-збереження цифрового зображення, розширення «.jpg», «.png».

- «Calculate dispersion»-обчислення дисперсії двох зображень, результат обчислення дисперсії вхідного зображення та обробленого записуються в поля «Dispersion of the photo»та «Dispersion of the result» відповідно.
- «Sum»-використати алгоритм підсумування, для чого необхідно задати кількість зображень в поле «Number of photos».
- «Save bin file»-збереження файлу в документ, як двобайтний масив.
- «Read bin file»-відкриття файлу з двобайтними числовими значеннями.
- «Filter»-кнопка для використання медіанної фільтрація.
- «Noise»-створення однокольорового зображення з шумом.
- «Contrast»-зміна контрасту зображення, для чого необхідно ввести мінімального та максимального контрасту в поля «Min contrast». І «Max contrast» відповідно.
- «Sum segments»-підсумування

3.2.Реалізація алгоритмів програми

Для початкової перевірки роботи коду було необхідно використовувати зображення з шумом. На Рис.3.2 зображений код для створення псевдовипадкового цифрового шуму на зображенні, за допомогою псевдовипадкової функції вихрю Мерсенна. Вихор Мерсенна, основується на простих числах Мерсенна та забезпечує швидку генерацію високоякісних за критерієм випадковості псевдовипадкових чисел. Він позбавлений багатьох недоліків, властивих іншим генераторам псевдовипадкових чисел, таких як малий період, передбачуваність, легко виявити статистичну закономірність.

```
private: System::Void noise_Click(System::Object^ sender, System::EventArgs^ e) {
    std::random_device rd;
    std::mt19937 mersenne(rd());
    Bitmap^ input = gcnew Bitmap(pictureBox->Image);
    Bitmap^ output = gcnew Bitmap(input->Width, input->Height);
    Img_Height = input->Height;
    Img_Width = input->Width;
    int x;
    int y;
    Img_Mass = new unsigned char[Img_Height * Img_Width];
    for (x = 0; x < Img_Width; x++)
    {
        for (y = 0; y < Img_Height; y++)
        {
            Img_Mass[x + y * Img_Width] = 120;
        }
    }
    for (int rand = 0; rand < 11000; rand++) {
        Img_Mass[(mersenne() % (Img_Height- 1 )) + y * (mersenne() % (Img_Width - 1))] = mersenne() % 255;
    }
    for (x = 0; x < Img_Width; x = x++)
    {
        for (y = 0; y < Img_Height; y++)
        {
            output->SetPixel(x, y, Color::FromArgb(255, Img_Mass[x + y * Img_Width], Img_Mass[x + y * Img_Width], Img_Mass[x + y * Img_Width]));
        }
    }
    pictureBox_processed->Image = output;
    delete[] Img_Mass;
}
```

Рис.3.2. Код програми для створення зображення з псевдовипадковим шумом

Для реалізації алгоритму підсумовування необхідно знати кількість цифрових зображень, яку можна ввести в інтерфейсі програми див. Рис.3.1. (поле «Number of photos»). Назва цифрового зображення, для коректного зчитування, повинна бути виду «test1», «test2» и т.д. На Рис.3.3. показано код для реалізації запропонованого алгоритму.

```
    for (number = 1; number < size+1; number++)
    {

        pictureBox->Load("D:\\Work\\X_Ray\\test_noise" + number + ".jpg");
        Bitmap^ input_new = gcnew Bitmap(pictureBox->Image);
        for (x = 0; x < Img_Width;x++)
        {
            for (y = 0; y < Img_Height; y++)
            {
                UInt32 pixel = (UInt32)(input_new->GetPixel(x, y).ToArgb());
                unsigned char R = (unsigned char)((pixel & 0x00FF0000) >> 16);
                unsigned char G = (unsigned char)((pixel & 0x0000FF00) >> 8);
                unsigned char B = (unsigned char)(pixel & 0x000000FF);

                Img_Mass[Num * Img_Height * Img_Width + (x + y * Img_Width)] = R;
            }
        }
        Num++;
    }
    for (x = 0; x < Img_Width;  x++)
    {
        for (y = 0; y < Img_Height; y++)
        {
            for (number = 0; number < size; number++)
            {
                rez = rez + Img_Mass[number * Img_Height * Img_Width +(x + y * Img_Width)];
            }
            Img_Mass[x + y * Img_Width] = rez / size;
            rez = 0;
        }
    }
    for (x = 0; x < Img_Width; x++)
    {
        for (y = 0; y < Img_Height; y++)
        {
            output->SetPixel(x, y, Color::FromArgb(255, Img_Mass[x + y * Img_Width], Img_Mass[x + y * Img_Width], Img_Mass[x + y * Img_Width]));
        }
    }
```

Рис.3.3. Код програми для підсумовування пікселів декількох зображень.

Як вже було описано, іноді, необхідно використати методи контрастування зображення для більшої наглядності. Для досягнення цього було використано метод лінійного контрастування (Рис.3.4.). Також в подальшому через те, що для збереження цифрового зображення (Рис.3.5.) зручніше використовувати тип данних с двома пікселями, було обрано тип «unsigned short». Для обробки зображення використовується тип «unsigned char» так,як на вхід функции «SetPixel» для значення пікселів можно використовуввати лише числа від 0 до 255. Тому, після запису числового значення пікселя в «unsigned short» (область значення якого від 0 до 65536) і подальшому считуванні такого цифрового зображення (Рис.3.6.), воно буде набагато світлішим, а ніж має бути. В такому випадку також використовуеться контрастування.

Важливо зауважити, що при лінійному контрастуванні необхідно задавати мінімальне (див. Рис.3.1. поле «Min contrast») та максимальне (див. Рис.3.1. поле «Max contrast») значення пікселя.

```
Img_Height = input->Height;
Img_Width = input->Width;
int x;
int y;
int min_contrast = Convert::ToInt32(textBox5->Text);
int max_contrast = Convert::ToInt32(textBox4->Text);
int photo_max = 0;
int  photo_min = 0;
Img_Mass = new unsigned char[Img_Height * Img_Width];
Short_Mass = new unsigned short[Img_Height * Img_Width];
for (x = 0; x < Img_Width; x++)
{
    for (y = 0; y < Img_Height; y++)
    {
        UInt32 pixel = (UInt32)(input->GetPixel(x, y).ToArgb());
        float R = (float)((pixel & 0x00FF0000) >> 16);
        Short_Mass[x + y * Img_Width] = R;
    }
}
for (x = 0; x < Img_Width; x++)
{
    for (y = 0; y < Img_Height; y++)
    {
        if (photo_min > Short_Mass[x + y * Img_Width]) {
            photo_min = Short_Mass[x + y * Img_Width];
        }
        if (photo_max < Short_Mass[x + y * Img_Width]) {
            photo_max = Short_Mass[x + y * Img_Width];
        }

    }
}
for (x = 0; x < Img_Width; x++)
{
    for (y = 0; y < Img_Height; y++)
    {
        Img_Mass[x + y * Img_Width] = (Short_Mass[x + y * Img_Width] - photo_min) / (photo_max - photo_min) * (max_contrast - min_contrast) + min_contrast;
    }
}
for (x = 0; x < Img_Width; x++)
{
    for (y = 0; y < Img_Height; y++)
    {
        output->SetPixel(x, y, Color::FromArgb(255, Short_Mass[x + y * Img_Width], Short_Mass[x + y * Img_Width], Short_Mass[x + y * Img_Width]));
    }
}
pictureBox_sum->Image = output;
delete[] Img_Mass;
}
```

Рис.3.4. Код програми для лінійного контрастування зображення.

```
774    private: System::Void button2_Click(System::Object^ sender, System::EventArgs^ e) {
775        //Save Img_Mass to bin file
776        Bitmap^ input = gcnew Bitmap(pictureBox->Image);
777        Bitmap^ output = gcnew Bitmap(input->Width, input->Height);
778        Img_Height = input->Height;
779        Img_Width = input->Width;
780        Img_Mass = new unsigned char[Img_Height * Img_Width];
781        Short_Mass = new unsigned short[Img_Height * Img_Width];
782        for (int x = 0; x < Img_Width; x++)
783        {
784            for (int y = 0; y < Img_Height; y++)
785            {
786                UInt32 pixel = (UInt32)(input->GetPixel(x, y).ToArgb());
787                float R = (float)((pixel & 0x00FF0000) >> 16);
788                float G = (float)((pixel & 0x0000FF00) >> 8);
789                float B = (float)(pixel & 0x000000FF);
790                float L = (float)(pixel & 0xFF000000);
791
792                Img_Mass[x + y * Img_Width] = R;
793                Short_Mass[x + y * Img_Width] = L + R * 255;
794            }
795        }
796        FILE* Mass = fopen("test.bin","wb");
797        FILE* Short = fopen("test_short.bin", "wb");
798        fwrite(Img_Mass, sizeof(char), sizeof(char) * Img_Width * Img_Height, Mass);
799        fwrite(Short_Mass, sizeof(short), sizeof(short) * Img_Width * Img_Height, Short);
800        fclose(Mass);
801        fclose(Short);
802        delete[] Img_Mass;
803        delete[] Short_Mass;
804    }
```

Рис.3.5. Код програми для збереження зображення

```
805    private: System::Void button3_Click(System::Object^ sender, System::EventArgs^ e) {
806        //Read Img_Mass from bin file
807        Bitmap^ input = gcnew Bitmap(pictureBox->Image);
808        Bitmap^ output = gcnew Bitmap(input->Width, input->Height);
809        Bitmap^ output_short = gcnew Bitmap(input->Width, input->Height);
810        Img_Height = input->Height;
811        Img_Width = input->Width;
812        Img_Mass = new unsigned char[Img_Height * Img_Width];
813        Short_Mass = new unsigned short[Img_Height * Img_Width];
814        FILE* Mass = fopen("test.bin","rb");
815        FILE* Short = fopen("test_short.bin", "rb");
816        long Size = Img_Width * Img_Height;
817        int c = fread(Img_Mass, sizeof(char), sizeof(char) * Img_Width * Img_Height, Mass);
818        int d = fread(Short_Mass, sizeof(char), sizeof(char) * Img_Width * Img_Height, Mass);
819        for (int x = 0; x < Img_Width; x = x++)
820        {
821            for (int y = 0; y < Img_Height; y++)
822            {
823                output->SetPixel(x, y, Color::FromArgb(255, Img_Mass[x + y * Img_Width], Img_Mass[x + y * Img_Width], Img_Mass[x + y * Img_Width]));
824                output_short->SetPixel(x, y, Color::FromArgb(Short_Mass[x + y * Img_Width]%256, Short_Mass[x + y * Img_Width] / 256, Short_Mass[x + y * Img_Width] / 256, Short_Mass[x + y * Img_Width] / 256));
825            }
826        }
827        pictureBox_processed->Image = output_short;
828        picturebox_sum->Image = output;
829        fclose(Mass);
830        fclose(Short);
831        delete[] Img_Mass;
832        delete[] Short_Mass;
833    }
```

Рис.3.6. Код програми для відкриття зображення

3.3.Порівняння результатів обробки цифрового зображення після запропонованого в дипломі алгоритму з алгоритмом медіанної фільтрації

Перед тим, як почати порівнювати результати обробки зображення різними алгоритмами необхідно це саме зображення отримати. Для початкової перевірки роботи програмного коду було створено однокольорове зображення, числове значення пікселя якого дорівнює 120, на яке було накладено псевдовипадковий шум.

Необхідно зазначити, що кількість пікселів з випадковим числовим значенням є константою. Важливо зауважити, що таким чином ми задаємо середню кількість пікселів на цифрових зображеннях. Тобто, якщо на першому зображенні кількість випадкових пікселів буде дорівнювати 1000, а на другому 800, середнє значення буде 900 і їх різна кількість ніяк не вплине на результати обчислення дисперсії зображення.

На Рис.3.7. показано скріншот цифрового зображення з однаковим числовим значенням пікселя та псевдовипадковим шумом.

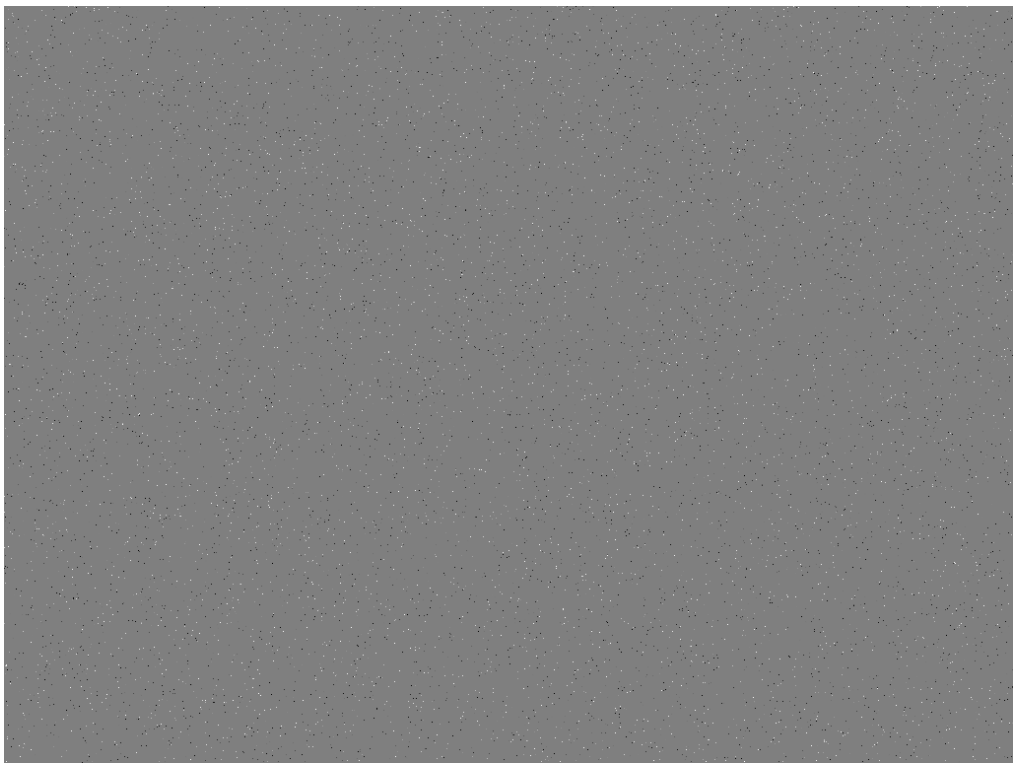


Рис.3.7. Цифрове зображення з псевдовипадковим шумом

Тепер необхідно провести медіанну фільтрацію цифрового зображення, для чого було створено програмний код (Рис.3.8)

```
for (x = 1; x < Img_Width; x = x + 3) {
    for (y = 1; y < Img_Height; y = y + 3) {
        x_cicle = 0;
        max_x = x;
        min_x = x;
        max_y = y;
        min_y = y;
        for (x_window = x - 1;  x_window < x + 2;  x_window++) {
            for (y_window = y - 1;  y_window < y + 2;  y_window++) {
                if ((x_window >= Img_Width) || (y_window >= Img_Height)) {
                    x_window = Img_Width - 1;
                    y_window = Img_Height - 1;
                }
                if (Img_Mass[max_x + max_y * Img_Width] < Img_Mass[x_window + y_window * Img_Width]) {
                    max_x = x_window;
                    max_y = y_window;
                }
                if (Img_Mass[min_x + min_y * Img_Width] > Img_Mass[x_window + y_window * Img_Width]) {
                    min_x = x_window;
                    min_y = y_window;
                }
                Mass_window[x_cicle] = Img_Mass[x_window + y_window * Img_Width];
                x_cicle++;
            }
        }
        cicle_size = 9;
        for (x_cicle = 0; x_cicle  < cicle_size; x_cicle++) {
            for (int x_cicle2 = x_cicle + 1; x_cicle2 < cicle_size; x_cicle2++) {
                cicle_min = x_cicle2;
                if (x_cicle2 >= cicle_size) {
                    break;
                }
                else if (Mass_window[x_cicle2] < Mass_window[cicle_min]) {
                    cicle_min = x_cicle2;
                }
            }
            prom = Mass_window[cicle_min];
            Mass_window[cicle_min] = Mass_window[x_cicle];
            Mass_window[x_cicle] = prom;

        }
        int avarage = 4;
        Img_Mass[max_x + max_y * Img_Width] = Mass_window[avarage];
        Img_Mass[min_x + min_y * Img_Width] = Mass_window[avarage];
```

Рис.3.8. Код для реалізації медіанної фільтрації.

Після чого було проведено фільтрацію даного зображення (Рис.3.9.).

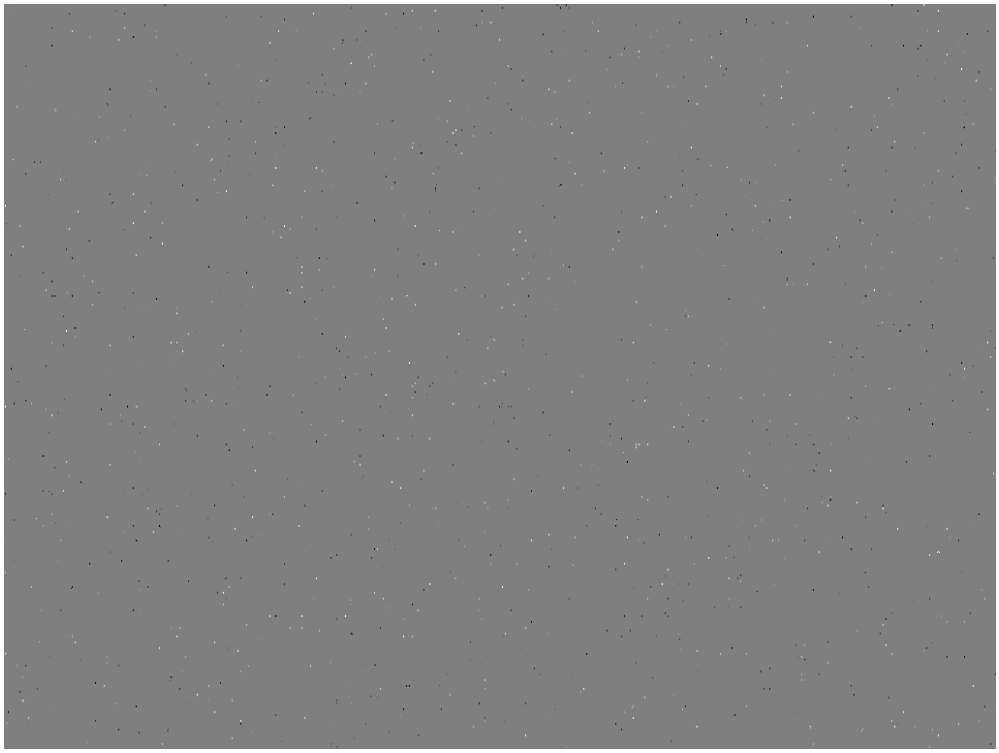


Рис.3.9. Цифрове зображення після використання медіанної фільтрації

На даному скріншоті зображення позбавилось більшості «солі», але випадкові пікселі все одно дуже помітні, що може привести к ускладненню оцінки стану деталі спеціалісту. Необхідно зазначити, що медіанна фільтрація, серед лінійних та нелінійних фільтрів використовується саме для вилучення цифрового шуму з цифрового зображення. В цьому випадку медіанної фільтрації було використано вікно 3×3, яке найбільш якісно позбавляється шуму, а ніж вікно 5×5 або 7×7.

Одразу після цього, для реалізації алгоритму підсумування, було створено ще три цифрових зображення з псевдовипадковим шумом (Рис.А.1, Рис.А.2., Рис.А.3.). Ці чотири зображення Для їх підсумування необхідно відповідні пікселі кожного зображення і поділені на кількість таких зображень (Рис.3.10.).

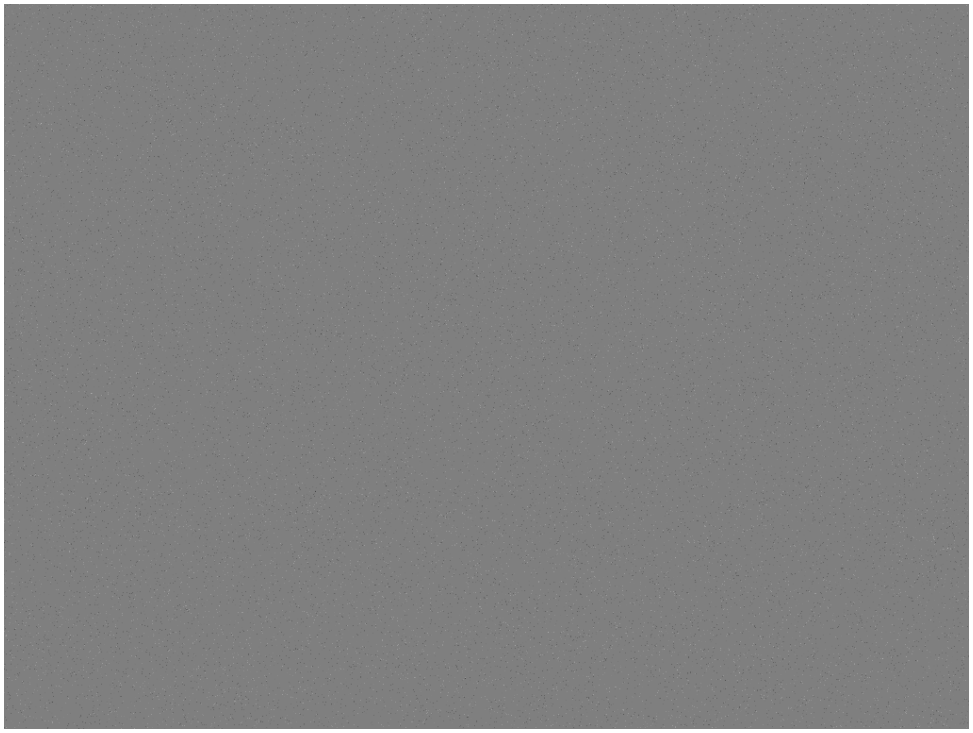


Рис.3.10. Цифрове зображення після використання алгоритму підсумовування

На Рис.3.11. і Рис.3.12 скріншоти коду для дисперсії обробленого зображення та для дисперсії одного з вхідних зображень відповідно.

```
Img_Height = input->Height;
Img_Width = input->Width;
Img_Mass = new unsigned char[Img_Height * Img_Width];
int x;
int y;
int x_min;
int y_min;
float Math_Exp1 = 0;
float Math_Exp2 = 0;
int number = 0;
int counter = 0;
double probab;
for (x = 0; x < Img_Width; x++)
{
    for (y = 0; y < Img_Height; y++)
    {
        UInt32 pixel = (UInt32)(input->GetPixel(x, y).ToArgb());
        float R = (float)((pixel & 0x00FF0000) >> 16);
        float G = (float)((pixel & 0x0000FF00) >> 8);
        float B = (float)(pixel & 0x000000FF);

        Img_Mass[x + y * Img_Width] = R;
    }
}
for (number = 0; number < 256; number++)
{
    for (x = 0; x < Img_Width; x++)
    {
        for (y = 0; y < Img_Height; y++)
        {
            if (Img_Mass[x + y * Img_Width] == number) {
                counter++;
            }
        }
    }
    probab = static_cast<double>(counter) / (Img_Height * Img_Width);
    counter = 0;
    Math_Exp1 = Math_Exp1 + probab * number;
    Math_Exp2 = Math_Exp2 + probab * pow(number,2);
}
Dispersion = Math_Exp2 - pow(Math_Exp1,2);
textBox2->Text = Convert::ToString(Dispersion);
Dispersion = 0;
Math_Exp1 = 0;
Math_Exp2 = 0;
```

Рис.3.11. Код програми для обчислення дисперсії вихідного зображення

```
742     for (x = 0; x < Img_Width; x++)
743     {
744         for (y = 0; y < Img_Height; y++)
745         {
746             UInt32 pixel = (UInt32)(input_photo->GetPixel(x, y).ToArgb());
747             float R = (float)((pixel & 0x00FF0000) >> 16);
748             float G = (float)((pixel & 0x0000FF00) >> 8);
749             float B = (float)(pixel & 0x000000FF);
750
751             Img_Mass[x + y * Img_Width] = R;
752         }
753     }
754     for (number = 0; number < 256; number++)
755     {
756         for (x = 0; x < Img_Width; x++)
757         {
758             for (y = 0; y < Img_Height; y++)
759             {
760                 if (Img_Mass[x + y * Img_Width] == number) {
761                     counter++;
762                 }
763             }
764         }
765         probab = static_cast<double>(counter) / (Img_Height * Img_Width);
766         counter = 0;
767         Math_Exp1 = static_cast<double>(Math_Exp1) + probab * number;
768         Math_Exp2 = static_cast<double>(Math_Exp2) + probab * pow(number, 2);
769     }
770     Dispersion = Math_Exp2 - pow(Math_Exp1, 2);
771     textBox3->Text = Convert::ToString(Dispersion);
772     delete[] Img_Mass;
773 }
```

Рис.3.12. Код програми для обчислення дисперсії вхідного зображення

В результаті, після дії алгоритму було розраховано дисперсії вхідного та обробленого зображення (Рис.3.13):

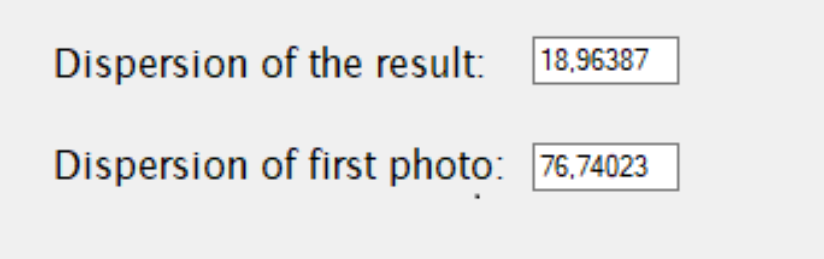


Рис.3.13. Значення дисперсії для обробленого зображення і одного з чотирьох вхідних зображень

Відношення дисперсії вхідного зображення до обробленого дорівнює:

$$\frac{D(x_{\text{вх.}})}{D(x_{\text{об.}})} \sim 4.04665$$

Вже при використанні дев’ятьох цифрових зображень дисперсія буде дорівнювати (Рис.3.14):

$$\frac{D(x_{\text{вх.}})}{D(x_{\text{об.}})} \sim 9{,}0534$$

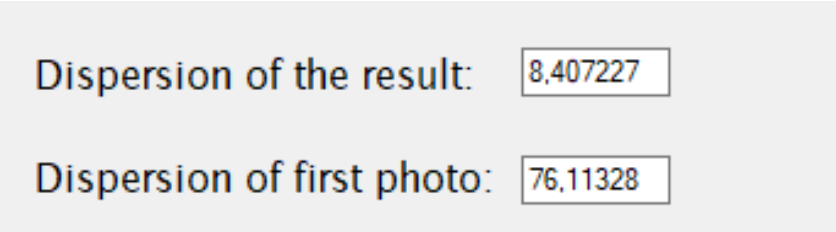


Рис.3.14. Значення дисперсії для обробленого зображення і одного з дев’яти вхідних зображень

Отримані результати дуже близькі до теоретичних, неточність другого знаку після крапки, швидше за всього виникла саме через псевдовипадковий спосіб накладання шуму.

Необхідно зауважити, що дисперсія не є числовим значенням шуму зображення, вона лише відображає відхилення від середнього значення пікселя цифрового зображення. Тому, щоб показати саме числове значення шуму необхідно створити однокольорове зображення та на нього накласти шум. Таким чином дисперсія буде відображати саме шум зображення.

На Рис.3.15. скріншот числового значення дисперсії цифрового зображення після медіанної фільтрації та алгоритму підсумування з використання чотирьох зображень. Дивлячись на вихідні значення дисперсії для дох алгоритмів, можливо зробити висновок, що вже при використанні чотирьох зображень алгоритм підсумування майже так само позбавляється від шуму, як і медіанний фільтр.

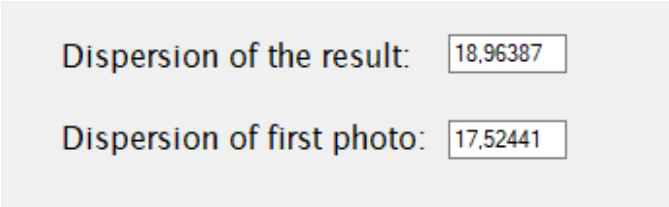


Рис.3.15. Порівняння дисперсії цифрового зображення після обробки медіанним фільтром (Dispersion of first photo) та після використання алгоритму підсумування (Dispersion of result)

3.4.Використання алгоритму для реального зображення

В сучасній радіографії для більш якісного зображення використовують наступний алгоритм покращення зображення. Об’єкт, який буде досліджуватись, розбивається на сегменти. Одне зображення складається с декількох знімків, які були отримані при цифровій рентгенографії одного сегменту досліджуваного об’єкту (Рис.3.16).



Рис.3.16. Цифрове зображення , яке складається з радіографічних знімків одного сегменту.

Після чого отримане зображення розбивається на сегменті, які підсумовуються, таким чином отримують якісне зображення одного з сегментів об’єкту (Рис.3.17.).



Рис.3.17. Оброблене цифрове зображення сегменту.

По такому алгоритму проходить цифрова радіографія всіх сегментів об’єкту і вже після чого ми отримуємо зображення об’єкту.

ВИСНОВКИ

У роботі було

1. Запропоновано та обґрунтовано переваг використання алгоритму підсумування над лінійними та нелінійними алгоритмами фільтрування зображень.
2. Розроблено програмне забезпечення для реалізації запропонованого алгоритму підсумування цифрового зображення.
3. Проведено тестування програми на реальних рентгенівських цифрових знімках.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1.http://www.ntcexpert.ru/85-acenter/953-metody-nerazrushayushchego-kontrolya

2.А.В. Федосов, Л.А. Гайнуллина, Метод неразрушающего контроля – Электротехнические и информационные комплексы и системы – 2015 – №2, т. 11 – С. 73-74

3.В. А. Троицкий, С. Р. Михайлов, Р. О. Пастовенский, Д. С. Шило, Современные системы радиационного неразрушающего контроля – Техническая диагностика и не разрушимый контроль – № 1 – 2015 – С.23-24

4.А.Д. Жуков, Д.И. Галкин, У.А. Иванайский, К вопросу о замене радиографической пленки на цифровые детекторы – Вестник МГТУ им. Н.Э. Баумана. Сер. Машиностроение – 2018 – № 1 – С. 47

5.В. А. Троицкий, Краткое пособие по контролю качества сварных соединений – Киев – Феникс – 2006 – С 3-6

6.А.А. Майоров, Компьютерная радиография с использованием флуоресцентных запоминающих пластин – что это такое? – В мире НК – 2004. – № 3 (25) – С. 42–43

7.L.Yu. Mogilner, E.N. Markina, Experience of Radiography Digital Technology in Practice of Construction and Operation of Pipeline – NDT World Review 2 (44) – June 2009 – С 42-46

8.В.Ю. Гусев, А.В. Крапивенко, Методика фильтрации периодических помех цифровых изображений – Электронный журнал «Труды МАИ» – Выпуск – № 50 – С. 1–7.

9.Л.Р. Юнусова, А.Р. Магсумова, Фильтрация шумов – Научный журнал – 2020 – С. 41-42

10.Н.Н. Бондина, А.С. Калмычков, В.Э. Кривенцов, Сравнительный анализ алгоритмов фильтрации медицинских изображени – Вестник НТУ «ХПИ» – 2012 – № 38 – С. 14-24.

11.А.А. Майоров, Цифровые технологии в радиационном контроле – «В мире НК » – № 1(35) – 2007 – с.5-12

12.Non-destructive testing. Image quality of radiographs. Part 5. Determination of the image unsharpness value using duplex wire-type image quality indicators. – ISO 19232-5 – 2013 – 3 p.

13.U. Ewert, U. Zscherpel, K Bavendie, Replacement of film radiography by digital techniques and enhancement of image quality – Berlin – BAM Federal Institute for Materials Research and Training – 2007 – 14 p.

14.Ю.П. Михайлюк, Д.В. Начаров, Метод улучшения различимости объектов на цифровых изображениях, полученных в условиях недостаточной видимости – «Журнал радиоэлектроники» №6 – 2015

15.Р. Гонзалес, Р. Вудс, Цифровая обработка изображений – М.Техносфера, 2006. – С. 1072

16.Н.Н. Бондина, А.С. Калмычков, В.Э. Кривенцов, Сравнение алгоритмов фильтрации медицинских изображений – Вестник НТУ "ХПИ" – 2012 – № 38 (1012) – С. 14-25

17.Н.Н. Бондина, Р.Ю. Муратов, Адаптивные алгоритмы фильтрации и изменения контраста изображения – Вестник НТУ "ХПИ" – 2014 – № 35 (1078) – С. 35 – 41

18.В. В. Стругайло, Обзор методов фильтрации и сегментации цифровых изображений – ФГБОУ ВПО "МГТУ им. Н.Э. Баумана". Эл № ФС 77 – 48211 – С. 270-278

19.Н.Н. Бондина, А.С. Калмычков, О.А. Козина, Сравнение алгоритмов фильтрации медицинских изображений по оценкам их качества – Вестник НТУ "ХПИ" – 2013 – № 39 (1012) – С. 15 – 21

Додаток А

Цифрові зображення с шумом

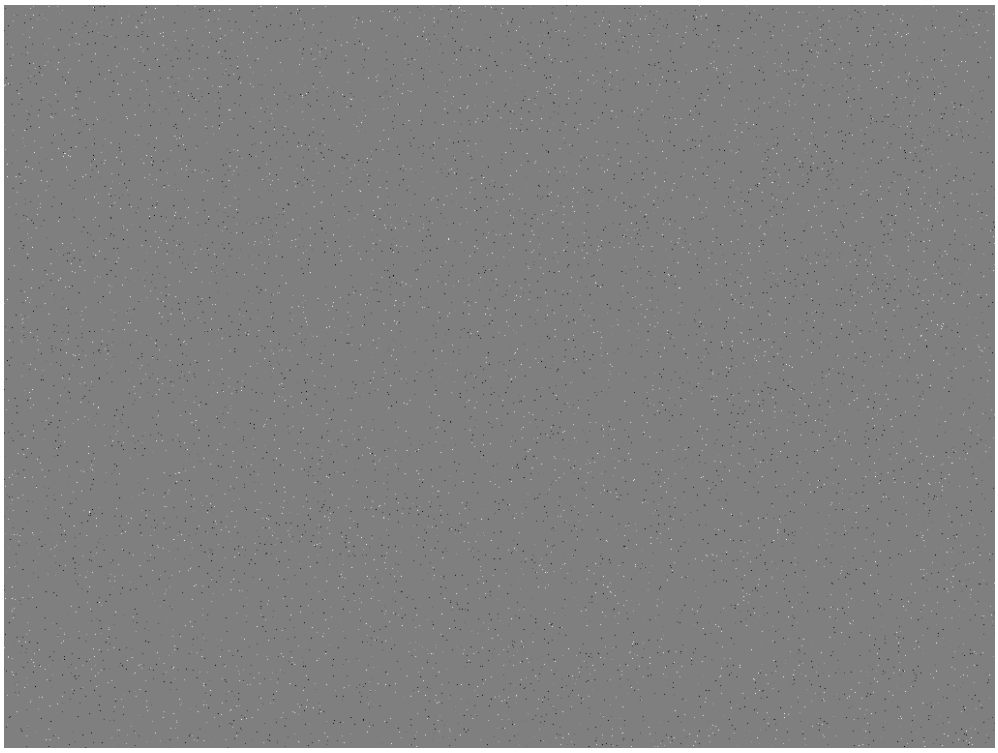


Рис.А.1. Цифрове зображення с шумом

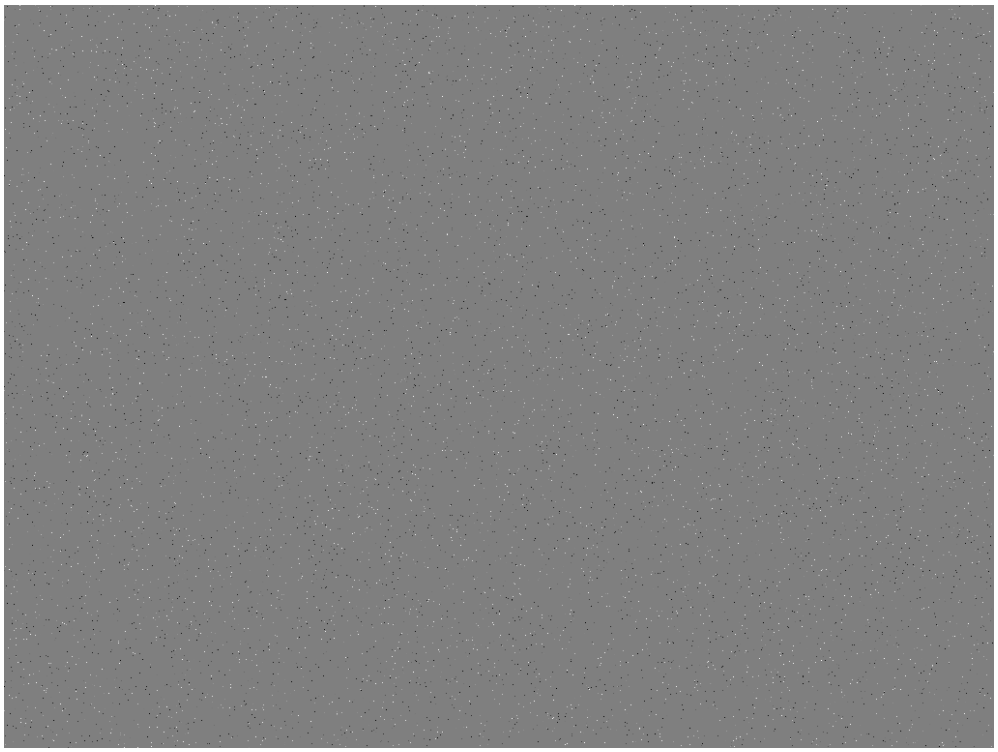


Рис.А.2. Цифрове зображення с шумом

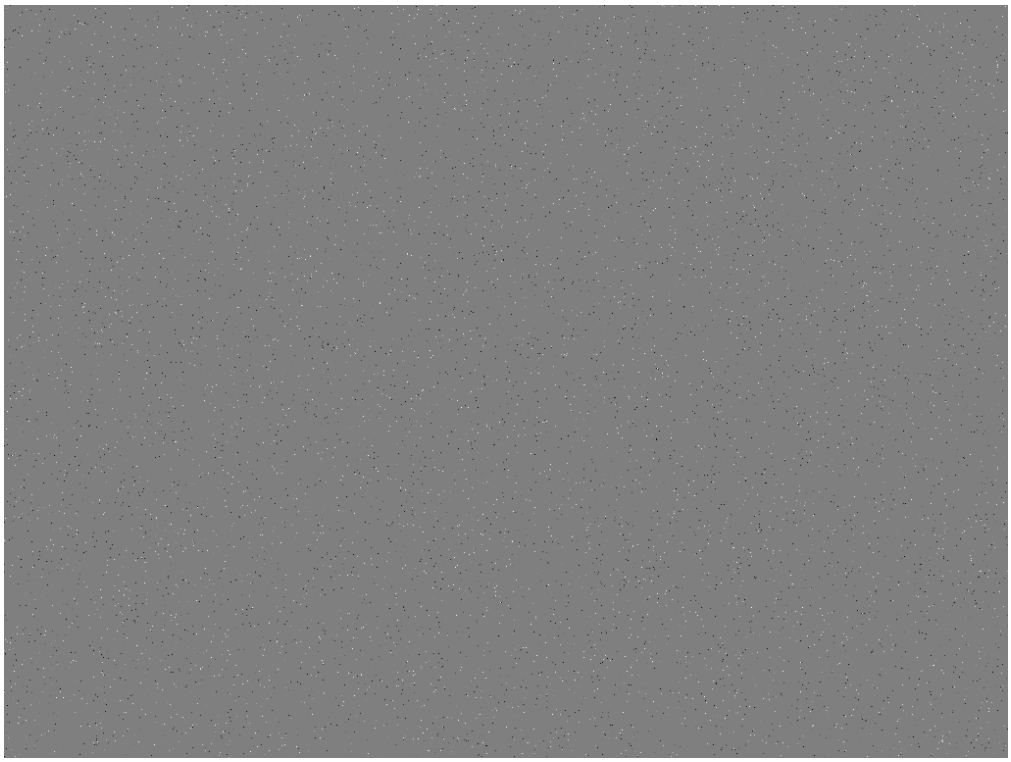


Рис.А.3. Цифрове зображення с шумом