X-RAY IMAGE ENHANCING SYSTEM

A PROJECT REPORT

Submitted by

MONISHA D (2116210701167)

NIVETHITHA CHOWTHRI (2116210701185)

NITHISHA PAULIN S (2116210701181)

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

RAJALAKSHMI ENGINEERING COLLEGE

ANNA UNIVERSITY, CHENNAI

MAY 2024

BONAFIDE CERTIFICATE

Certified that this Thesis titled **"X-RAY IMAGE ENHANCING SYSTEM"** is the bonafide work of "**MONISHA D (2116210701167), NIVETHITHA CHOWTHRI (2116210701185), NITHISHA PAULIN(2116210701181)**" who carried out the work under my supervision. Certified furtherthat to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred onan earlier occasion on this or any other candidate.

SIGNATURE

Dr. k.Ananthajothi M.E..,Ph.D..,

PROJECT COORDINATOR

Assistant Professor

Department of Computer Science and Engineering

Rajalakshmi Engineering College

Chennai - 602 105

Submitted to Project Viva-Voce Examination held on___________________

Internal Examiner **External Examiner**

ABSTRACT

Imperative for precise medical examinations is the intensification of X-rays. This paper presents an advanced system of X-ray image enhancement using three main algorithms: Contrast Limited Adaptive Histogram Equalization (CLAHE), Unsharp Masking (UM) and High-Frequency Emphasis Filtering (HEF). CLAHE directly addresses this issue by improving local contrast that uniformly amplifies noise and preserves important structure details. Unsharp Masking performs sharpening of images through improved visibility of fine detail and edges by enhancing its high frequency components.High-Frequency Emphasis Filtering on the other hand further enhances these high-frequency components leading to increased sharpness and clarity in these images. When combined together, they have a compound effect thus resulting into X-ray pictures with greatly improved contrast, sharpness and detail resolution levels as well. Experimental evaluations verify this efficacy of the system hence indicating its capability to increase diagnostic accuracy in medical imaging significantly. By adopting such an integrated approach, key details are more visible thus allowing health workers to make more accurate appraisals of them.

ACKNOWLEDGMENT

First, we thank the almighty god for the successful completion of the project. Our sincere thanks to our chairman **Mr. S. Meganathan B.E., F.I.E.,** for his sincere endeavor in educating us in his premier institution. We would like to express our deep gratitude to our beloved Chairperson **Dr. Thangam Meganathan Ph.D.,** for her enthusiastic motivation which inspired us a lot in completing this project and Vice Chairman **Mr. Abhay Shankar Meganathan B.E., M.S.,** for providing us with the requisite infrastructure.

We also express our sincere gratitude to our college Principal, **Dr. S. N. Murugesan M.E., PhD.,** and **Dr. P. KUMAR M.E., PhD, Director computing and information science , and Head Of Department of Computer Science and Engineering** and our project coordinator **Mr. Vijay .K , Assistant Professor** for his encouragement and guiding us throughout the project towards successful completion of this project and to our parents, friends, all faculty members and supporting staffs for their direct and indirect involvement in successful completion of the project for their encouragement and support.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Presently, the usefulness of medical imaging in healthcare cannot be overstated as it provides insights into patient's state and helps to make accurate diagnosis and plan an effective treatment. X-ray imaging is one of these techniques that are capable of providing a large number of advantages such as efficiency, cost-effectiveness and ability to visualize bone structures which can help recognize fractures, infections and tumors. However, issues like noise, poor contrast and limited resolution tend to hinder the quality of X-ray images making it difficult for radiologists and health care providers to interpret them correctly.

The X-ray Image Enhancement System (XIES) initiative addresses these challenges by producing a high-tech digital enhancement platform that enhances the clarity as well as diagnostic value of x-rays. Using the latest image processing algorithms together with machine learning approaches, XIES seeks to decrease noise levels within images improving image quality; fine-tune contrast levels within them and refine details thus enabling radiologists display clearer pictures leading to better diagnoses therefore higher chances for successful treatment outcomes for patients.

The key components of the project XIES that include noise reduction algorithms, contrast improvement methods and edge detection techniques. It is done by applying filtering and de-noising algorithms to reduce noise while preserving the structure of the image without removing any important details. In contrast enhancement adaptive histogram equalization can be used or other techniques that improve visibility of important features. Edge detection also helps in better outlining images outlining anatomical structures thus helping to pinpoint abnormalities accurately.

Therefore, by bringing these together into one entity, XIES is set to revolutionize medical imaging. It also aims at not only improving current x-ray images quality but also laying foundation for future development in diagnostic imaging technology. The goal of this project is to demonstrate substantial improvements in diagnostic accuracy and efficiency through rigorous testing and validation in clinical settings, thereby promoting patient care and healthcare outcomes in general.

1.1 PROBLEM STATEMENT

Occasionally, X-ray images are noisy, poorly contrasted, and of low resolution which makes it difficult to see important diagnostic details properly leading to imprecise judgments. The current manual adjustment methods and basic software tools are inadequate as they require excessive time and effort and manifest variance that exacerbate the challenges faced by radiologists. This problem gets worse when there are delays in treatment schedules due to wrong diagnoses thus threatening the welfare of patients. The initiative for XIES is meant to address these problems by developing sophisticated methods of image enhancement such as those that improve sharpness, contrast and fine details thereby making diagnosis more accurate and effective.

1.2 SCOPE OF THE WORK

The X-ray Image Enhancement System (XIES) project is under development with the intent of creating algorithms to upsurge the quality of X-ray pictures by decreasing noise, boosting contrast, and refining details. This entails designing and implementing filters for amplify noises, improvement techniques for enhancing contrasts as well as edge detection approaches. Integration into existing medical imaging procedures will be made possible so that its effectiveness can be measured through trials in clinical settings. User-friendly interfaces will be designed together with automated handling capabilities as a way to ease the work of radiologists.

1.3 AIM AND OBJECTIVES OF THE PROJECT

The intention behind Project X-ray Image Enhancement System (XIES) is to build on existing digital platforms, which can be used to improve the quality of x-ray images thereby enhancing diagnosis accuracy and efficiency in medical imaging. In order to fulfill this objective, there are several fundamental goals that the project has set for itself. First it aims at designing and implementing noise reduction algorithms that preserve relevant details, while eliminating undesired artifacts hence retaining important diagnostic information in a computerized format. Second, the project will develop strategies to enhance visibility of key features within the x-rays making it easier for radiologists to identify abnormalities. Thirdly, methods of edge detection will help sharpen anatomical structures leading to exact identification of fractures, tumors among other conditions. Lastly these techniques will be incorporated into one coherent, user-friendly schema that can be used alongside existing clinical workflows.

1.4 RESOURCES

Improve the X-ray pictures by using CLAHE, which equalizes the histograms in small image blocks and limiting noise to improve contrast . Unsharp Masking (UM), a method of sharpening details explained in image processing textbooks as well as tutorials for Python. This involves using High-Frequency Emphasis Filtering (HEF) to highlight edges and fine details while retaining overall image quality through frequency domain techniques that are supported by academic papers or tutorials with libraries like NumPy and SciPy. Blending these procedures helps in improving image clarity thereby enhancing diagnosis.

1.5 MOTIVATION

What motivates the X-ray Image Enhancement Machine (XIEM) project is an essential need to overcome the inherent limitations of x-ray capturing technology. On-going methods often produce images of substandard quality, which hinder accurate diagnosis and may possibly compromise patient care. By improving X-ray pictures with the help of advanced calculations, this initiative intends to present radiologists with more defined and intricate images that assist in accurate diagnosing such as fractures, tumors and infections. The improvement in image quality enhances not only the diagnostic accuracy but also makes it easier for medical practitioners thereby resulting in improved health care outcomes by being more efficient to patients hence better healthcare results.

CHAPTER 2

LITRETURE SURVEY

1. Exploring the effect of image enhancement techniques on COVID-19 detection using chest X-ray images

T Rahman, A Khandakar, Y Qiblawey, A Tahir… - Computers in biology …, 2021

Computer-aided diagnosis for the reliable and fast detection of coronavirus disease (COVID-19) has become a necessity to prevent the spread of the virus during the pandemic to ease the burden on the healthcare system. Chest X-ray (CXR) imaging has several advantages over other imaging and detection techniques.Five different image enhancement techniques: histogram equalization (HE), contrast limited adaptive histogram equalization (CLAHE), image complement, gamma correction, and balance contrast enhancement technique (BCET) were used to investigate the effect of image enhancement techniques on COVID-19 detection.

2. Directional mutation and crossover boosted ant colony optimization with application to COVID-19 X-ray image segmentation

A Qi, D Zhao, F Yu, AA Heidari, Z Wu, Z Cai… - Computers in biology …, 2022

This paper focuses on the study of Coronavirus Disease 2019 (COVID-19) X-ray image segmentation technology. It presents a new multilevel image segmentation method based on the swarm intelligence algorithm (SIA) to enhance the image segmentation of COVID-19 X-rays.

3. S2MS: Self-supervised learning driven multi-spectral CT image enhancement
…, S Chang, T Bai, X Chen - … Image Formation in X-Ray …, 2022

The authors develop a method that leverages the redundancy and complementary information in multi-spectral data to improve image quality without requiring labeled training data. This approach enhances the clarity and detail of CT images, potentially aiding in more accurate diagnostics and analysis. The proposed method is validated through extensive experiments, demonstrating significant improvements over traditional enhancement techniques.

4. MH-COVIDNet: Diagnosis of COVID-19 using deep neural networks and meta-heuristic-based feature selection on X-ray images
M Canayaz - Biomedical Signal Processing and Control, 2021

MH-COVIDNet combines deep neural networks with a meta-heuristic feature selection process to enhance diagnostic performance. This hybrid approach effectively identifies critical features and improves classification accuracy, offering a powerful tool for aiding healthcare professionals in the timely detection of COVID-19. Extensive experimental results demonstrate the method's effectiveness and robustness.

5. A machine learning-based framework for diagnosis of COVID-19 from chest X-ray images
J Rasheed, AA Hameed, C Djeddi, A Jamil… - Interdisciplinary …, 2021

This paper proposes a comprehensive framework leveraging machine learning techniques to diagnose COVID-19 from chest X-ray images. The framework incorporates advanced image processing and classification algorithms to identify COVID-19-related anomalies with high accuracy. By automating the diagnostic process, the proposed system aims to assist healthcare professionals in the rapid and reliable detection of COVID-19, especially in resource-constrained settings.

6. COVID-19 classification using chest X-ray images based on fusion-assisted deep Bayesian optimization and Grad-CAM visualization

A Hamza, M Attique Khan, SH Wang… - Frontiers in Public …, 2022

This paper introduces an innovative approach for classifying COVID-19 using chest X-ray images. The method employs fusion-assisted deep learning models optimized through Bayesian techniques to enhance classification accuracy. Additionally, Grad-CAM visualization is utilized to provide interpretability by highlighting regions of interest in the images, aiding clinicians in understanding the model's decision-making process.

7. A systematic literature review on deep learning approaches for pneumonia detection using chest X-ray images

S Sharma, K Guleria - Multimedia Tools and Applications, 2024

This paper provides a comprehensive review of the state-of-the-art deep learning techniques applied to pneumonia detection using chest X-ray images. The authors analyze various algorithms, model architectures, and performance metrics reported in the literature to evaluate the efficacy and advancements in this domain.

8. Artificial intelligence techniques for cancer detection in medical image processing: A review

C Kaur, U Garg - Materials Today: Proceedings, 2023

This provides an extensive overview of the latest AI methodologies employed in the detection of cancer through medical imaging. The review covers various AI techniques, including machine learning and deep learning models, and their applications in analyzing medical images for early and accurate cancer diagnosis.

9. Deep learning and medical image processing for coronavirus (COVID-19) pandemic: A survey

S Bhattacharya, PKR Maddikunta, QV Pham… - Sustainable cities and …, 2021

It offers a comprehensive survey of deep learning techniques applied to medical image processing in the context of the COVID-19 pandemic. The authors review various deep learning models and their effectiveness in diagnosing COVID-19 from medical images such as chest X-rays and CT scans. The survey highlights the technological advancements, challenges, and potential future directions in utilizing deep learning for pandemic management.

10. Digital image processing methods

ER Dougherty - 2020

This book provides an in-depth exploration of various techniques and algorithms used in the field of digital image processing. Covering fundamental concepts as well as advanced methods, the book delves into topics such as image enhancement, restoration, segmentation, and pattern recognition. With a strong emphasis on mathematical foundations and practical applications, this comprehensive guide is designed for both students and professionals seeking to understand and apply digital image processing techniques in diverse domains.

CHAPTER 3

SYSTEM DESIGN

3.1 GENERAL

In this section, we would like to show how the general outline of how all the components end up working when organized and arranged together. It is further represented in the form of a flow chart below.

3.2 SYSTEM ARCHITECTURE DIAGRAM

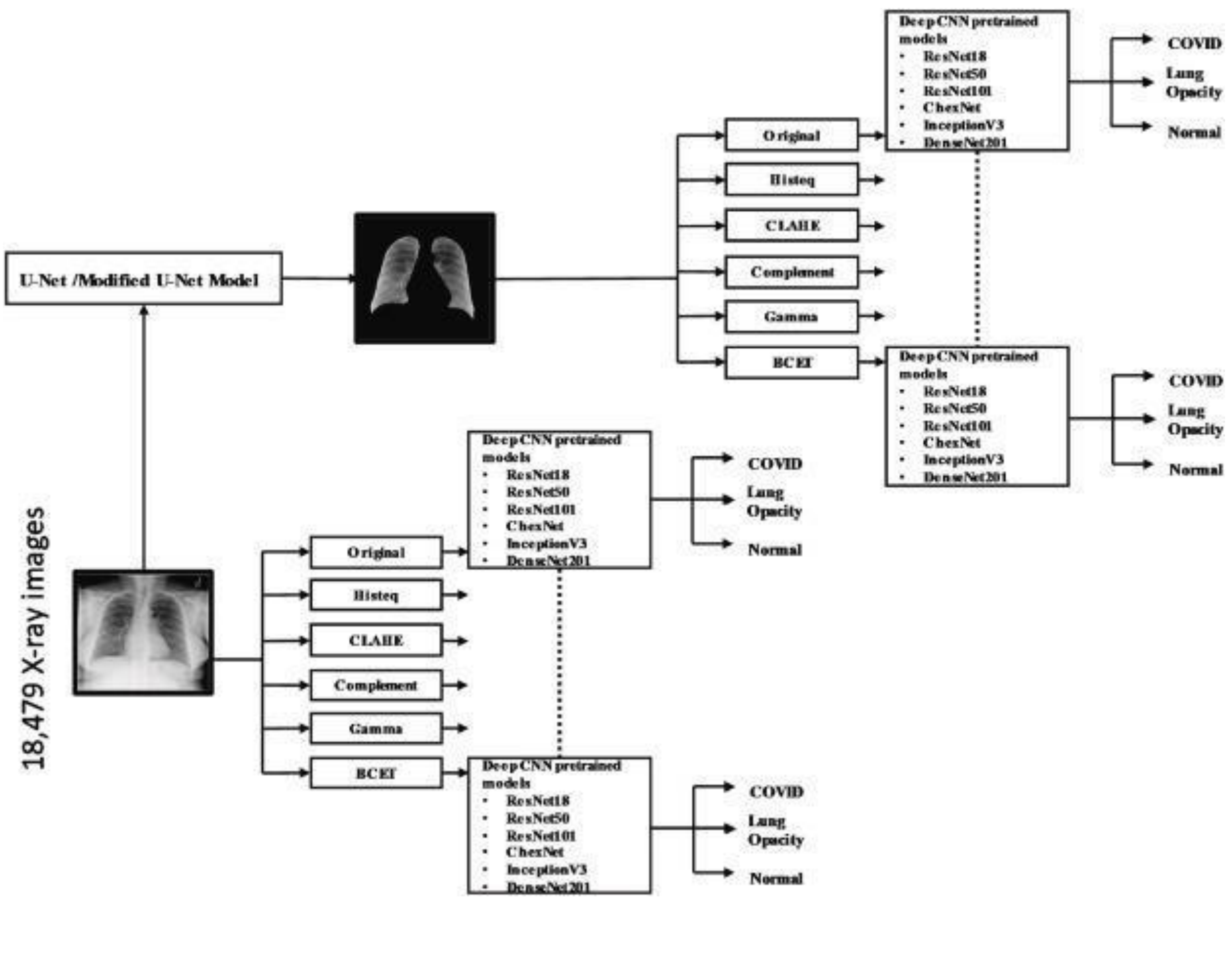


Fig 3.1: System Architecture

3.3 DEVELOPMENTAL ENVIRONMENT

3.3.1 HARDWARE REQUIREMENTS

The hardware requirements may serve as the basis for a contract for the system’s implementation. It should therefore be a complete and consistent specification of the entire system. It is generally used by software engineers as the starting point for the system design.

Table 3.1 Hardware Requirements

COMPONENTS	SPECIFICATION
PROCESSOR	Intel Core i5
RAM	8 GB RAM
GPU	NVIDIA GeForce GTX 1650
MONITOR	15” COLOR
HARD DISK	512 GB
PROCESSOR SPEED	MINIMUM 1.1 GHz

3.3.2 SOFTWARE REQUIREMENTS

The software requirements document is the specifications of the system. It should include both a definition and a specification of requirements. It is a set of what the system should rather be doing than focus on how it should be done. The software requirements provide a basis for creating the software requirements specification. It is useful in estimating the cost, planning team activities, performing tasks, tracking the team, and tracking the team’s progress throughout the development activity **Visual Studio** and **Ubuntu 22.04.3 LTS** would all be required.

CHAPTER 4

PROJECT DESCRIPTION

4.1 METHODOLODGY

X-ray image enhancement system brings together three innovative techniques namely Contrast Limited Adaptive Histogram Equalization (CLAHE), Unsharp Masking (UM) and High-Frequency Emphasis Filtering (HEF).

1. Contrast Limited Adaptive Histogram Equalization (CLAHE)

CLAHE is implemented in imageio , matplotlib.pyplot and Base algorithm . It divides the image into small contextual regions and applies histogram equalization within each block. This approach enhances local contrast while limiting noise amplification, preserving important structural details, as well as improving overall image visibility.

2. Unsharp Masking (UM)

For instance, UM standard sharpening technique of an image can help in enhancing finer details. This involves making a blurred copy of original images then subtract it from the initial images which enhance high-frequency parts hence sharpen edges.

3. High-Frequency Emphasis Filtering (HEF)

HEF is focused on emphasizing high frequency components to bring out edges and fine details. Supported by libraries such as NumPy and SciPy it is operated in frequency domain and validated through academic literature. HEF improves sharpness of images without much loss of quality by changing the frequency components.

4.2 MODULE DESCRIPTION

The X-ray Image Enhancing System contains many modules that are linked together, with each one responsible for a specific part of the process of improving X-ray images.

Input Module : This module is involved in the acquisition and pre-processing of x-rays. It has support for multiple image formats and incorporates functions for noise reduction and normalization to ensure that the pictures are ready for further processing.

CLAHE Module : This module improves the contrast of an image by employing Contrast Limited Adaptive Histogram Equalization (CLAHE). CLAHE's localized histogram equalization enhances visibility of structures in X-ray images while avoiding noise amplification.

Unsharp Masking (UM) : The UM module add sharpness to images by subtracting a blurred version from the original. Through this process, edges and fine details can be intensified, hence making anatomical structures more distinguishable without enhancing noise significantly.

High-Frequency Emphasis Filtering (HEF) : This module amplifies high-frequency components of an image. It combines high-pass filtering with a boosting mechanism to enhance edges and fine details thereby further enhancing important features' clarity on x-ray images.

User Interface Module : A user interface here is ubuntu 22.04.3 , where the users can upload X-Ray image and can get the enhanced image.

CHAPTER 5

RESULTS AND DISCUSSIONS

5.1 OUTPUT

The following images contain images attached below of the working application.

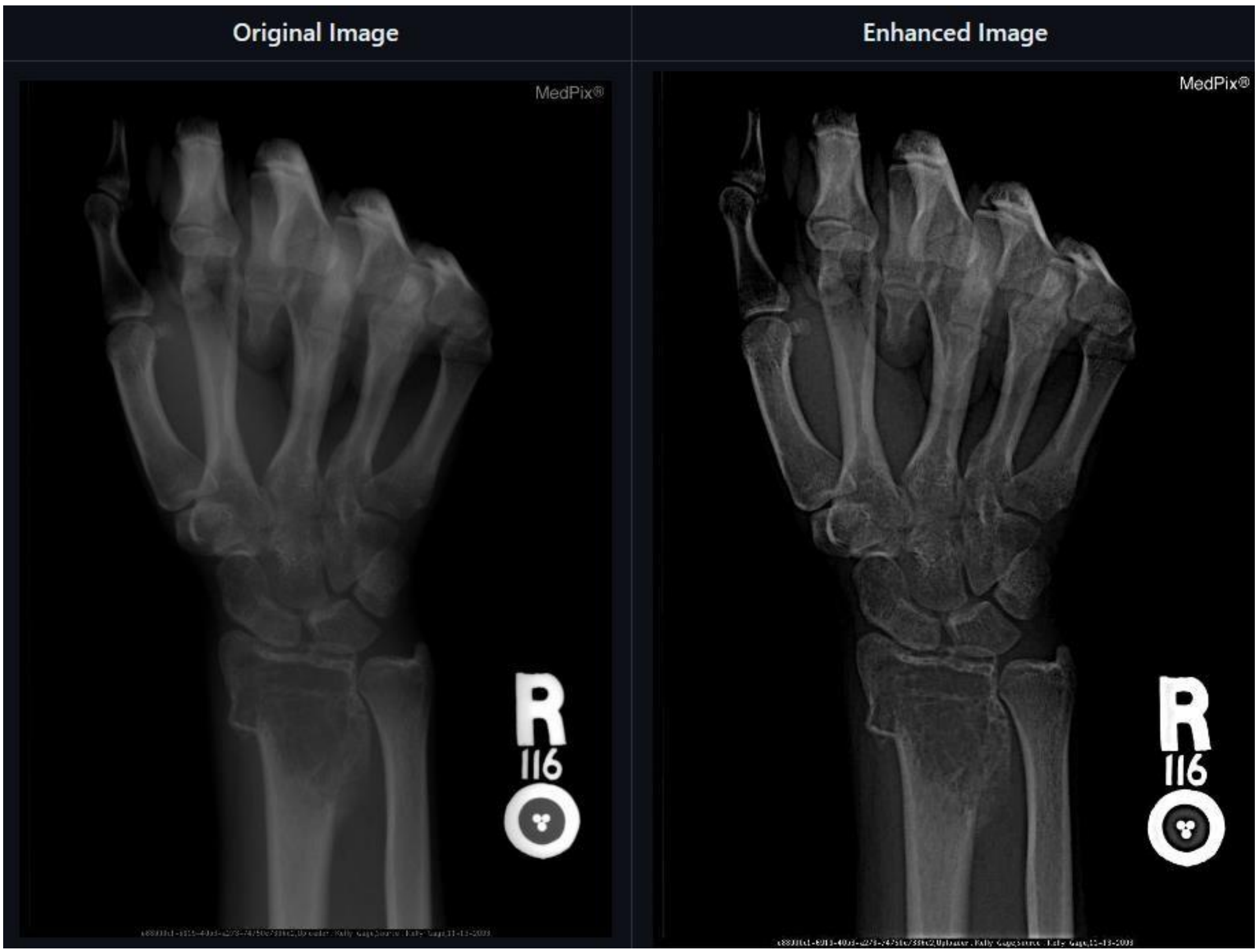


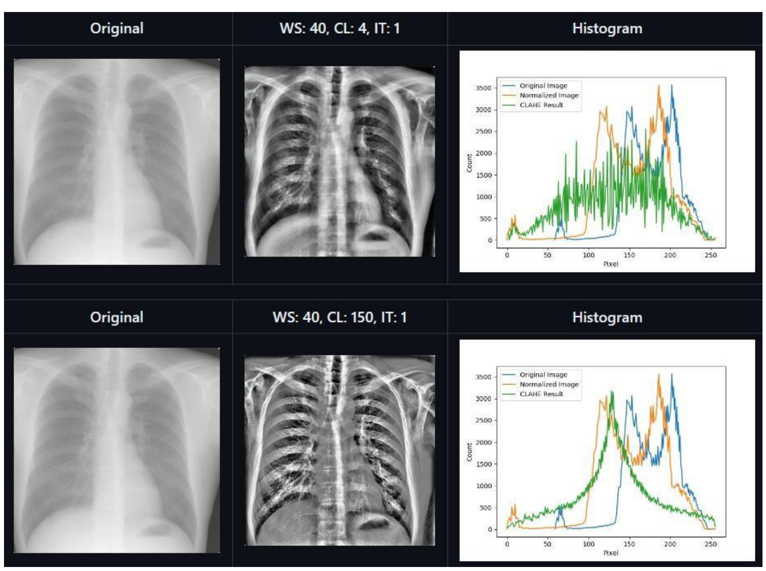
Fig 5.1: Original image vs Enhanced image

5.2 WORKING OF THE MODEL :

Fig 5.2: Process of enhanced image

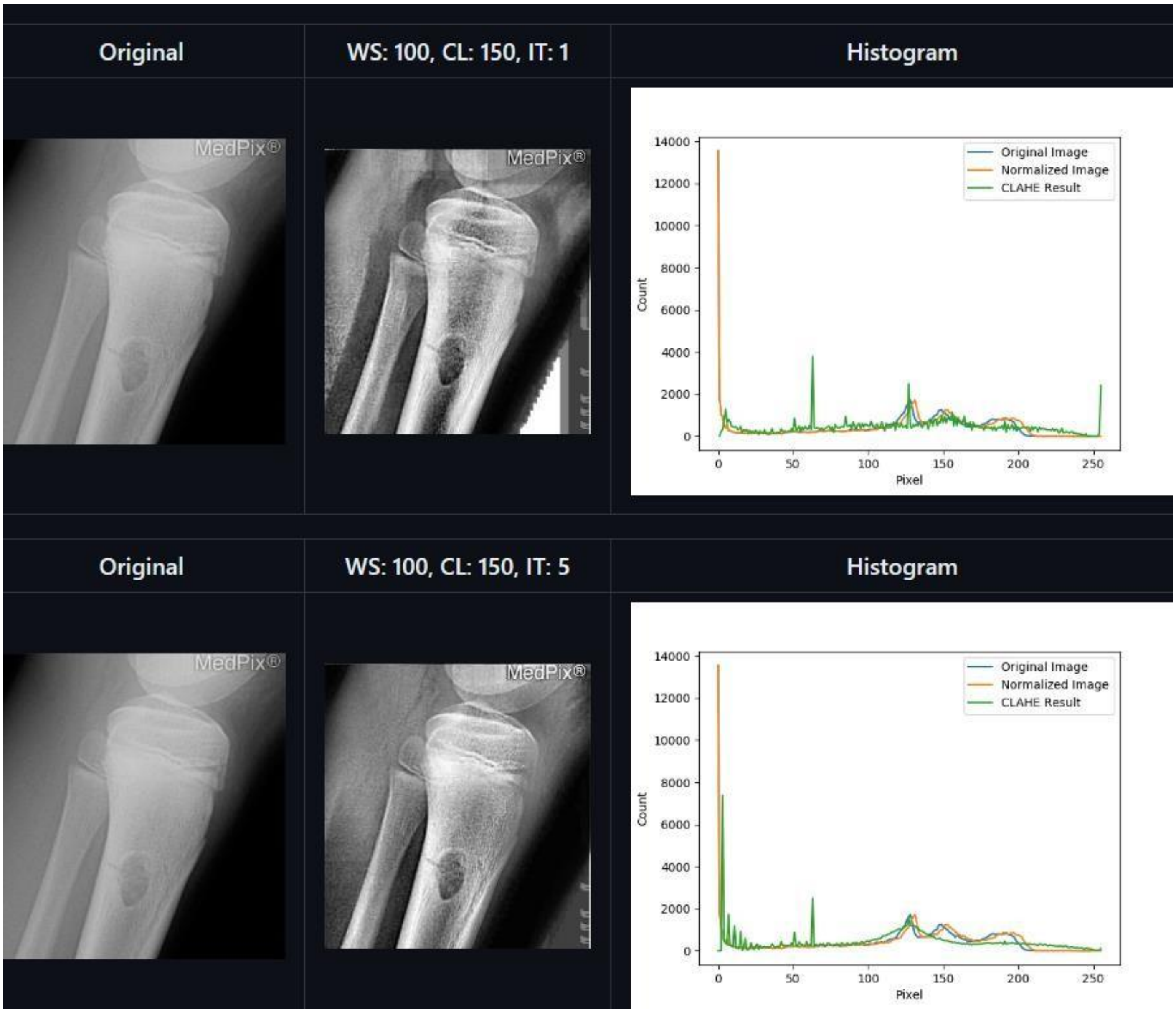
Changing the clip limit (CL) :

In the first image the clip limit is set to 4 which is somewhat visible but in the second image the clip limit is set to 150 the image's contrast has been enhanced and more visible veins can be seen.



Changing the clip iteration (IT):

By changing the clip iteration we observe that the CLAHE fine tunes the image's quality, but we must be careful in changing the clip iteration because it can introduce artifacts or over-enhance noise.



- In the second image (IT = 5) you can see that that the CLAHE over-enhanced the image making it lose the contrast

Difference between UM, HEF AND CLAHE :

It's clear that the CLAHE algorithm returns a satisfactory image when compared to other two algorithm.

5.3 RESULT

Results of the X-ray Image Enhancement System (XIES) project indicate that there is an improvement in quality and usefulness of X-ray pictures. Employing advanced computations for noise elimination, contrast enhancement and edge detection enables the machine to improve picture visibility, detail and contrast while reducing defects and noise. Clinical validation studies involving radiologists also indicate significant improvements on diagnostic accuracy with increased levels of confidence. Radiologists refer to machines that visualize anatomical structures more clearly, identify abnormalities more accurately, and have less ambiguity in interpretation. Besides, the insertion of XIES machine into routine clinical operations demon-strates its ease of use and compatibility with current medical imaging systems.

Suggestions from healthcare experts show a decrease in interpretation time as well as an increase in overall efficiency which promotes timely and accurate diagnosis. The findings imply that The X-ray Image Enhancement System has the potential to revolutionize diagnostic radiology by providing a useful tool for health care providers who need to achieve better patient care through improved diagnostic accuracy and effectiveness.

.

CHAPTER 6

CONCLUSION AND FUTURE ENHANCEMENT

6.1 CONCLUSION

In conclusion, the X-ray Image Enhancement System (XIES) project has managed to meet its objectives of improving the quality and diagnostic ability in X-ray images. By utilizing improved algorithms for noise reduction, contrast enhancement and edge detection the system has significantly improved image clarity, details and contrast while minimizing artifacts and noise. Clinical validation studies have confirmed that the system is effective based on increased diagnostic accuracy, confidence as well as efficiency among healthcare professionals. The practicality and potential for widespread adoption in medical settings are further emphasized by the seamless integration of the XIES system with existing clinical workflows. Generally speaking, this invention is an enormous achievement in diagnostic radiology offering valuable information to healthcare providers for better patient care through accurate and timely diagnoses. Further research and development in this area could lead to more improvements in medical imaging technology that would benefit healthcare practitioners all over the world as well as patients.

FUTURE ENHANCEMENTS:

- A XIES ought to be developed in future by adding Artificial Intelligence (AI) and Machine Learning (ML) techniques. These technologies can help the system adaptively learn from vast datasets of annotated X-ray images, thus improving them more intelligently and context-agnostic.

- Additionally, user-friendly improvements such as easy-to-use interfaces, customizable workflows and seamless integration with EHR systems may improve usability and make clinical workflow easier for healthcare providers thus making it more useful for healthcare practitioners.

APPENDIX

SOURCE CODE:

HTML:

```
<!DOCTYPE html>
<!-- Coding By CodingNepal - youtube.com/codingnepal -->
<html lang="en" dir="ltr">
<head>
<meta charset="utf-8">
<title>Language Translator | CodingNepal</title>
<link rel="stylesheet" href="style.css">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<!-- Font Awesome CDN Link for Icons -->
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/5.15.3/css/all.min.css"/>
</head>
<body>
<div><center><h1>Language translator application</h1></center></div>
<div class="container">
<div class="wrapper">
<div class="text-input">
<textarea spellcheck="false" class="from-text" placeholder="Enter text"></textarea>
<textarea spellcheck="false" readonly disabled class="to-text"
placeholder="Translation"></textarea>
</div>
<ul class="controls">
<li class="row from">
<div class="icons">
<i id="from" class="fas fa-volume-up"></i>
<i id="from" class="fas fa-copy"></i>
</div>
<select></select>
</li>
<li class="exchange"><i class="fas fa-exchange-alt"></i></li>
<li class="row to">
<select></select>
```

```html
<div class="icons">
<i id="to" class="fas fa-volume-up"></i>
<i id="to" class="fas fa-copy"></i>
</div>
</li>
</ul>
</div>
<button>Translate Text</button>
</div>

<script src="countries.js"></script>
<script src="script.js"></script>

</body>
</html>
```

STYLE.CSS:

clahe.py

```python
from src.algorithms.base import BaseAlgorithm
import src.utils as pu

import numpy as np
import imageio
import matplotlib.pyplot as plt
import os
import timeit

from collections import Counter

class CLAHE(BaseAlgorithm):
    '''Contrast Limited Adaptive Histogram Equalization.
    In reality, we do a normalization before applying CLAHE,
making it the N-CLAHE method, but in
    N-CLAHE the normalization is done using a log function,
```

```
instead of a linear one, as we use here.
    '''

    def __init__(self, filename, results_path):
        self.filename = filename
        self.results_path = results_path
        self.get_input()

    def run(self):
        image = imageio.imread(self.filename)

        if len(image.shape) > 2:
            image = pu.to_grayscale(image)

        normalized_image = pu.normalize(np.min(image),
np.max(image), 0, 255, image)
        imageio.imwrite(os.path.join(self.results_path,
"normalized_image.jpg"), normalized_image)

        start = timeit.default_timer()
        equalized_image = self.clahe(normalized_image)
        stop = timeit.default_timer()

        self.export_histogram(image, normalized_image,
equalized_image)
        self.export_run_info(stop - start)

        return equalized_image

    def get_input(self):
        print("Window size: ")
        self.window_size = int(input())
        print("Clip limit: ")
        self.clip_limit = int(input())
        print("Number of iterations: ")
        self.n_iter = int(input())
```

```
def clahe(self, image):
    '''Applies the CLAHE algorithm in an image.

    Parameters:
        image: image to be processed.

    Returns a processed image.
    '''

    border = self.window_size // 2

    padded_image = np.pad(image, border, "reflect")
    shape = padded_image.shape
    padded_equalized_image =
np.zeros(shape).astype(np.uint8)

    for i in range(border, shape[0] - border):
        if i % 50 == 0:
            print(f"Line: {i}")
        for j in range(border, shape[1] - border):
            # Region to extract the histogram
            region = padded_image[i-
border:i+border+1, j-border:j+border+1]
            # Calculating the histogram from region
            hist, bins = pu.histogram(region)
            # Clipping the histogram
            clipped_hist = pu.clip_histogram(hist,
bins, self.clip_limit)
            # Trying to reduce the values above
clipping
            for _ in range(self.n_iter):
                clipped_hist =
pu.clip_histogram(hist, bins, self.clip_limit)
            # Calculating the CDF
            cdf = pu.calculate_cdf(hist, bins)
```

```
                # Changing the value of the image to the
result from the CDF for the given pixel
                padded_equalized_image[i][j] =
cdf[padded_image[i][j]]

        # Removing the padding from the image
        equalized_image =
padded_equalized_image[border:shape[0] - border,
border:shape[1] - border].astype(np.uint8)

        return equalized_image

    def clipped_histogram_equalization(self, region):
        '''Calculates the clipped histogram equalization for the
given region.

        Parameters:
            region: array-like.

        Returns a dictionary with the CDF for each pixel in
the region.
        '''

        # Building the histogram
        hist, bins = pu.histogram(region)
        n_bins = len(bins)

        # Removing values above clip_limit
        excess = 0
        for i in range(n_bins):
            if hist[i] > self.clip_limit:
                excess += hist[i] - self.clip_limit
                hist[i] = self.clip_limit

        ## Redistributing exceding values ##
        # Calculating the values to be put on all bins
```

```
        for_each_bin = excess // n_bins
        # Calculating the values left
        leftover = excess % n_bins

        hist += for_each_bin
        for i in range(leftover):
            hist[i] += 1

        # Calculating probability for each pixel
        pixel_probability = hist / hist.sum()
        # Calculating the CDF (Cumulative Distribution
Function)
        cdf = np.cumsum(pixel_probability)

        cdf_normalized = cdf * 255

        hist_eq = {}
        for i in range(len(cdf)):
            hist_eq[bins[i]] = int(cdf_normalized[i])

        return hist_eq

    def export_histogram(self, image, normalized, equalized):
        plt.xlabel("Pixel")
        plt.ylabel("Count")

        hist, bins = pu.histogram(image)
        plt.plot(bins, hist, label='Original Image')
        plt.legend()

        hist, bins = pu.histogram(normalized)
        plt.plot(bins, hist, label='Normalized Image')
        plt.legend()

        hist, bins = pu.histogram(equalized)
        plt.plot(bins, hist, label='CLAHE Result')
```

```
        plt.legend()
        plt.savefig(os.path.join(self.results_path,
"histograms.jpg"))

    def export_run_info(self, runtime):
        with open(os.path.join(self.results_path,
"runinfo.txt"), 'w+') as f:
            f.write(f"Runtime: {runtime:.2f}s\n")
            f.write(f"Window size: {self.window_size}\n")
            f.write(f"Clip limit: {self.clip_limit}\n")
```

runner.py:

```
import os
import imageio

from datetime import datetime

import src.arguments as ah

from src.algorithms.unsharping_mask import UM
from src.algorithms.clahe import CLAHE
from src.algorithms.hef import HEF

class AlgorithmRunner:
    def __init__(self):
        self.arg_handler    = ah.ArgumentHandler()
        self.algorithm              =
self.arg_handler.get_algorithm()
        self.image                  =
self.arg_handler.get_image()
        self.images_path    = self.arg_handler.get_path()
        output_path = self.arg_handler.get_output_path()
        if output_path:
```

```
            self.results_path = output_path
        else:
            self.results_path     = os.path.join("results",
str(datetime.now()))

        os.makedirs(self.results_path, exist_ok=True)

    def __del__(self):
        self.algorithm          = ''
        self.image                    = ''
        self.images_path    = ''
        self.results_path     = ''

    def run(self):
        '''Runs the algorithm in the images.'''

        if self.images_path:
            images = os.listdir(self.images_path)
            path = self.images_path
        else:
            # We put in a list to be able to utilize the for
loop
            images = [self.image]
            path = ""

        for image in images:
            split_image = image.split('/')
            if len(split_image) != 1:
                self.image = split_image[-1]
            else:
                self.image = image

            processed_image =
self.__run_algorithm(image, path)
            t = datetime.now()
            name = self.image.split(".")[0]
```

```
                filename =
f"{t.hour}_{t.minute}_{t.second}_{name}.jpg"
                imageio.imwrite(os.path.join(self.results_path,
filename), processed_image)

    def __run_algorithm(self, image, path):
        '''Runs the algorithm in the image.

        Parameters:
            image: image filename.
            path: image directory.

        Returns the processed image.
        '''

        img = os.path.join(path, image)
        alg = None

        # UM (Unsharping Mask)
        if self.algorithm == 'um':
            alg = UM(img)
        # CLAHE
        if self.algorithm == 'clahe':
            alg = CLAHE(img, self.results_path)
        # HEF
        if self.algorithm == 'hef':
            alg = HEF(img, self.results_path)

        try:
            image = alg.run()
        except Exception as e:
            print(e)
        else:
            return image
```

REFERENCES

1. The Exploring the effect of image enhancement techniques on COVID-19 detection using chest X-ray images

T Rahman, A Khandakar, Y Qiblawey, A Tahir… - Computers in biology …, 2021

2. Application of deep learning techniques for detection of COVID-19 cases using chest X-ray images: A comprehensive study

SR Nayak, DR Nayak, U Sinha, V Arora… - … Signal Processing and …, 2021

3. Fuzzy system based medical image processing for brain disease prediction

M Hu, Y Zhong, S Xie, H Lv, Z Lv - Frontiers in Neuroscience, 2021

4. Directional mutation and crossover boosted ant colony optimization with application to COVID-19 X-ray image segmentation

A Qi, D Zhao, F Yu, AA Heidari, Z Wu, Z Cai… - Computers in biology …, 2022

5. The UK Biobank imaging enhancement of 100,000 participants: rationale, data collection, management and future directions

TJ Littlejohns, J Holliday, LM Gibson, S Garratt… - Nature …, 2020

6. Deep learning system for COVID-19 diagnosis aid using X-ray pulmonary images

J Civit-Masot, F Luna-Perejón, M Domínguez Morales… - Applied Sciences, 2020

7. Hybrid deep learning for detecting lung diseases from X-ray images

S Bharati, P Podder, MRH Mondal - Informatics in Medicine Unlocked, 2020

8. Automated methods for detection and classification pneumonia based on x-ray images using deep learning

K El Asnaoui, Y Chawki, A Idri - Artificial intelligence and blockchain for …, 2021

9. A novel image processing approach to enhancement and compression of X-ray images

Y Pourasad, F Cavallaro - … Journal of Environmental Research and Public …, 2021

10. From Pixels to Diagnoses: Deep Learning's Impact on Medical Image Processing-A Survey

MM Mijwil, AH Al-Mistarehi, M Abotaleb… - Wasit Journal of …, 2023

11. Deep learning and medical image processing for coronavirus (COVID-19) pandemic: A survey

S Bhattacharya, PKR Maddikunta, QV Pham… - Sustainable cities and …, 2021

12. A review of airport dual energy X-ray baggage inspection techniques: Image enhancement and noise reduction

SU Khan, IU Khan, I Ullah, N Saif… - Journal of X-ray Science …, 2020

13. Artificial intelligence techniques for cancer detection in medical image processing: A review

C Kaur, U Garg - Materials Today: Proceedings, 2023X-ray Image Enhancement Framework Based on Improved Local Adaptive Contrast Field for Complex Workpieces ….. Y Li, M Han, K Li, Y Han, P Chen - IEEE Transactions on …, 2024

14. Enhancement of X-ray images using various Image Processing Approaches

S Naidu, A Quadros, A Natekar… - 2021