Department of Computer Science and Engineering (Data Science)
Image Processing and Computer Vision I (DJ19DSL603)

Lab 5: Image Enhancement in Spatial Domain using Neighbourhood Processing Techniques

NAME : Aaditya Malani **SAP ID : 60009220192** **BATCH:D1-1**

Aim: To perform image enhancement in spatial domain using neighbourhood processing techniques: Basic High Pass and High Boost filtering

Theory:

1. **Basic High Pass:**

   The principal objective of high pass (sharpening) filter is to highlight fine detail in an image or to enhance detail that has been blurred, either in error or as a natural effect of a particular method of image acquisition. Uses of image sharpening vary and include applications ranging from electronic printing and medical imaging to industrial inspection and autonomous target detection in military systems.

   The shape of the impulse response needed to have a high pass (sharpening) spatial filter indicates that the filter should have positive coefficients in the outer periphery. For a 3 x 3 mask, choosing a positive value in the centre location with negative coefficients in the rest of the mask meets this condition. Thus when the mask is over an area of constant or slowly varying gray level, the output of the mask is zero or very small. This result is consistent with what is expected from the corresponding frequency domain filter.

−1	−1	−1
−1	8	−1
−1	−1	−1

fig 1. A high pass filter mask

2. **High Boost Filter:**

   The goal of high boost filtering is to enhance the high frequency information without completely eliminating the background of the image.

   We know that:

   (High-pass filtered image)=(Original image)-(Low-pass filtered image)

   We define:

(High boost filtered image) = A × (Original image) - (Low-pass filtered image)
(High boost) = (A−1) × (Original) + (Original) - (Low-pass)
(High boost) = (A−1) × (Original) + (High-pass)

Note:

i. when A>1, part of the original is added back to the high-pass filtered version of the image in order to partly restore the lowfrequency components that would have been eliminated with standardhigh-pass filtering.
ii. Typical values for A are values slightly higher than 1, as for example 1.15, 1.2, etc.

The resulting image looks similar to the original image with some edge enhancement.

The spatial mask that implements the high boost filtering algorithm is shown below.

$$\frac{1}{9} \times \begin{array}{|c|c|c|}\hline -1 & -1 & -1 \\ \hline -1 & w = 9A-1 & -1 \\ \hline -1 & -1 & -1 \\ \hline \end{array}$$

fig 2. High boost filter mask

The resulting image depends on the choice of A.

$A = 1.15$

$A = 1.2$

Lab Assignments to complete in this session

Shri Vile Parle Kelavani Mandal's
DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING
(Autonomous College Affiliated to the University of Mumbai)
NAAC Accredited with "A" Grade (CGPA : 3.18)

Department of Computer Science and Engineering (Data Science)
Image Processing and Computer Vision I (DJ19DSL603)

Lab 5: Image Enhancement in Spatial Domain using Neighbourhood Processing Techniques

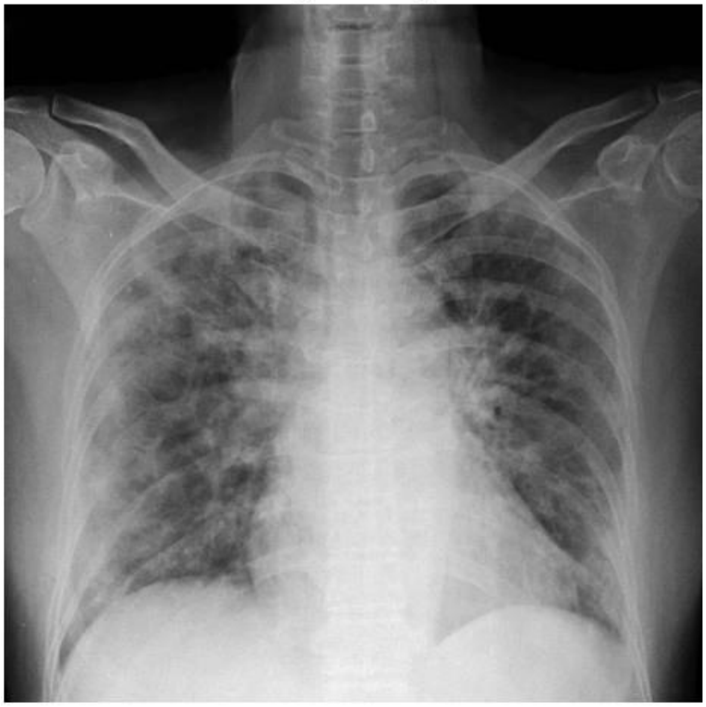
Problem Statement: Develop a Python program utilizing the OpenCV library to enhace the images in spatial domain using neighbourhood processing with sharpening operators (High pass filtering and High boost filtering). The program should address the following tasks:

1. Read any low contrast image from COVID 19 Image Dataset.
   Dataset Link: **Covid-19 Image Dataset**
2. Display the before & after image(s) used in every task below.
3. Apply basic high pass filter and compare the before and after result.
4. Apply basic high boost filter and compare the before and after result.

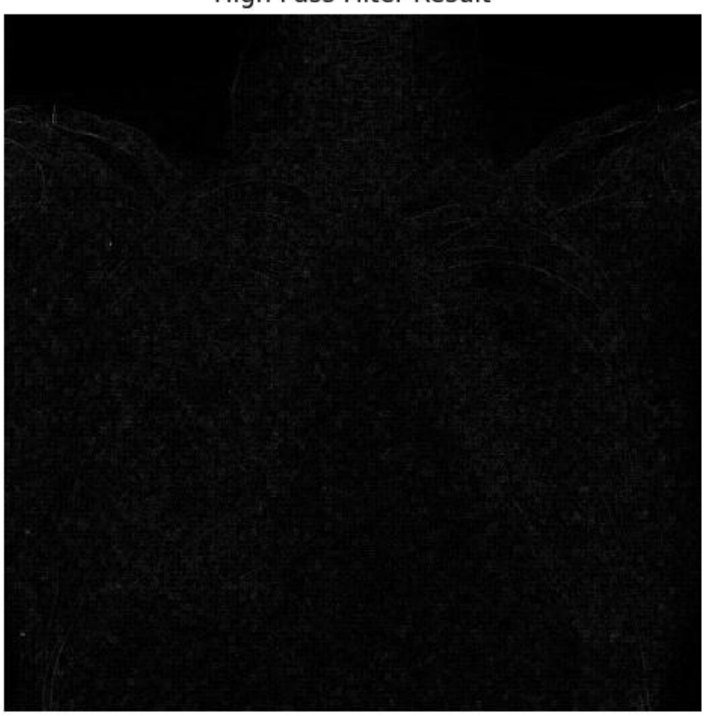
The solution to the operations performed must be produced by scratch coding without the use of built in OpenCV methods.

```python
import cv2
import numpy as np
import matplotlib.pyplot as plt


def display_image(title, img):
    plt.figure(figsize=(6, 6))
    plt.imshow(img, cmap='gray')
    plt.title(title)
    plt.axis('off')
    plt.show()

def high_pass_filter(image):
    kernel = np.array([[-1, -1, -1],
                       [-1,  8, -1],
                       [-1, -1, -1]])
    high_pass = cv2.filter2D(image, -1, kernel)
    return high_pass

def high_boost_filter(image, A):
    low_pass_kernel = np.ones((3, 3), np.float32) / 9.0
    low_pass = cv2.filter2D(image, -1, low_pass_kernel)
    high_pass = image - low_pass
    high_boost = A * image - low_pass
    return high_boost

def apply_high_boost_for_multiple_A(image, A_values):
    for A in A_values:
        high_boost = high_boost_filter(image, A)
        display_image(f'High Boost Filter (A = {A})', high_boost)

image_path    =    '/content/x-ray-image-2b_full.jpg'
image = cv2.imread(image_path, cv2.IMREAD_GRAYSCALE)


display_image('Original Image', image)


high_pass = high_pass_filter(image)
display_image('High Pass Filter Result', high_pass)


A_values = [1.1, 1.2, 1.5, 2.0]
apply_high_boost_for_multiple_A(image, A_values)
```

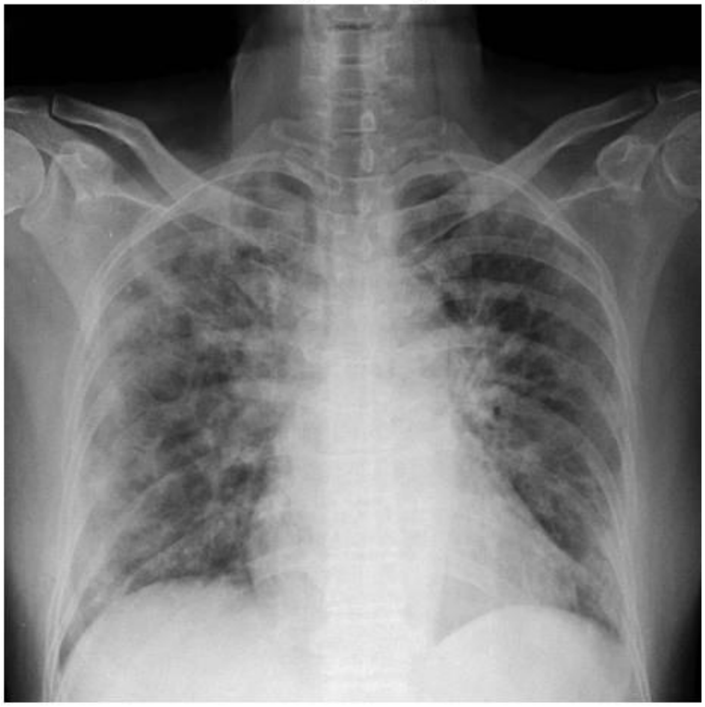
Original Image



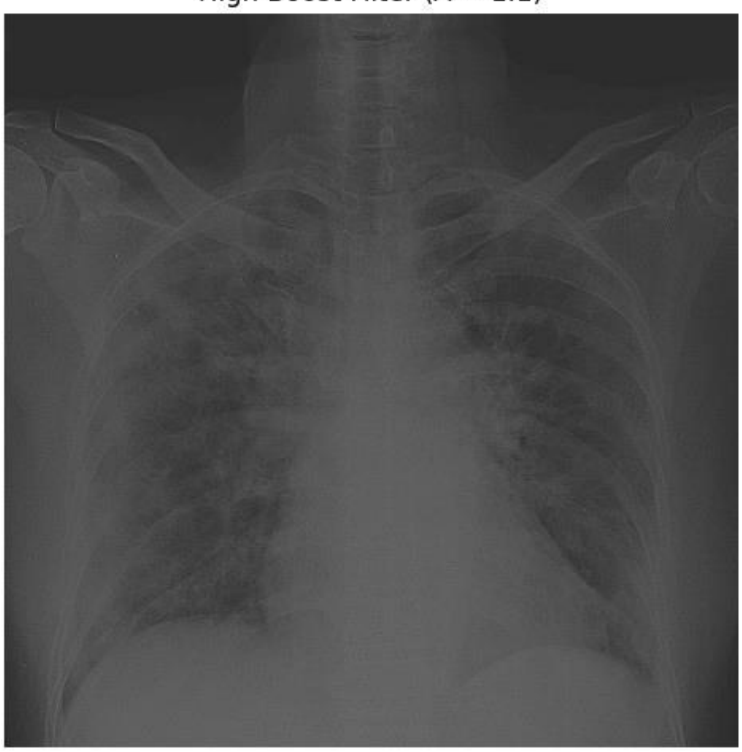
High Pass Filter Result



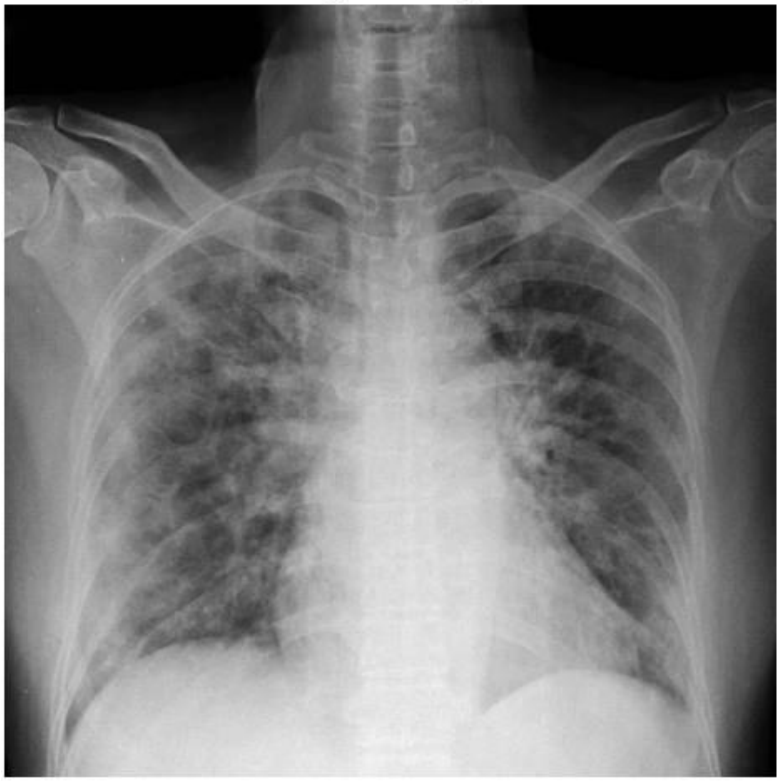
Original Image



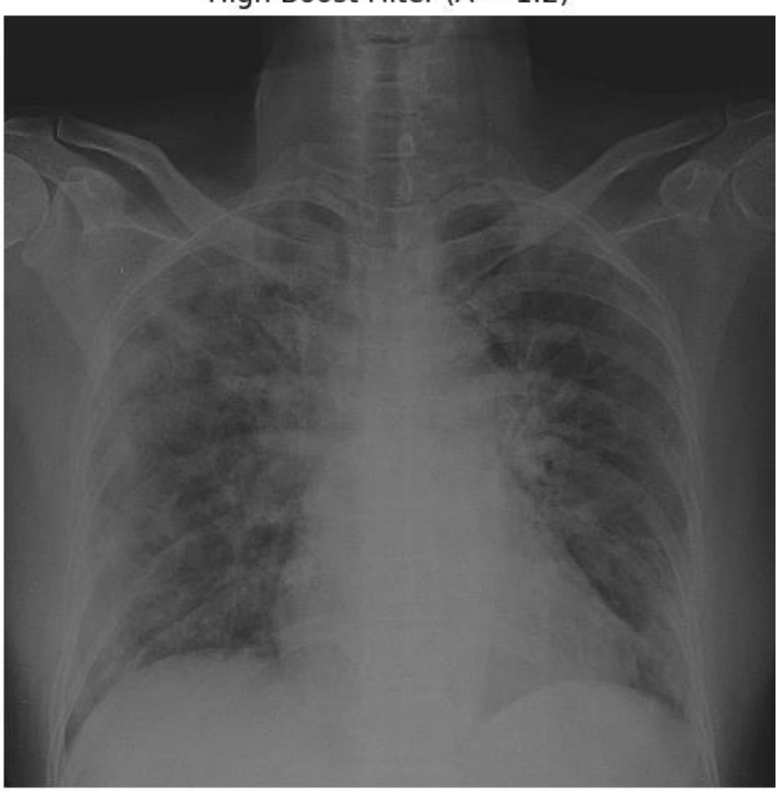
High Boost Filter (A = 1.1)



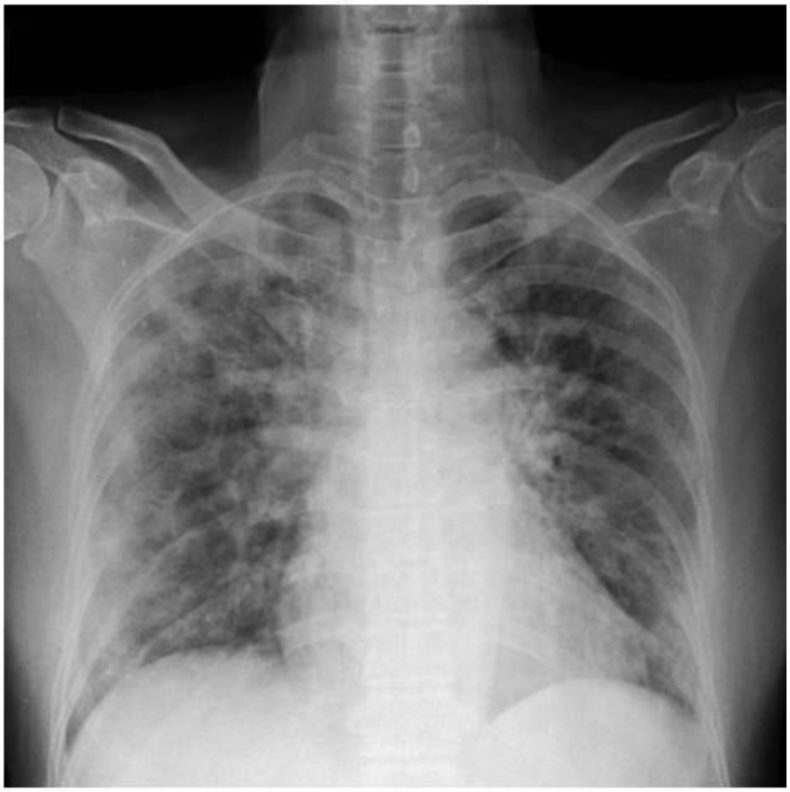
Original Image



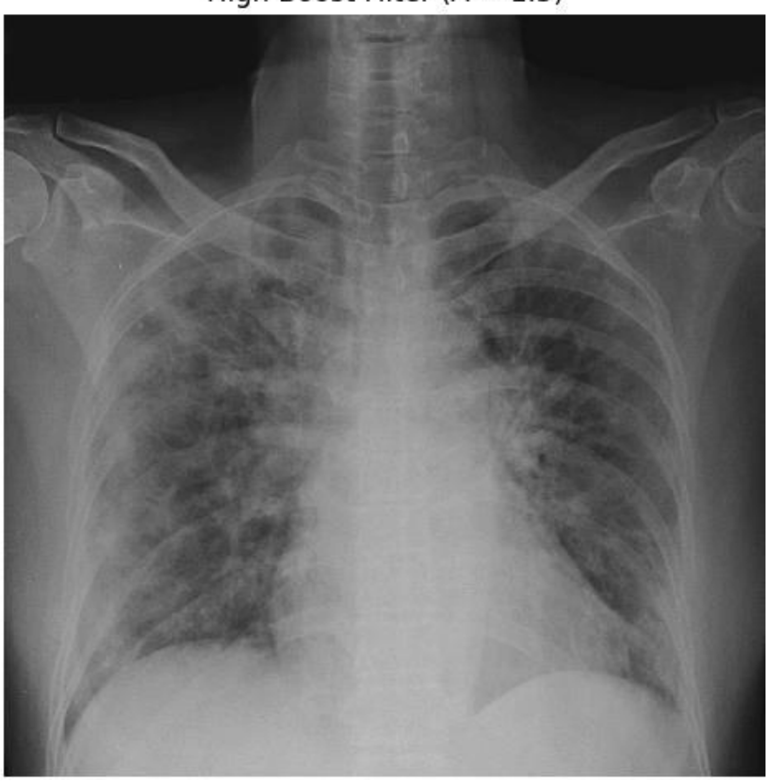
High Boost Filter (A = 1.2)



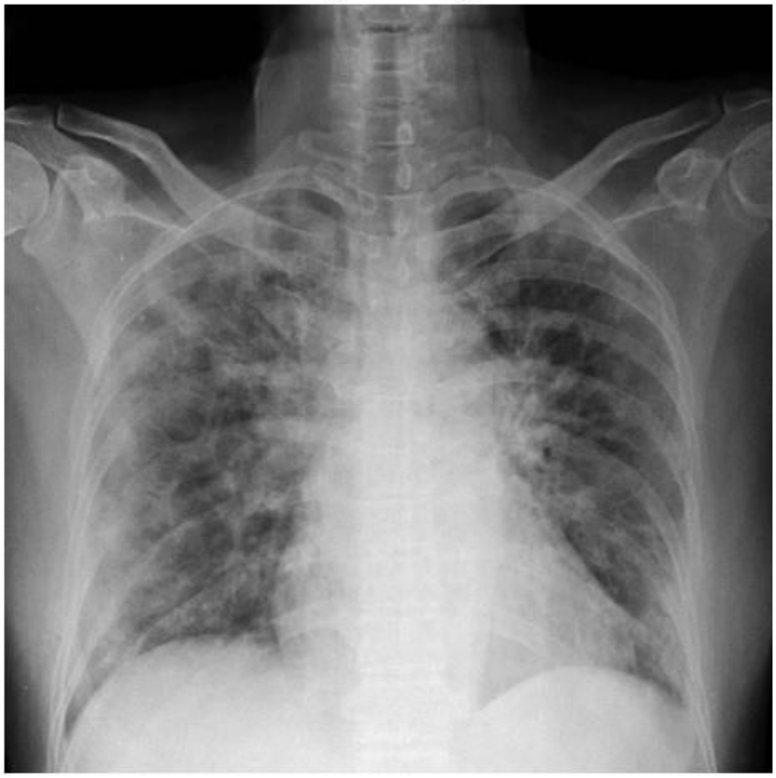
Original Image



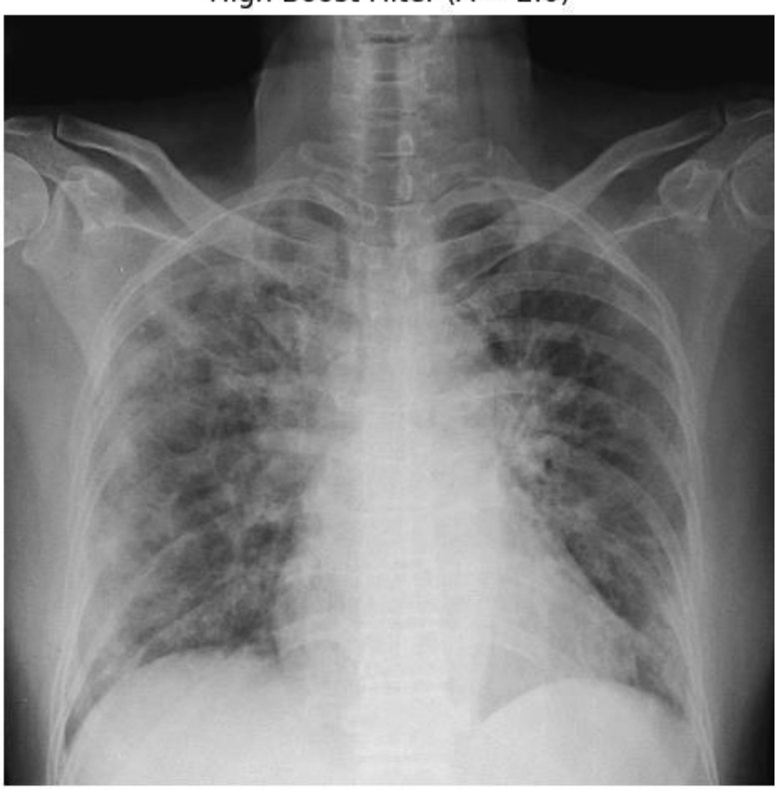
High Boost Filter (A = 1.5)



Original Image



High Boost Filter (A = 2.0)



Conclusion:
In the High Pass and High Boost filtering experiments, both methods enhance specific frequency components of an image, but their purposes differ.
–High Pass Filtering is used to emphasize the high-frequency details (such as edges, textures, and fine details) while suppressing low-frequency components (like smooth areas and gradients). This technique helps in edge detection and image sharpening.
- High Boost Filtering takes High Pass filtering a step further by not only enhancing the high-frequency components but also retaining a portion of the original image. This produces a sharper result while preserving some of the low-frequency components, thus combining both enhancement and preservation of details. High Boost filtering is useful when subtle image details need to be highlighted without losing the overall image content.

Application:
1. Image Sharpening: High pass and high boost filters are commonly used to sharpen images by highlighting the edges and fine details. These techniques are vital in improving the visual quality of images in fields like photography, medical imaging, and remote sensing.
2. Edge Detection: High pass filtering helps in detecting the boundaries between objects, which is crucial in computer vision and pattern recognition systems such as object detection or facial recognition. 3. Enhancing Fine Details in Medical Imaging: In applications like MRI or X-ray imaging, High Boost filtering is used to highlight fine structures and details that might otherwise be difficult to detect.
4. Satellite Image Processing: Both techniques are used in satellite imaging to sharpen terrain details and enhance structures such as roads, buildings, and other features of interest.
5. Video Enhancement: In video processing, these filters can be applied to enhance edges and fine details to improve the overall visual experience, especially in HD or 4K video formats.