LUNG DISEASE DETECTION

(1)

Problem Statement:

Design and implement a Lung Disease Detection model using image processing techniques to assist medical professionals in the accurate and early diagnosis of lung diseases, including Bacterial pneumonia, Viral pneumonia and Covid_19,based on medical imaging data such as X-ray or CT scans.

Motivation:

The early diagnosis and treatment of lung diseases through innovative image processing technologies hold the promise of a brighter future in healthcare. By detecting lung diseases at its earliest, most treatable stages, we aim to significantly improve survival rates and reduce the heavy toll this disease takes on individuals and their families. Simultaneously, early detection can help cut down healthcare costs associated with late-stage treatments and invasive procedures, making quality care more accessible. Moreover, this approach doesn't just extend lives; it enhances the quality of life for patients by minimizing the burdens of aggressive treatments. As we strive for progress, the public health impact cannot be understated, as it brings us closer to reducing the overall burden of lung diseases worldwide. Lastly, it fosters awareness and education, empowering individuals to take proactive steps towards early screening and healthier living, creating a comprehensive approach to tackling this pervasive health challenge.

Application:

The applications of early lung diseases diagnosis and treatment, driven by innovative image processing techniques, are wide-ranging and impactful. These advancements can revolutionize the field of healthcare by enabling timely and accurate identification of lung diseases, which is crucial for improving patient outcomes. Medical professionals can integrate these technologies into routine screenings and diagnostic processes, leading to earlier interventions and potentially life-saving treatments. Additionally, these applications have the potential to reduce the economic burden on healthcare systems by curbing the costs associated with late-stage diseases treatments and hospitalizations. Beyond the clinical realm, this technology promotes public health by addressing a leading cause of diseases-related mortality and fostering awareness and education about lung diseases risk factors and early detection methods. Ultimately, the applications of early lung diseases diagnosis through image processing offer a holistic approach to improving healthcare, from individual patient care to broader societal health outcomes.

(2) Dataset Details:

URL = https://www.kaggle.com/datasets/darshan1504/covid19-detection-xray-dataset

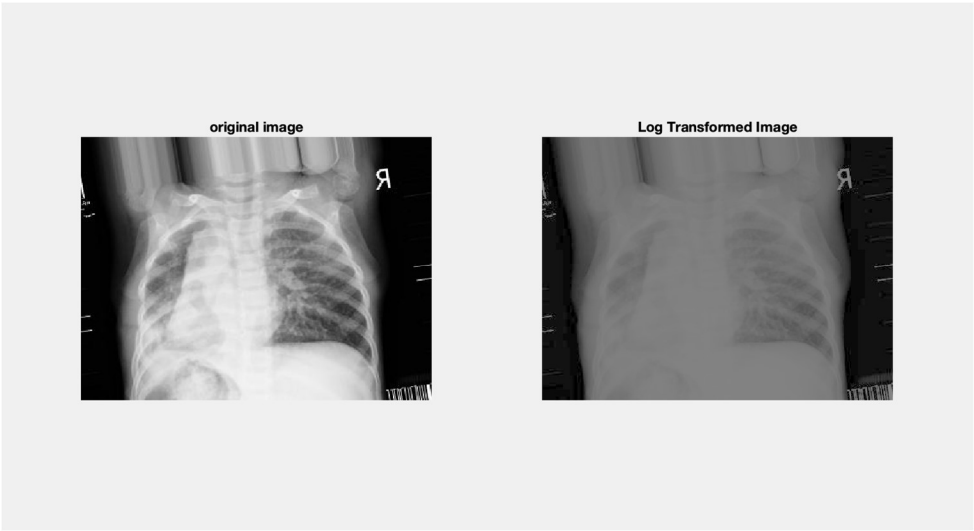
	BacterialPneumonia	ViralPneumonia	COVID_19	Normal
Training Data Size	650	412	60	901
Test Data Size	324	205	9	450

Image Size = 300x400x1

(3) Intensity Transformation:

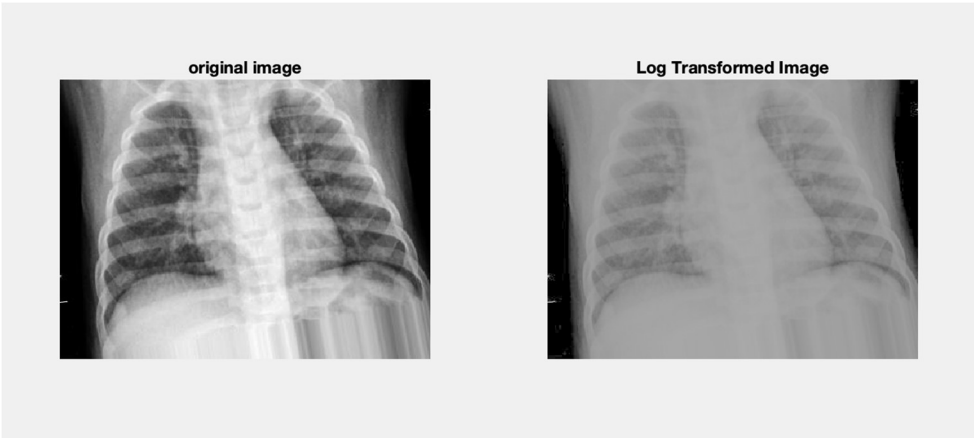
(1) Log Transformation

Bacterial Pneumonia



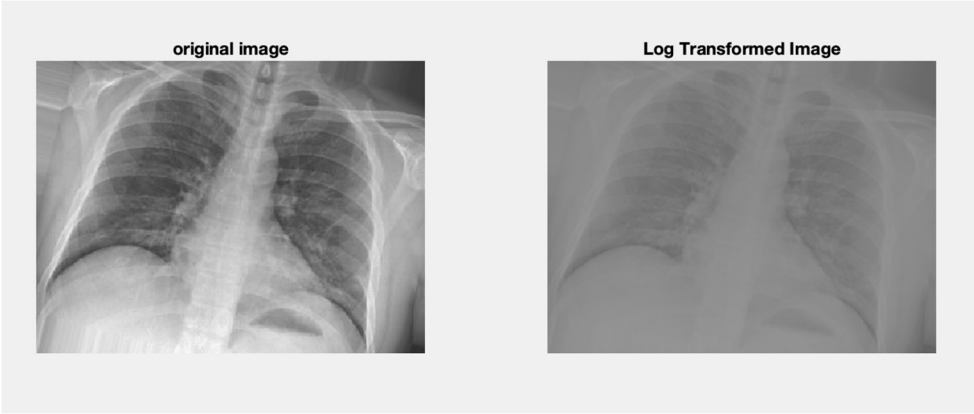
```
>> a = imread("/Users/nithylesh/Downloads/imgs/BacterialPneumonia.jpeg")
a = imread("/Users/nithylesh/Downloads/imgs/BacterialPneumonia.jpeg")
logimg = (uint8(35 * (0.8*log(1+double(a)/log(10)))))
subplot(1,2,1)
imshow(a)
title('original image')
subplot(1,2,2)
imshow(logimg)
title('Log Transformed Image')
```

Viral Pneumonia



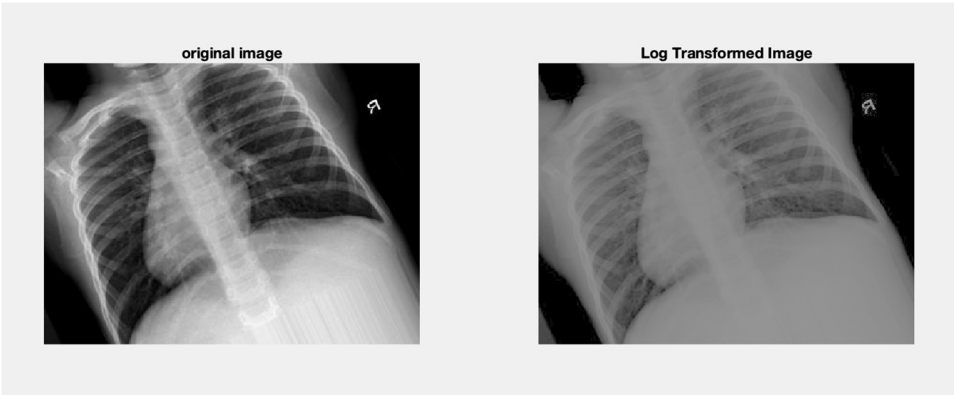

```
>> a = imread("/Users/nithylesh/Downloads/imgs/ViralPneumonia.jpeg")
logimg = (uint8(45 * (0.8*log(1+double(a)/log(10)))))
subplot(1,2,1)
imshow(a)
title('original image')
subplot(1,2,2)
imshow(logimg)
title('Log Transformed Image')
```

COVID_19



```
>> a = imread("/Users/nithylesh/Downloads/imgs/COVID_19.jpeg")
logimg = (uint8(40 * (0.8*log(1+double(a)/log(10)))))
subplot(1,2,1)
imshow(a)
title('original image')
subplot(1,2,2)
imshow(logimg)
title('Log Transformed Image')
```

Normal



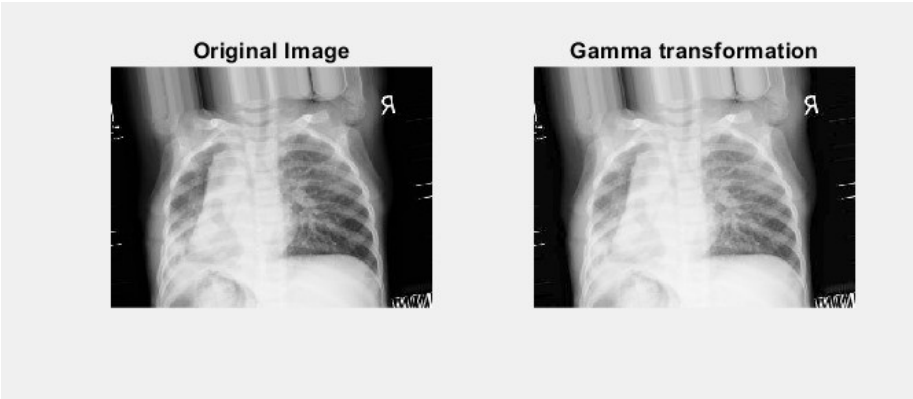

```
>> a = imread("/Users/nithylesh/Downloads/imgs/Normal.jpeg")
logimg = (uint8(40 * (0.8*log(1+double(a)/log(10)))))
subplot(1,2,1)
imshow(a)
title('original image')
subplot(1,2,2)
imshow(logimg)
title('Log Transformed Image')
```

Inference:

log transformation is a valuable tool in enhancing X-ray images, particularly for improving the visibility of subtle details and low-contrast regions.But in our case it has reduced the visibility of disease affected regions in X Ray of lungs.

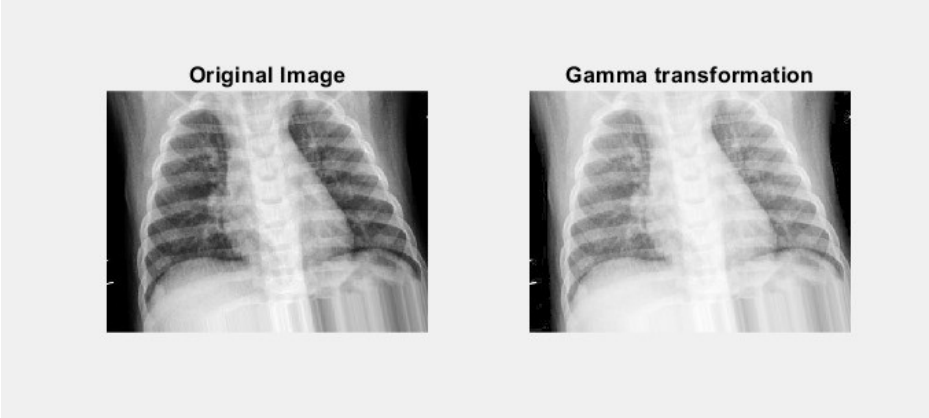
(2) Gamma Transformation

Bacterial Pneumonia



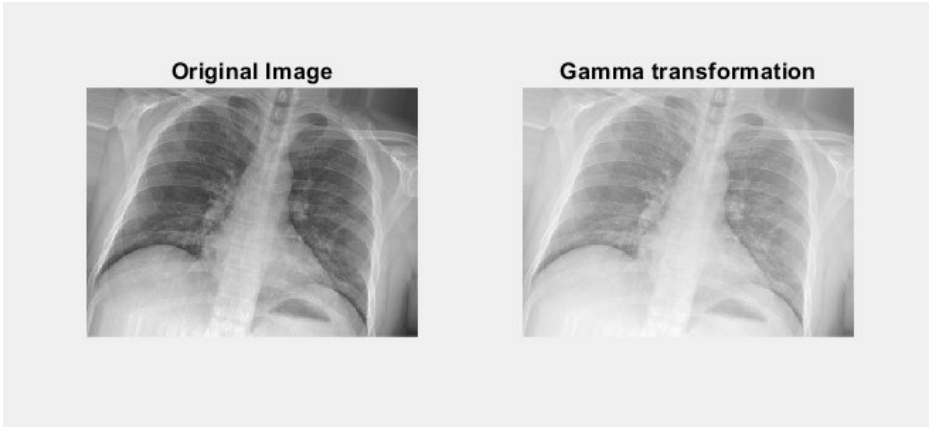

```
%Gamma Transformation
gamma = uint8(255 * (double(Bacteria) / 255).^0.7);
figure;
subplot(1, 2, 1);
imshow(Bacteria);
title('Original Image');
subplot(1, 2, 2);
imshow(gamma);
title("Gamma transformation");
```

Viral Pneumonia



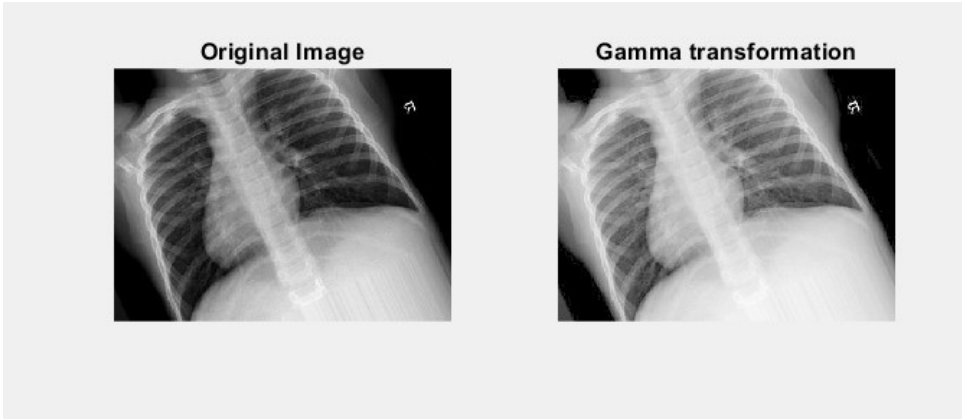

```
%Gamma Transformation
gamma = uint8(255 * (double(viral) / 255).^0.5);
figure;
subplot(1, 2, 1);
imshow(viral);
title('Original Image');
subplot(1, 2, 2);
imshow(gamma);
title("Gamma transformation");
```

COVID_19




```
%Gamma Transformation
gamma = uint8(255 * (double(covid) / 255).^0.5);
figure;
subplot(1, 2, 1);
imshow(covid);
title('Original Image');
subplot(1, 2, 2);
imshow(gamma);
title("Gamma transformation");
```

Normal



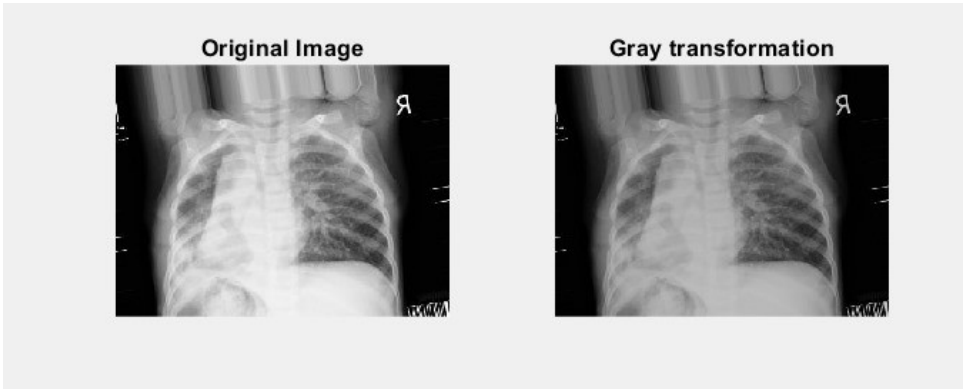

```
%Gamma Transformation
gamma = uint8(255 * (double(Normal) / 255).^0.5);
figure;
subplot(1, 2, 1);
imshow(Normal);
title('Original Image');
subplot(1, 2, 2);
imshow(gamma);
title("Gamma transformation");
```

Inference:

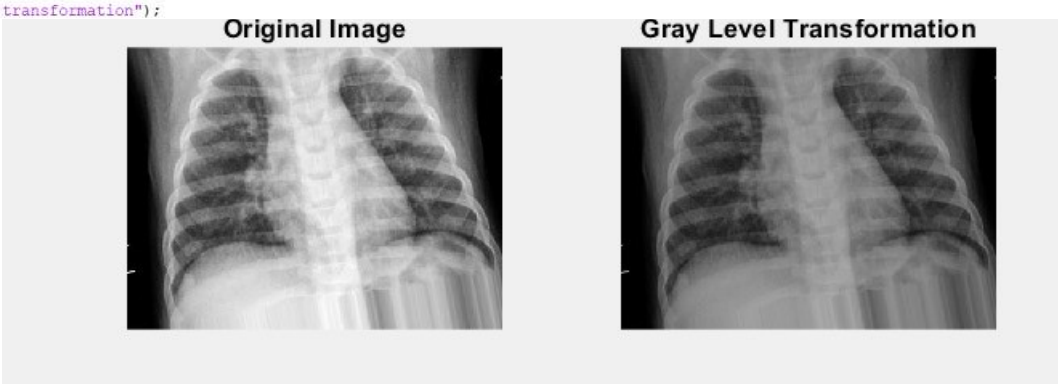
Gamma transformation is a versatile tool for enhancing X-ray images by adjusting their contrast and brightness. We have used Gamma Transformation to enhance the X Rays which improves the visibility of disease affected regions.

(3) Grey Transformation

Bacterial Pneumonia

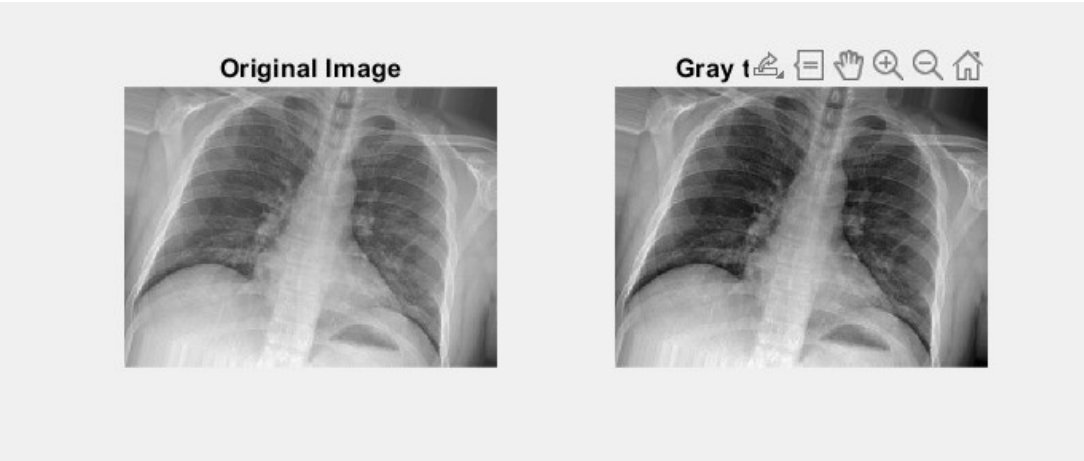



```
%Gray level Transformation
min_intensity = 0;
max_intensity = 200;
Bacteria1 = double(Bacteria);
transformed_gray = uint8((double(Bacteria1) - min(Bacteria1(:))) / (max(Bacteria1(:)) - min(Bacteria1(:))) * (max_intensity - min_intensity) + min_intensity);
figure;
subplot(1, 2, 1);
imshow(Bacteria);
title('Original Image');
subplot(1, 2, 2);
imshow(transformed_gray);
title("Gray transformation");
```



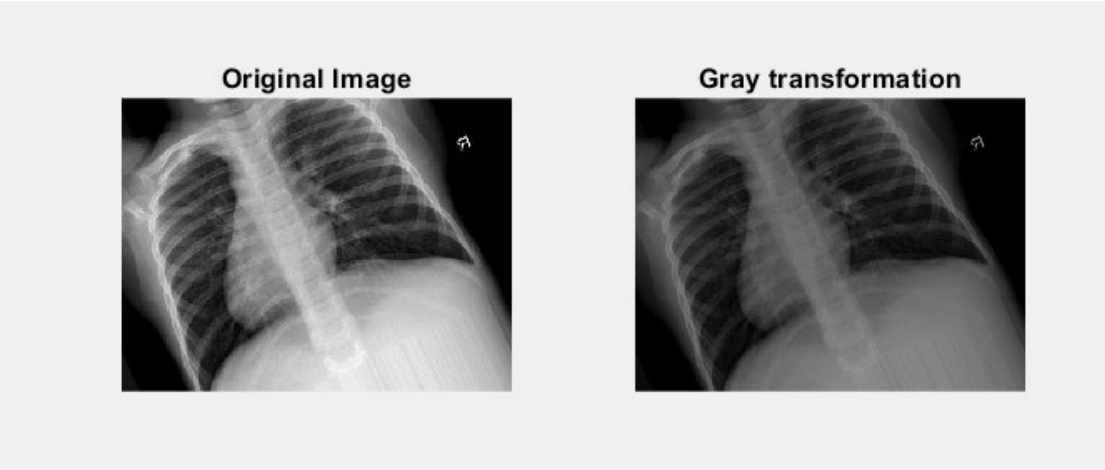

```
%Gray level Transformation
min_intensity = 0;
max_intensity = 170;
viral1 = double(viral);
transformed_gray = uint8((double(viral1) - min(viral1(:))) / (max(viral1(:)) - min(viral1(:))) * (max_intensity - min_intensity) + min_intensity);
figure;
subplot(1, 2, 1);
imshow(viral);
title('Original Image');
subplot(1, 2, 2);
imshow(transformed_gray);
title('Gray Level Transformation');
```

COVID_19




```
%Gray level Transformation
min_intensity = 0;
max_intensity = 255;
covid1 = double(covid);
transformed_gray = uint8((double(covid1) - min(covid1(:))) / (max(covid1(:)) - min(covid1(:))) * (max_intensity - min_intensity) + min_intensity);
figure;
subplot(1, 2, 1);
imshow(covid);
title('Original Image');
subplot(1, 2, 2);
imshow(transformed_gray);
title("Gray transformation");
```

Normal



```
%Gray level Transformation
min_intensity = 0;
max_intensity = 150;
Normal1 = double(Normal);
transformed_gray = uint8((double(Normal1) - min(Normal1(:))) / (max(Normal1(:)) - min(Normal1(:))) * (max_intensity - min_intensity) + min_intensity);
figure;
subplot(1, 2, 1);
imshow(Normal);
title('Original Image');
subplot(1, 2, 2);
imshow(transformed_gray);
title("Gray transformation");
```

Inference:

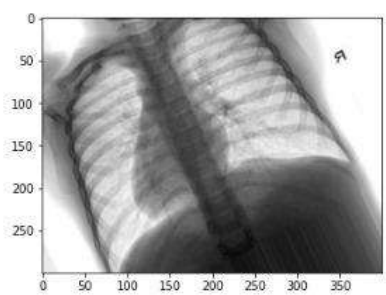
The inference for grayscale transformation of X-ray images is that it simplifies the image representation, focuses on intensity information, and enhances visualization and interpretability. However in our case this has decreased the visibility of diseased regions in lungs.

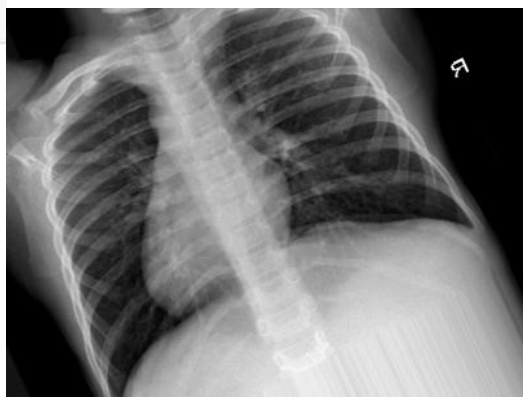
Negative Transformation **Original Image**

For normal

```
complement_normal_img = 255 - normal_img
plt.imshow(complement_normal_img)
```

```
<matplotlib.image.AxesImage at 0x21efee38700>
```

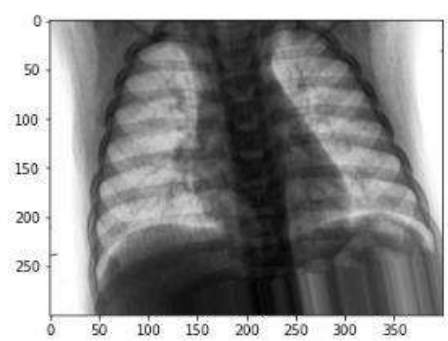




For Viral

```
complement_vpneumonia_img = 255 - vpneumonia_img
plt.imshow(complement_vpneumonia_img)
```

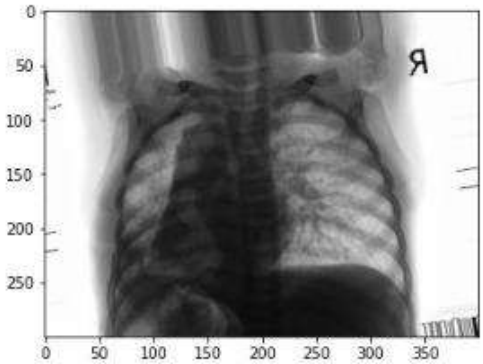
```
<matplotlib.image.AxesImage at 0x21e810242e0>
```

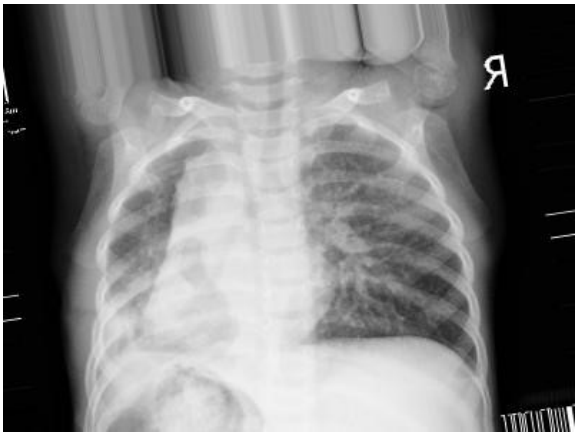


For Bacterial Pneumonia

```
complement_bpneumonia_img = 255 - bpneumonia_img
plt.imshow(complement_bpneumonia_img)
```

```
<matplotlib.image.AxesImage at 0x21efed28fd0>
```

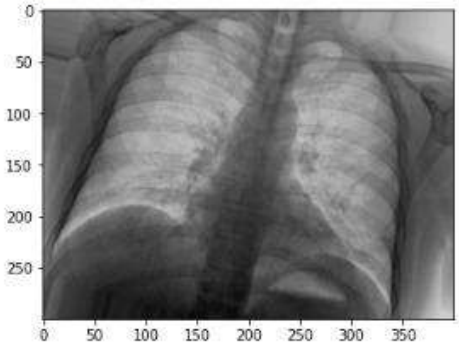




For Covid

```
complement_covid_img = 255 - covid_img
plt.imshow(complement_covid_img)
```

```
<matplotlib.image.AxesImage at 0x21efec55450>
```



Inference:

Negative transformation can be used to enhance certain features and structures in X-ray images, particularly lungs. However in our case it reduces the visibility of diseased regions.

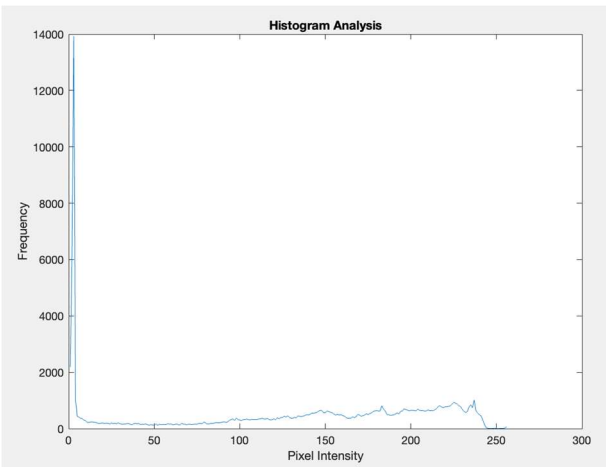
Final Inference:

We have considered Gamma transformation as the final technique to enhance our image because it prioritizes the diseased regions in lungs whereas if you consider log transformation it prioritizes on enhancing subtle regions and low contrast regions, gray transformation prioritizes on intensity transformation but it reduces the disease visibility, negative transformation prioritizes in enhancing lung region but it reduces the visibility of diseased regions.

Histogram Transformation:

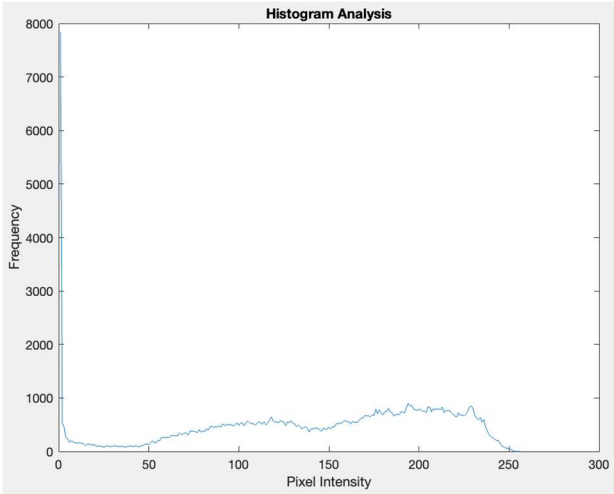
(4) Histogram

Bacterial Pneumonia



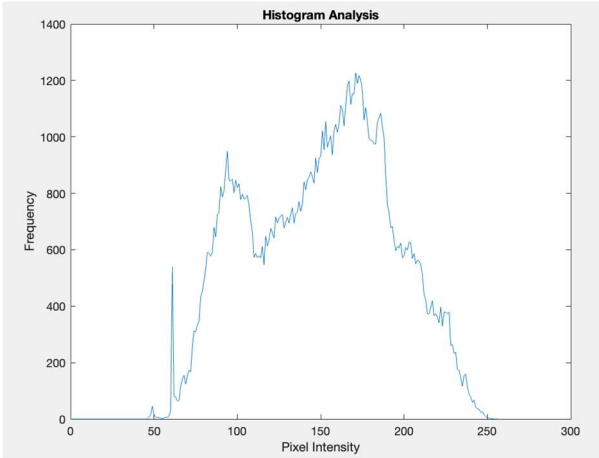

```
inputImage = imread('/Users/nithylesh/Downloads/imgs/BacterialPneumonia.jpeg');
histogramValues = imhist(inputImage);
plot(histogramValues);
title('Histogram Analysis');
xlabel('Pixel Intensity');
ylabel('Frequency');
```

Viral Pneumonia



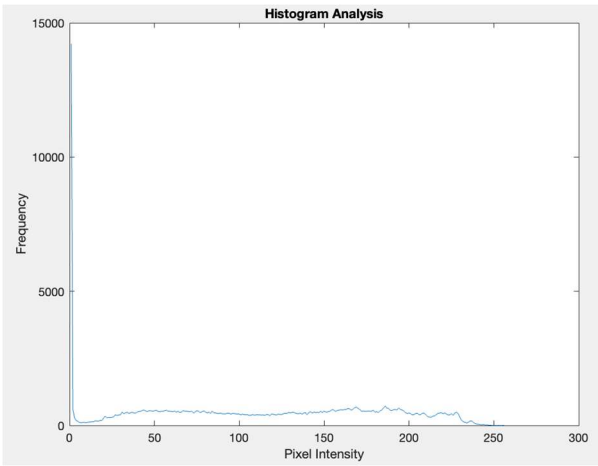

```
>> inputImage = imread('/Users/nithylesh/Downloads/imgs/ViralPneumonia.jpeg');
histogramValues = imhist(inputImage);
plot(histogramValues);
title('Histogram Analysis');
xlabel('Pixel Intensity');
ylabel('Frequency');
```

COVID 19




```
>> inputImage = imread('/Users/nithylesh/Downloads/imgs/COVID_19.jpeg');
histogramValues = imhist(inputImage);
plot(histogramValues);
title('Histogram Analysis');
xlabel('Pixel Intensity');
ylabel('Frequency');
```

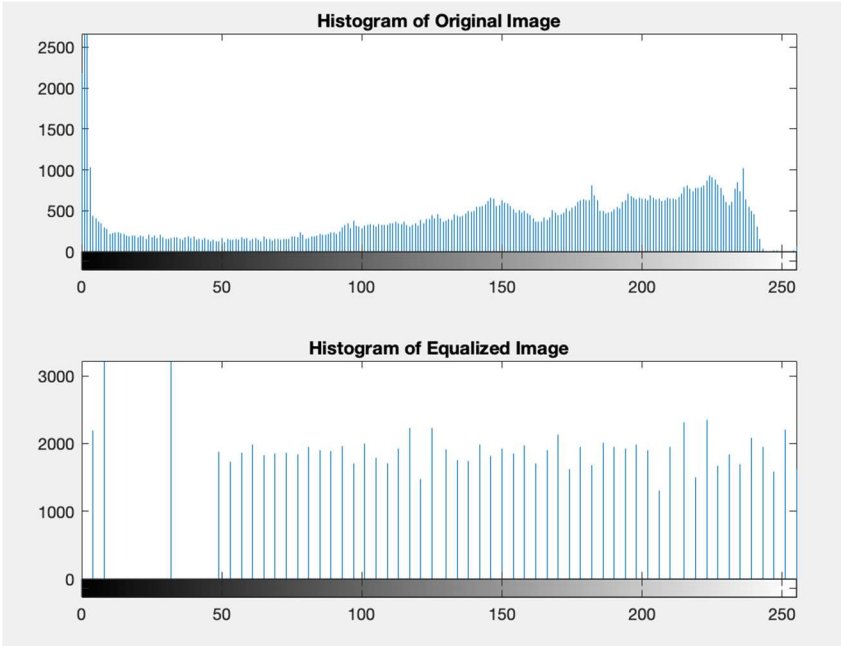
Normal

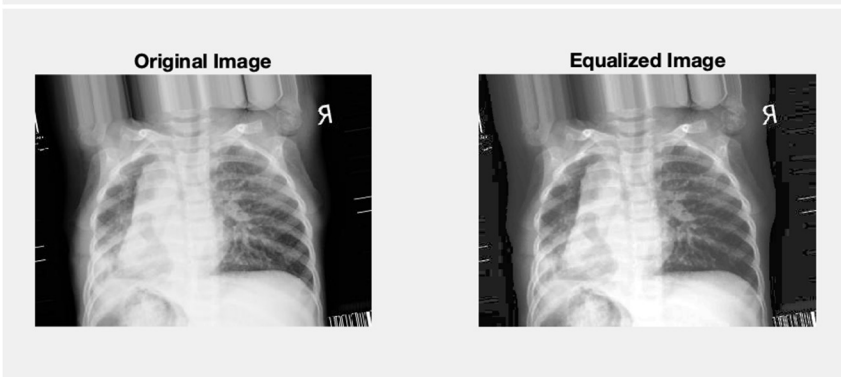



```
>> inputImage = imread('/Users/nithylesh/Downloads/imgs/Normal.jpeg');
histogramValues = imhist(inputImage);
plot(histogramValues);
title('Histogram Analysis');
xlabel('Pixel Intensity');
ylabel('Frequency');
```

Histogram Equalization:

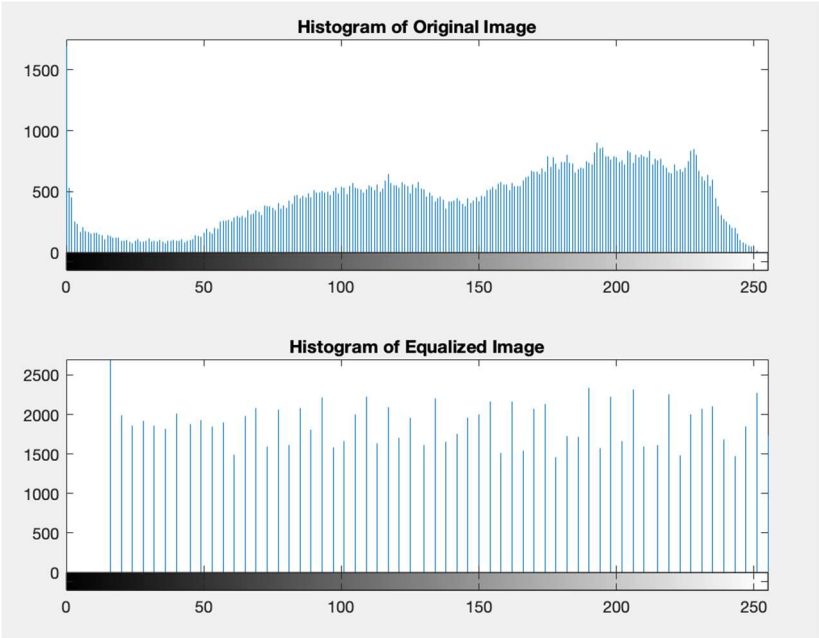
Bacterial Pneumonia

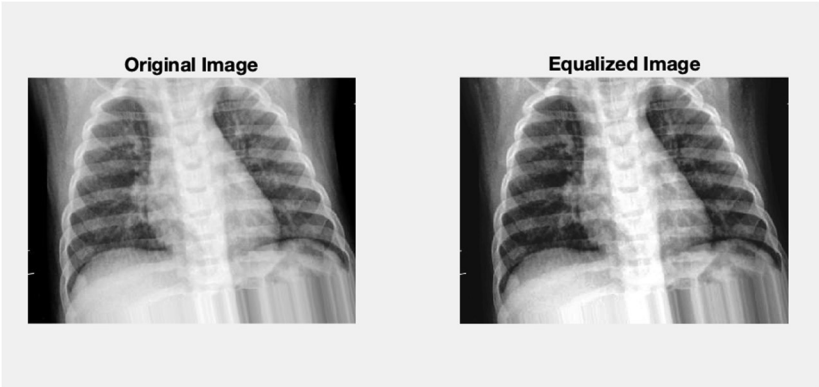




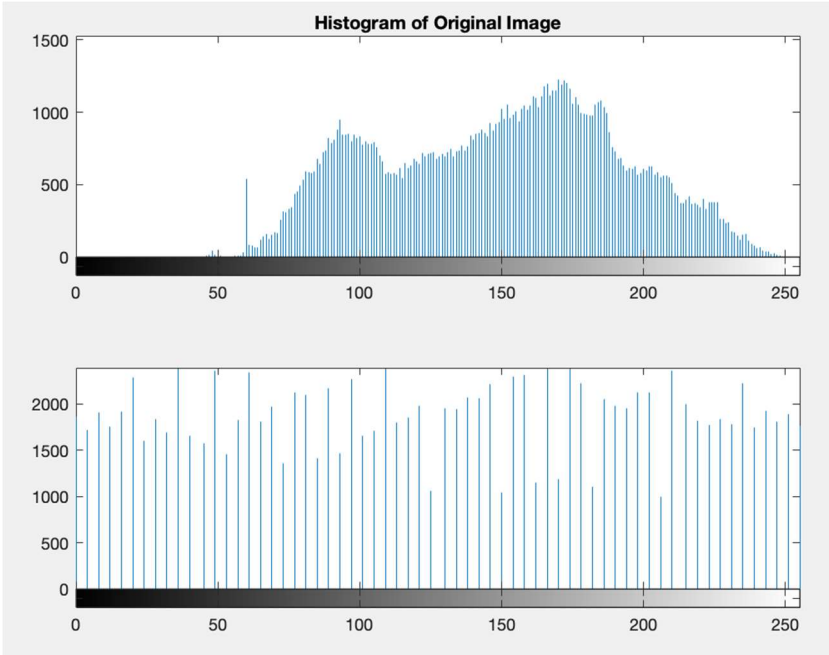

```
inputImage = imread('/Users/nithylesh/Downloads/imgs/BacterialPneumonia.jpeg');
outputImage = histeq(inputImage);
subplot(1, 2, 1);
imshow(inputImage);
title('Original Image');
subplot(1, 2, 2);
imshow(outputImage);
title('Equalized Image');
figure;
subplot(2, 1, 1);
imhist(inputImage);
title('Histogram of Original Image');
subplot(2, 1, 2);
```

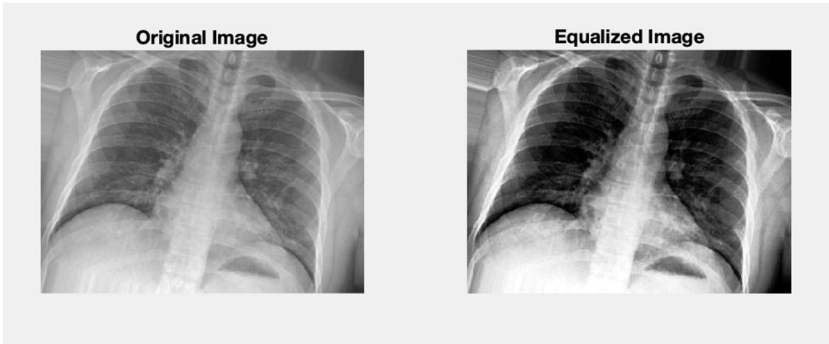
Viral Pneumonia


Histogram of Original Image
1500
1000
500
0
0
50
100
150
200
250
Histogram of Equalized Image
2500
2000
1500
1000
500
0
0
50
100
150
200
250


Original Image
Equalized Image

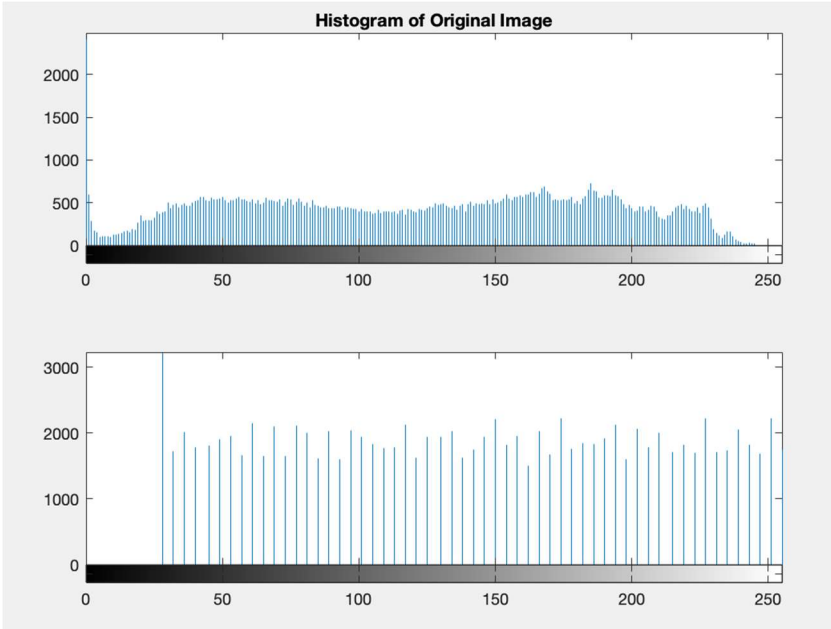
COVID 19

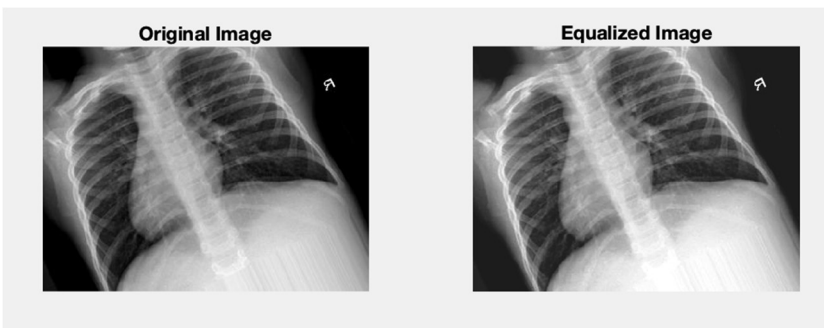





```
>> inputImage = imread('/Users/nithylesh/Downloads/imgs/COVID_19.jpeg');
outputImage = histeq(inputImage);
subplot(1, 2, 1);
imshow(inputImage);
title('Original Image');
subplot(1, 2, 2);
imshow(outputImage);
title('Equalized Image');
figure;
subplot(2, 1, 1);
imhist(inputImage);
title('Histogram of Original Image');
subplot(2, 1, 2);
imhist(outputImage);
```

Normal

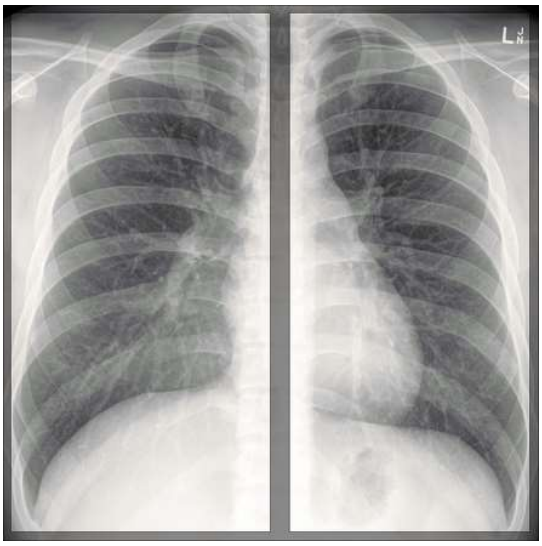




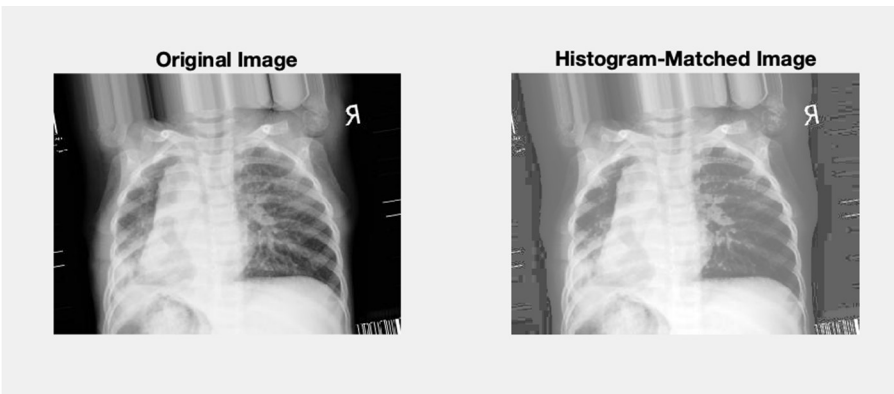

```
>> inputImage = imread('/Users/nithylesh/Downloads/imgs/Normal.jpeg');
outputImage = histeq(inputImage);
subplot(1, 2, 1);
imshow(inputImage);
title('Original Image');
subplot(1, 2, 2);
imshow(outputImage);
title('Equalized Image');
figure;
subplot(2, 1, 1);
imhist(inputImage);
title('Histogram of Original Image');
subplot(2, 1, 2);
imhist(outputImage);
```

Histogram Matching

Reference image:

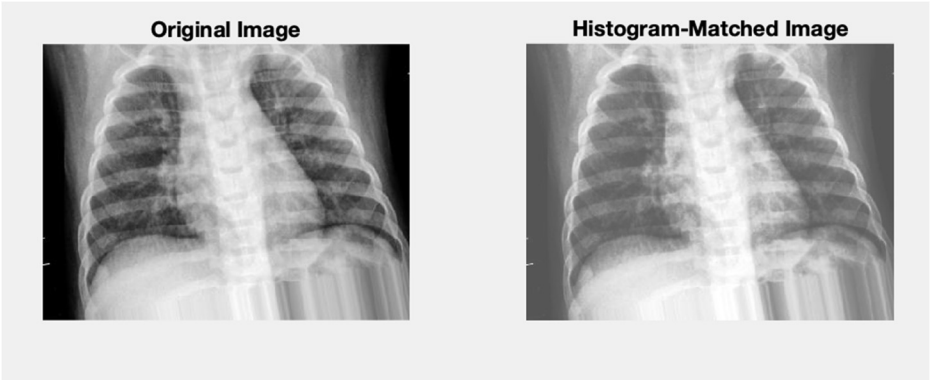


Bacterial Pneumonia



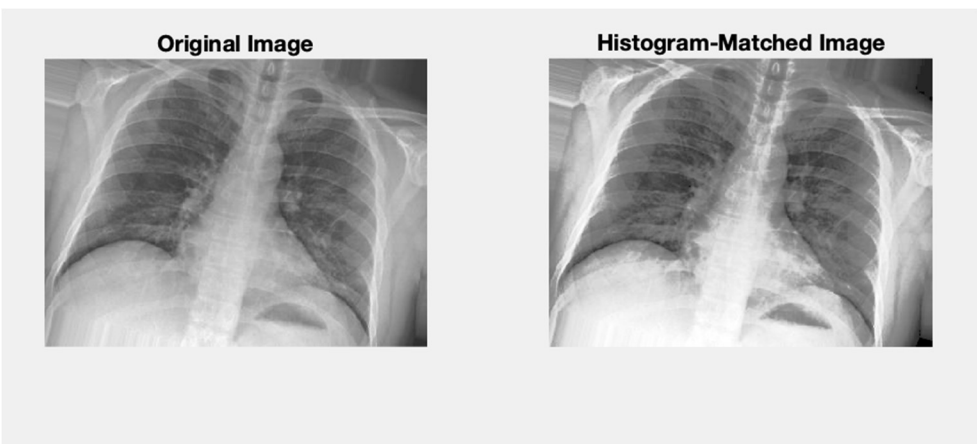

```
inputImage = imread('/Users/nithylesh/Downloads/imgs/BacterialPneumonia.jpeg');
referenceImage = imread('/Users/nithylesh/Downloads/imgs/histogramref/refimg.jpeg');
referenceHistogram = imhist(referenceImage);
outputImage = histeq(inputImage, referenceHistogram);
subplot(1, 2, 1);
imshow(inputImage);
title('Original Image');
subplot(1, 2, 2);
imshow(outputImage);
```

Viral Pneumonia



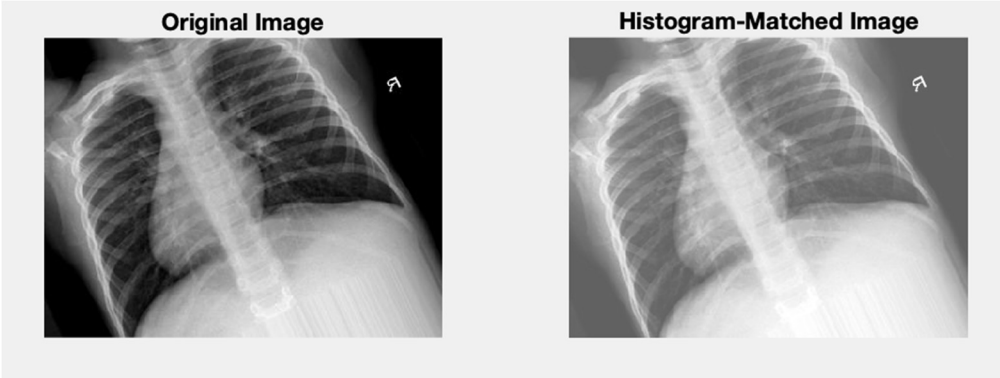

```
>> inputImage = imread('/Users/nithylesh/Downloads/imgs/ViralPneumonia.jpeg');
referenceImage = imread('/Users/nithylesh/Downloads/imgs/histogramref/refimg.jpeg');
referenceHistogram = imhist(referenceImage);
outputImage = histeq(inputImage, referenceHistogram);
subplot(1, 2, 1);
imshow(inputImage);
title('Original Image');
subplot(1, 2, 2);
imshow(outputImage);
title('Histogram-Matched Image')
```

COVID 19




```
>> inputImage = imread('/Users/nithylesh/Downloads/imgs/COVID_19.jpeg');
referenceImage = imread('/Users/nithylesh/Downloads/imgs/histogramref/refimg.jpeg');
referenceHistogram = imhist(referenceImage);
outputImage = histeq(inputImage, referenceHistogram);
subplot(1, 2, 1);
imshow(inputImage);
title('Original Image');
subplot(1, 2, 2);
imshow(outputImage);
title('Histogram-Matched Image')
```

Normal




```
>> inputImage = imread('/Users/nithylesh/Downloads/imgs/Normal.jpeg');
referenceImage = imread('/Users/nithylesh/Downloads/imgs/histogramref/refimg.jpeg');
referenceHistogram = imhist(referenceImage);
outputImage = histeq(inputImage, referenceHistogram);
subplot(1, 2, 1);
imshow(inputImage);
title('Original Image');
subplot(1, 2, 2);
imshow(outputImage);
title('Histogram-Matched Image')
```

Inference:

In histogram equalization it darkens the disease free regions and and lightens the diseased regions such that the diseased regions much more clearer than the diseased regions in the original image.

(5) Spatial Domain Filters

BOX Filter(Smoothening)

Box Filter

For Covid

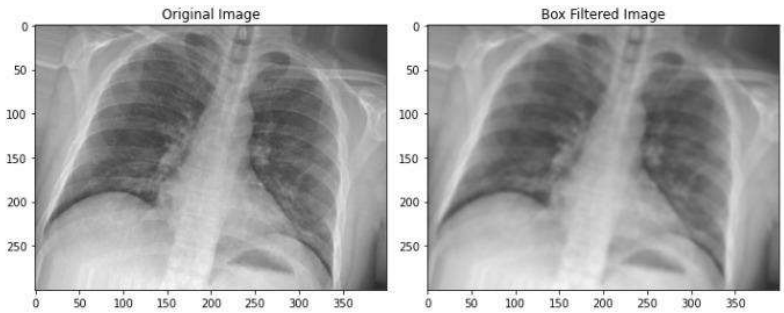
```
kernel_size = (5, 5)  # Change the size as needed

# Apply the box filter
filtered_image = cv2.boxFilter(covid_img, -1, kernel_size)

# Display the original and filtered images
plt.figure(figsize=(10, 4))
plt.subplot(1, 2, 1)
plt.title('Original Image')
plt.imshow(cv2.cvtColor(covid_img, cv2.COLOR_BGR2RGB))

plt.subplot(1, 2, 2)
plt.title('Box Filtered Image')
plt.imshow(cv2.cvtColor(filtered_image, cv2.COLOR_BGR2RGB))

plt.tight_layout()
plt.show()
```



For Bacterial Pneumonia

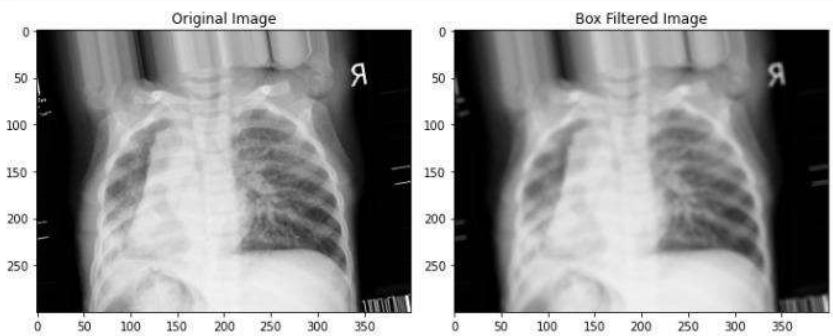
```
kernel_size = (5, 5)  # Change the size as needed

# Apply the box filter
filtered_image = cv2.boxFilter(bpneumonia_img, -1, kernel_size)

# Display the original and filtered images
plt.figure(figsize=(10, 4))
plt.subplot(1, 2, 1)
plt.title('Original Image')
plt.imshow(cv2.cvtColor(bpneumonia_img, cv2.COLOR_BGR2RGB))

plt.subplot(1, 2, 2)
plt.title('Box Filtered Image')
plt.imshow(cv2.cvtColor(filtered_image, cv2.COLOR_BGR2RGB))

plt.tight_layout()
plt.show()
```



For Viral Pneumonia

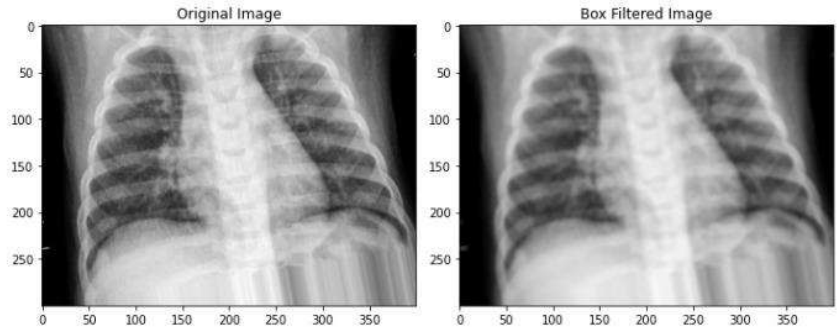
```
kernel_size = (5, 5)  # Change the size as needed

# Apply the box filter
filtered_image = cv2.boxFilter(vpneumonia_img, -1, kernel_size)

# Display the original and filtered images
plt.figure(figsize=(10, 4))
plt.subplot(1, 2, 1)
plt.title('Original Image')
plt.imshow(cv2.cvtColor(vpneumonia_img, cv2.COLOR_BGR2RGB))

plt.subplot(1, 2, 2)
plt.title('Box Filtered Image')
plt.imshow(cv2.cvtColor(filtered_image, cv2.COLOR_BGR2RGB))

plt.tight_layout()
plt.show()
```



For Normal

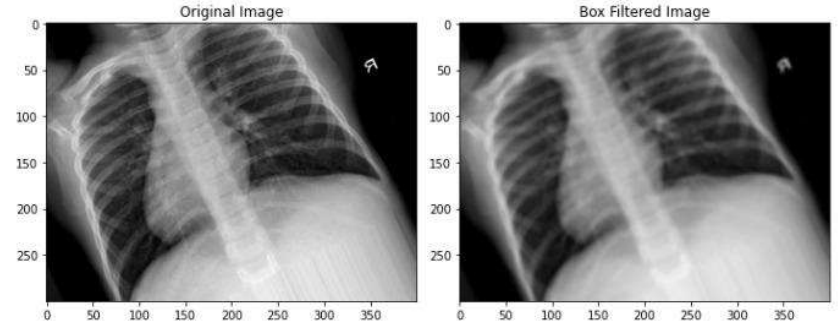
```
kernel_size = (5, 5)  # Change the size as needed

# Apply the box filter
filtered_image = cv2.boxFilter(normal_img, -1, kernel_size)

# Display the original and filtered images
plt.figure(figsize=(10, 4))
plt.subplot(1, 2, 1)
plt.title('Original Image')
plt.imshow(cv2.cvtColor(normal_img, cv2.COLOR_BGR2RGB))

plt.subplot(1, 2, 2)
plt.title('Box Filtered Image')
plt.imshow(cv2.cvtColor(filtered_image, cv2.COLOR_BGR2RGB))

plt.tight_layout()
plt.show()
```

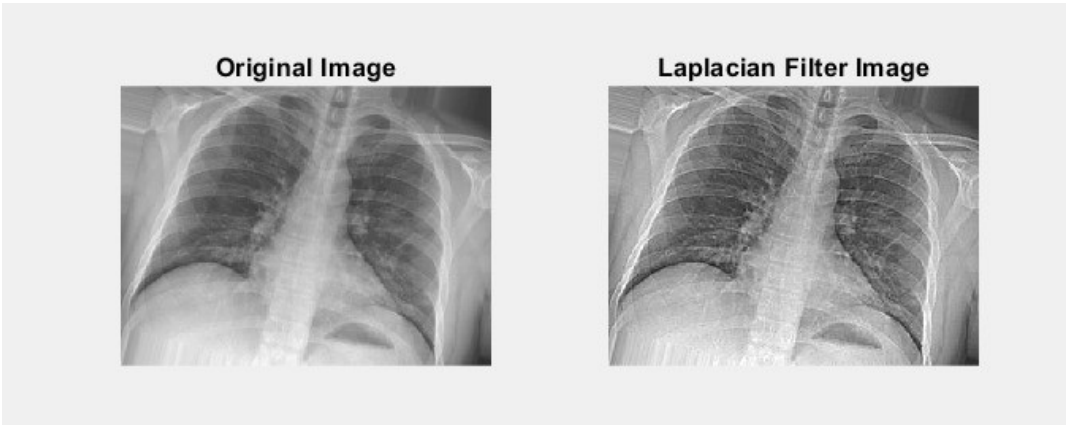


Inference:

Spatial filtering using a box (or averaging) filter is a common image processing technique. In this method, a rectangular neighborhood (the "box") is applied to each pixel in an image, and the pixel's value is replaced by the average value of the pixels within that neighborhood. It gives you a blur / smoothing effect but in our case we need small details to find the differences between normal and infected lungs so we can not use an average filter.

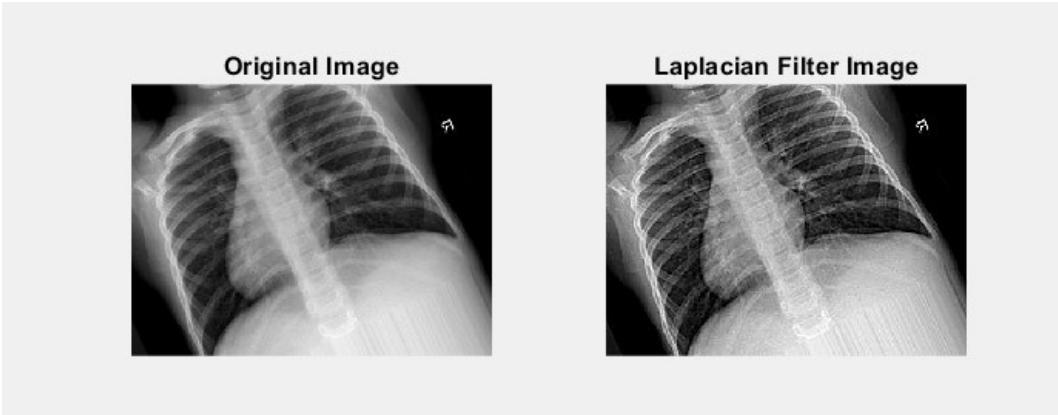
Laplacian of Gaussian Filter(Sharpening)

Bacterial Pneumonia



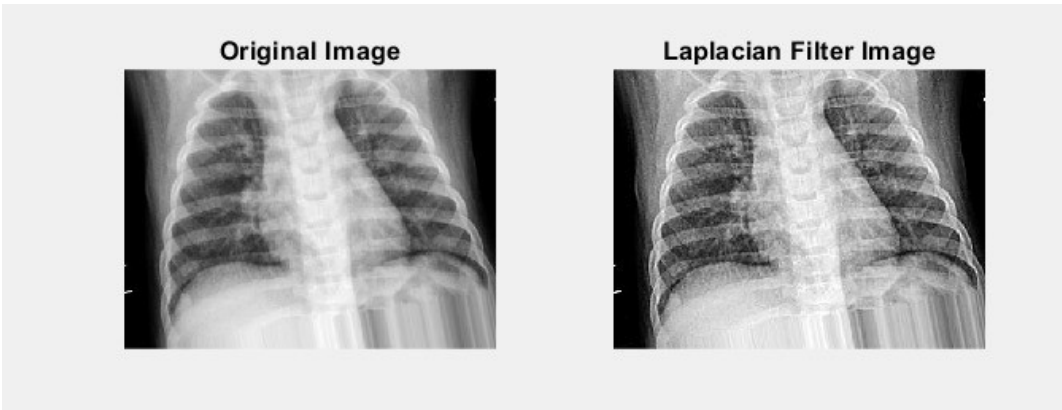

```
sharpeningAmount = 1.5;
sharpeningRadius = 2.0;
sharpenedImage = imsharpen(Bacteria,'Amount', sharpeningAmount, 'Radius', sharpeningRadius);
figure;
subplot(1, 2, 1), imshow(Bacteria), title('Original Image');
subplot(1, 2, 2), imshow(sharpenedImage), title('Laplacian Filter Image');
```

COVID_19



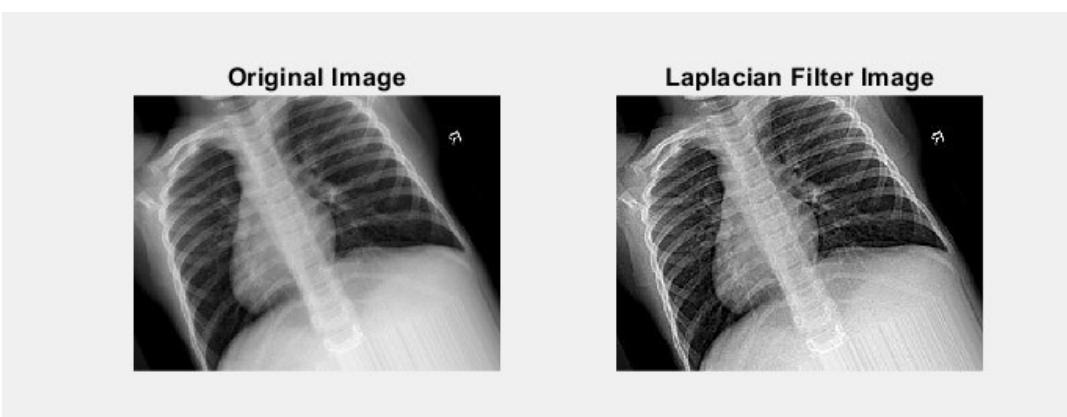
```
sharpeningAmount = 1.5;
sharpeningRadius = 2.0;
sharpenedImage = imsharpen(covid,'Amount', sharpeningAmount, 'Radius', sharpeningRadius);
figure;
subplot(1, 2, 1), imshow(covid), title('Original Image');
subplot(1, 2, 2), imshow(sharpenedImage), title('Laplacian Filter Image');
```

Viral Pneumonia




```
sharpeningAmount = 1.5;
sharpeningRadius = 2.0;
sharpenedImage = imsharpen(viral,'Amount', sharpeningAmount, 'Radius', sharpeningRadius);
figure;
subplot(1, 2, 1), imshow(viral), title('Original Image');
subplot(1, 2, 2), imshow(sharpenedImage), title('Laplacian Filter Image');
```

Normal



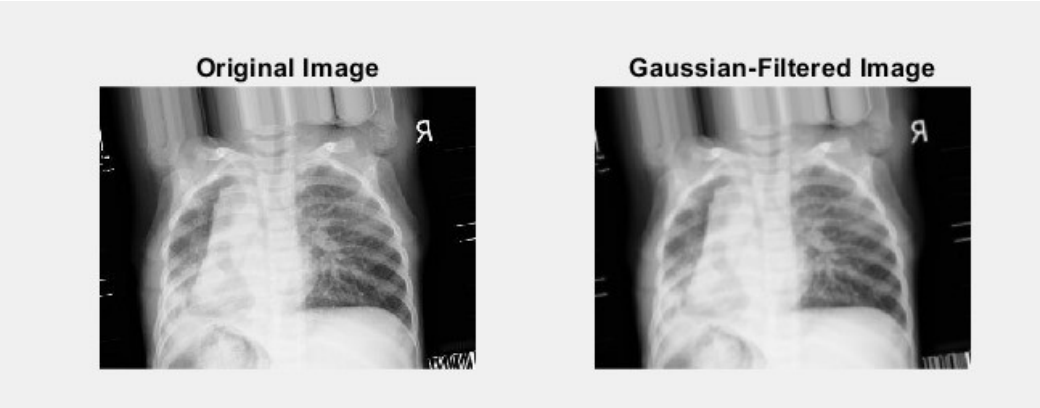
```
>> Normal=imread("C:\Users\Praveen nallasivam\Desktop\codes\New\Lung Cancer\imgs\Normal.jpeg");
sharpeningAmount = 1.5;
sharpeningRadius = 2.0;
sharpenedImage = imsharpen(Normal,'Amount', sharpeningAmount, 'Radius', sharpeningRadius);
figure;
subplot(1, 2, 1), imshow(Normal), title('Original Image');
subplot(1, 2, 2), imshow(sharpenedImage), title('Laplacian Filter Image');
```

Inference:

Laplacian sharpening is a powerful technique for enhancing edges and fine details in images, But in our case it sharpens the bones and brings some noise which will give you errors while training the model. We need the noise to be as little as possible to find the differences between normal and infected lungs so we can not use Laplacian filters.

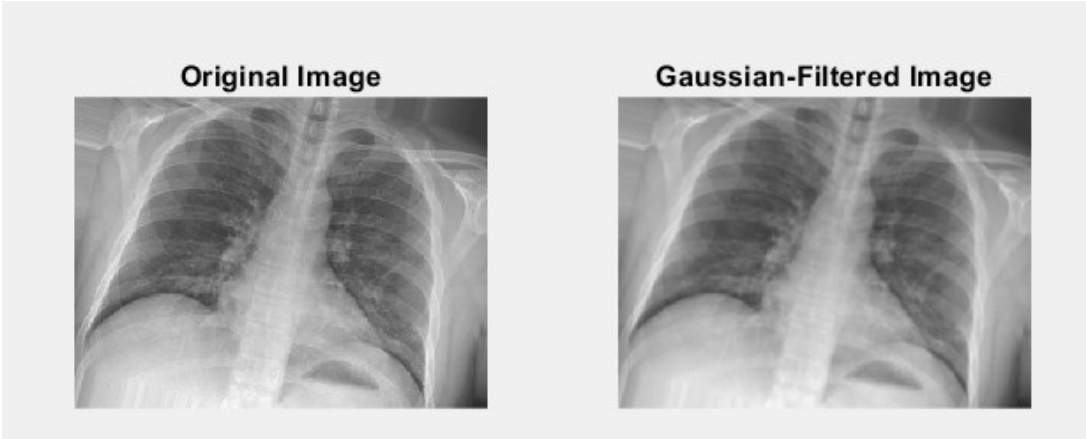
Gauss Filtering

Bacterial Pneumonia




```
sigma = 1;
gaussianFiltered = imgaussfilt(Bacteria, sigma);
figure;
subplot(1, 2, 1); imshow(Bacteria); title('Original Image');
subplot(1, 2, 2); imshow(gaussianFiltered); title('Gaussian-Filtered Image');
```

COVID_19

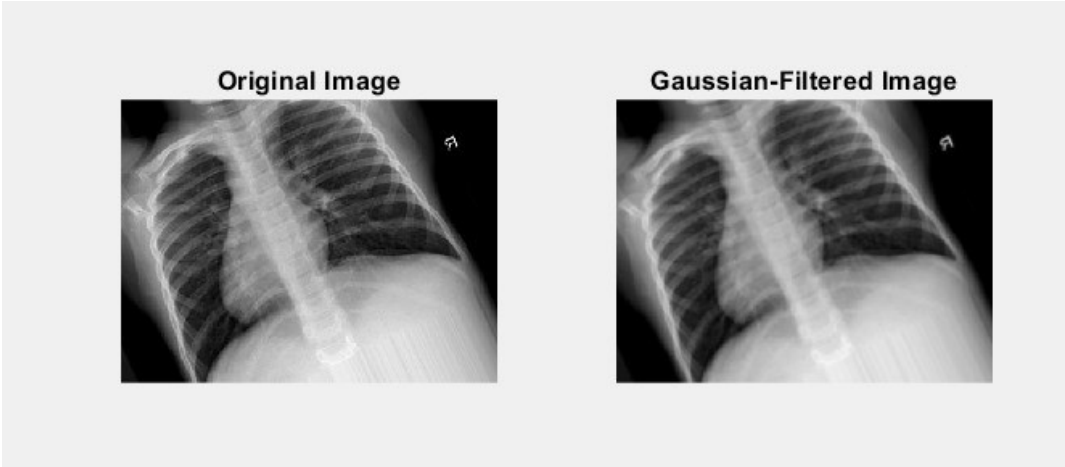



```
sigma = 1;
gaussianFiltered = imgaussfilt(covid, sigma);
figure;
subplot(1, 2, 1); imshow(covid); title('Original Image');
subplot(1, 2, 2); imshow(gaussianFiltered); title('Gaussian-Filtered Image');
```

Viral Pneumonia

```
sigma = 1;
gaussianFiltered = imgaussfilt(viral, sigma);
figure;
subplot(1, 2, 1); imshow(viral); title('Original Image');
subplot(1, 2, 2); imshow(gaussianFiltered); title('Gaussian-Filtered Image');
```

Normal



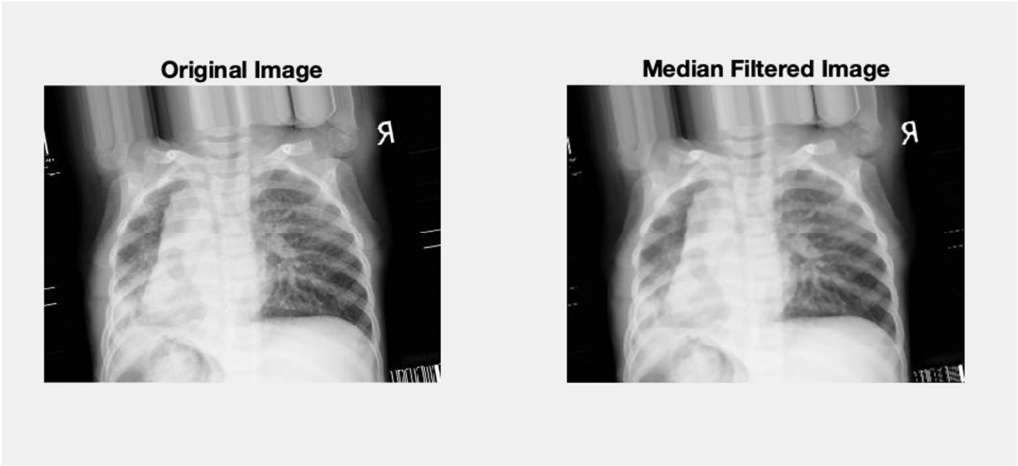

```
sigma = 1;
gaussianFiltered = imgaussfilt(Normal, sigma);
figure;
subplot(1, 2, 1); imshow(Normal); title('Original Image');
subplot(1, 2, 2); imshow(gaussianFiltered); title('Gaussian-Filtered Image');
```

Inference:

Spatial filtering using a Gaussian filter is a common image processing technique used for various purposes, including smoothing, noise reduction, and image enhancement.It gives you a blur / smoothing effect but in our case we need small details to find the differences between normal and infected lungs so we can not use an gaussian filter.

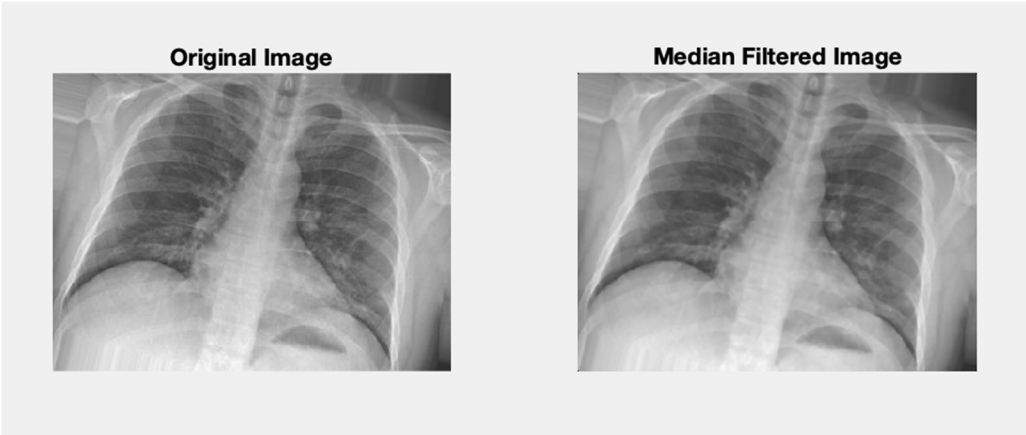
Median Filtering

Bacterial Pneumonia



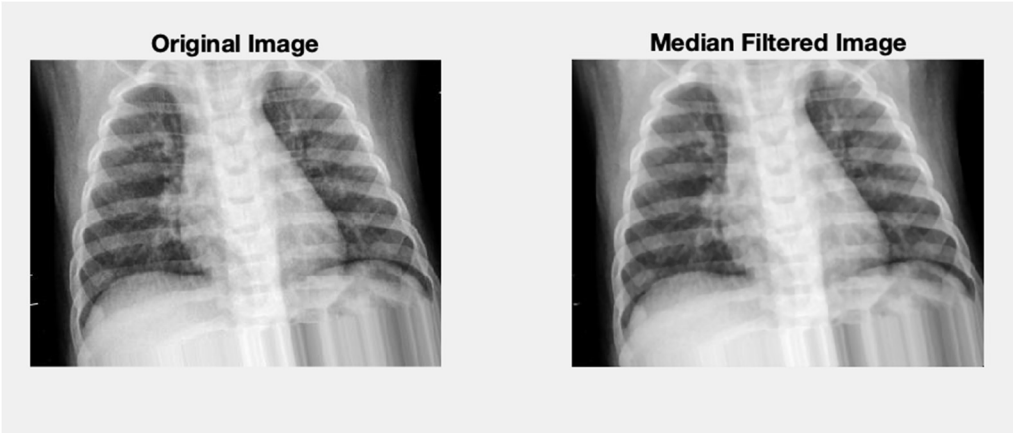
```
inputImage = imread('/Users/nithylesh/Downloads/imgs/BacterialPneumonia.jpeg');
windowSize = [3, 3];
outputImage = medfilt2(inputImage, windowSize);
subplot(1, 2, 1);
imshow(inputImage);
title('Original Image');
subplot(1, 2, 2);
imshow(outputImage);
title('Median Filtered Image');
```

COVID_19



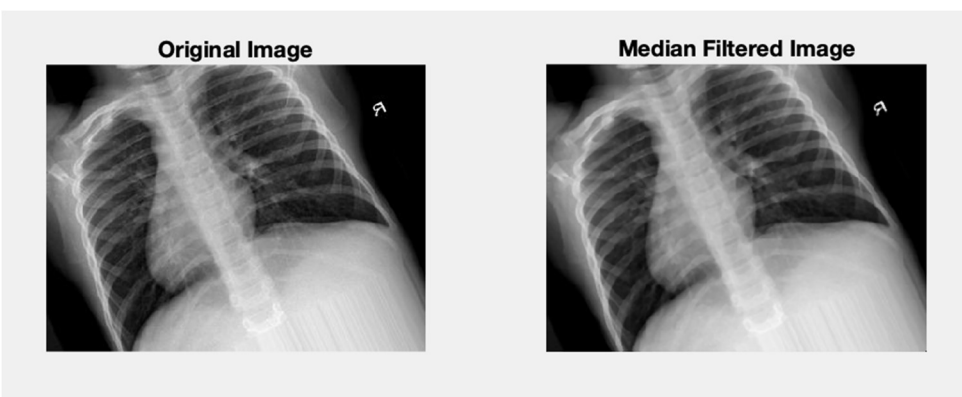

```
>> inputImage = imread('/Users/nithylesh/Downloads/imgs/COVID_19.jpeg');
windowSize = [3, 3];
outputImage = medfilt2(inputImage, windowSize);
subplot(1, 2, 1);
imshow(inputImage);
title('Original Image');
subplot(1, 2, 2);
imshow(outputImage);
title('Median Filtered Image');
```

Viral Pneumonia




```
>> inputImage = imread('/Users/nithylesh/Downloads/imgs/ViralPneumonia.jpeg');
windowSize = [3, 3];
outputImage = medfilt2(inputImage, windowSize);
subplot(1, 2, 1);
imshow(inputImage);
title('Original Image');
subplot(1, 2, 2);
imshow(outputImage);
title('Median Filtered Image');
```

Normal




```
>> inputImage = imread('/Users/nithylesh/Downloads/imgs/Normal.jpeg');
windowSize = [3, 3];
outputImage = medfilt2(inputImage, windowSize);
subplot(1, 2, 1);
imshow(inputImage);
title('Original Image');
subplot(1, 2, 2);
imshow(outputImage);
title('Median Filtered Image');
```

Inference:

median filtering is a valuable spatial filtering technique known for its effectiveness in noise reduction, particularly in the presence of salt-and-pepper noise. Here in our case this

filtering method is the best fit because it doesn't remove too much details or add noise to the image.

Final Inference:

We have chosen Median filter as the final one because it removes salt and pepper noise which will cause error if present while training the model, in case of Box filtering it blurs the image resulting in loss of small details which is not preferable, in case of gaussian filter it is also an smoothing filter which blurs the image which is not required in our case, in case of sharpening it sharpens noise which would reduce the accuracy of the model.

(6) Inference based on the results:

After doing all these filters Median filter gives us the best result combined with gamma transformation. Median Filter removes salt and pepper noise and gamma transformation enhances the contrast and brightness which improves the visibility of diseased regions in X-Ray of lungs.