Histogram Equalization

This technique is used for increasing contrast in an image. The goal is to redistribute gray levels among pixels so that the resulting histogram is flat. First, the histogram of the image is computed which represents the pixels intensities. The cumulative distribution function is then calculated from the histogram followed by the intensity transformation which is the normalizing the intensity range to cover the entire intensity range. finally, each pixel in the image is replaced with its corresponding intensity in the normalized cumulative distribution function. This results in an image with improved contrasts as the intensities spread out more evenly across the entire range. Darker area become darker and lighter area become lighter, leading to enhanced details and appealing image.

Log transform.

This technique is also a point wise operation. Each pixel is transformed with the formula $f(u) = c * \log(1 + |u|)$. u is the pixel and c is a constant to scale the transformation. It is a dynamic range compression technique that results in brightness enhancement. It enhances the visibility of low intensity details image particularly in the darker region. This is because the logarithmic function compresses the high intensity values more than the low intensity value.

Image rotation

This is a fundamental operation in image processing that involves rotating an image around its center point. The process of rotating image involves transforming each pixel's coordinates according to a specific rotation angle. A transformation matrix is defined which is then applied to each coordinate. This technique is derived from principles of linear algebra and trigonometry

Gaussian filtering

It is widely used to reduce noise and smooth images. It operates by convolving the images with a gaussian kernel or mask. The values or weights on the mask or kernel are computed from a Gaussian function. Convolution operations are often represented by asks with which image is convolved. It is done by

1. place the mask on pixel(i,j) for which the results need to be computed
2. compute the wise point wise multiplication of all pixels of the image and mask that overlap
3. sum up the products
4. replace pixel(i,j) value with the computed sum of products
5. slide the mask over to the next pixel and repeat the operation.

By convolving the image with the Gaussian kernel, the noise is smoothed out, leading to a cleaner appearance. Fine details in the image, such as small texture patterns or sharp transitions, are smoothed out. This is achieved because the Gaussian kernel's weighting emphasizes nearby pixels over distant ones, resulting in a blending effect that reduces sharp transitions. One of the key advantages of Gaussian blur over other blurring techniques is its ability to preserve edges in the image. While smoothing out noise and fine details, Gaussian blur tends to preserve the edges' sharpness and contrast, maintaining the image's overall structure and visual coherence. The degree of blurring can be controlled by adjusting the standard deviation (sigma) parameter of the Gaussian kernel. A larger sigma value results in more significant blurring, while a smaller sigma value preserves more detail in the image

Median filtering

The goal of this technique is to reduce noise in an image while preserving edges and details. The input pixel is replaced by the median value in a specific window. The size of the window determines the neighborhood considered for computing the median. Larger window can effectively suppress the noise but may also blur the image more. It is effective for removing noise such as the salt and pepper noise without blurring the image excessively. Below are some purpose of the median blur

1. **Noise Reduction**: Similar to Gaussian blur, median blurring effectively reduces noise in the image. By replacing each pixel's value with the median value of its neighborhood, outliers and noisy pixels are removed, resulting in a smoother appearance.
2. **Preservation of Edges**: Unlike Gaussian blur, median blurring does not affect edges as much. Since the median operation selects the central value in the sorted neighborhood, sharp transitions and edges are preserved, maintaining the image's structural integrity.
3. **Detail Preservation**: Median blurring is effective in preserving details in the image. While smoothing out noise, it retains fine details and texture patterns since the median operation does not introduce blurring artifacts or loss of detail.

4. **Artifact Reduction**: Median blurring is particularly useful for removing certain types of noise, such as salt-and-pepper noise, which appears as isolated bright or dark pixels in the image. The median operation replaces these outliers with values closer to the true pixel intensity, resulting in a cleaner image without introducing blurring artifacts.

Analysis/Interpretation

Histogram Equalization: The intensity range was stretched which resulted in a more balanced distribution of pixel intensities across the histogram, hereby enhancing the contrast of the image. For example, In the tire image, regions were already dark became darker and light regions became brighter. This improvement made the subtle features in the image more visible and distinguishable. This technique can be used in medical imaging to enhance the visibility of anatomical structures in X-ray, MRI, or Ct scans making it easier for medical personnel to detect abnormalities. Other areas include but not limited to satellite imaging, enhancement of old photographs etc. The results were the same when compared to standard tools.

Log transformation:

A log transformation is a type of point operation used in image processing to enhance the dynamic range of an image. It applies a logarithmic function to each pixel intensity value in the image. It affected the image by enhancing the contrasts making the subtle part of the image more visible. I also notice that the scaling factor c allows for the adjustment of the overall brightness of the transformed image. By modifying this parameter, you can control the overall brightness level to achieve the desired visual effect. Finally, when compared to standard tools, the results were also the same.

Rotation: The function takes an image a rotate it with some angle given as a parameter. It affects the image by producing a new image rotated at some specific angle. When comparing my implementation to the standard tool, I noticed that my results was different. I defined the center of my image as the center and rotated around that. I believe that this may be the reason for difference in results. Although, I do not know how the standard package tools implemented theirs.

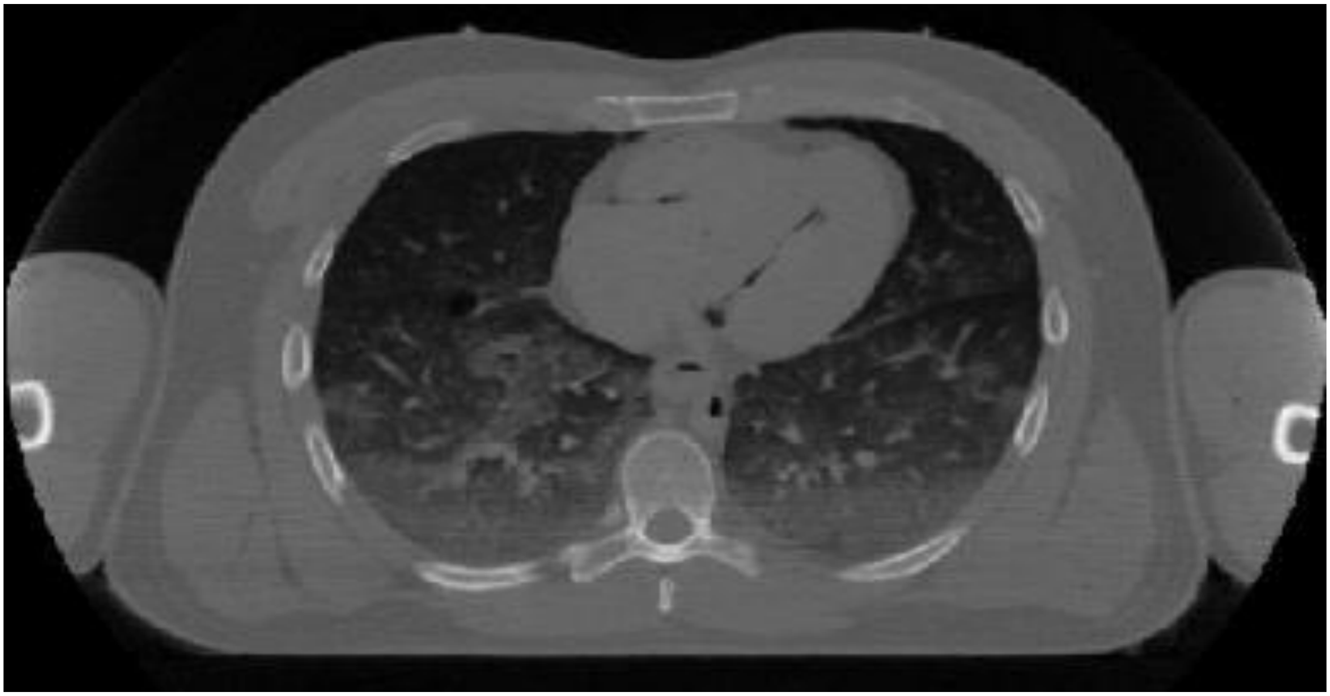
Gaussian Blur: This function applies a convolution operation to the image using a Gaussian kernel, which is a 2D matrix with values that follow a Gaussian distribution. It affects the image by reducing the noise. Gaussian blur effectively reduces high-frequency noise present in the image, such as salt-and-pepper noise or Gaussian noise. The resulting image became somewhat blurry. When applied to remove enhance an image affected by salt and pepper degradation, it proved effective. The results were also the same when compared to standard tools.

Median Blur: Unlike the Gaussian, instead of calculating the weighted average of neighboring pixels like Gaussian blur, median blurring replaces each pixel's value with the median value of its neighborhood. The resulting image looked somewhat blurry. When applied to remove enhance an image affected by salt and pepper degradation, it proved effective. The results were also the same when compared to standard tools.

Original Images

Car.pnm

Tire.pnm



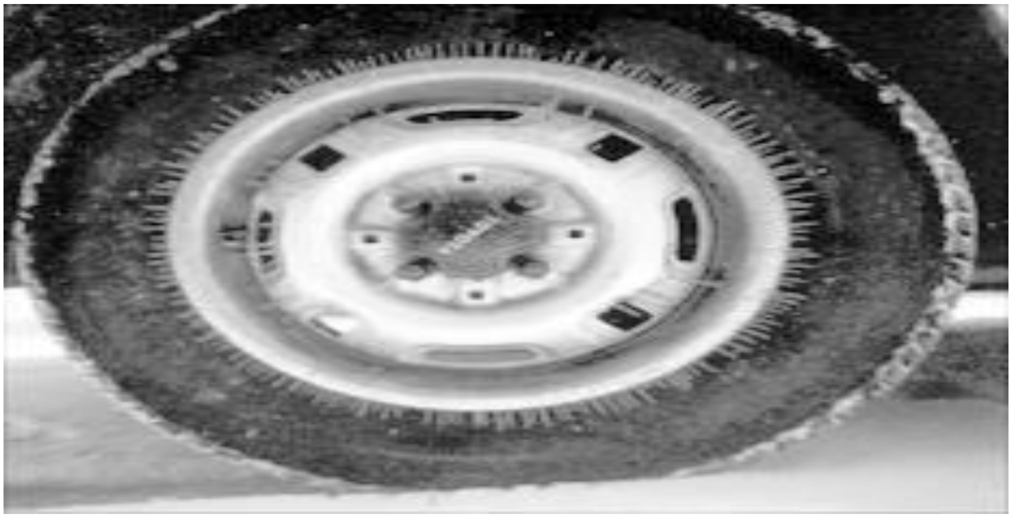
Ct_scan.pnm

Child.pnm

Building.pnm

Effects of Histogram Equalization on Images: Comparison with standard tools

I compare my implementation with standard python packages.



My implementation

Standard package implementation

My implementation

Standard package implementation


My implementation


Standard package implementation

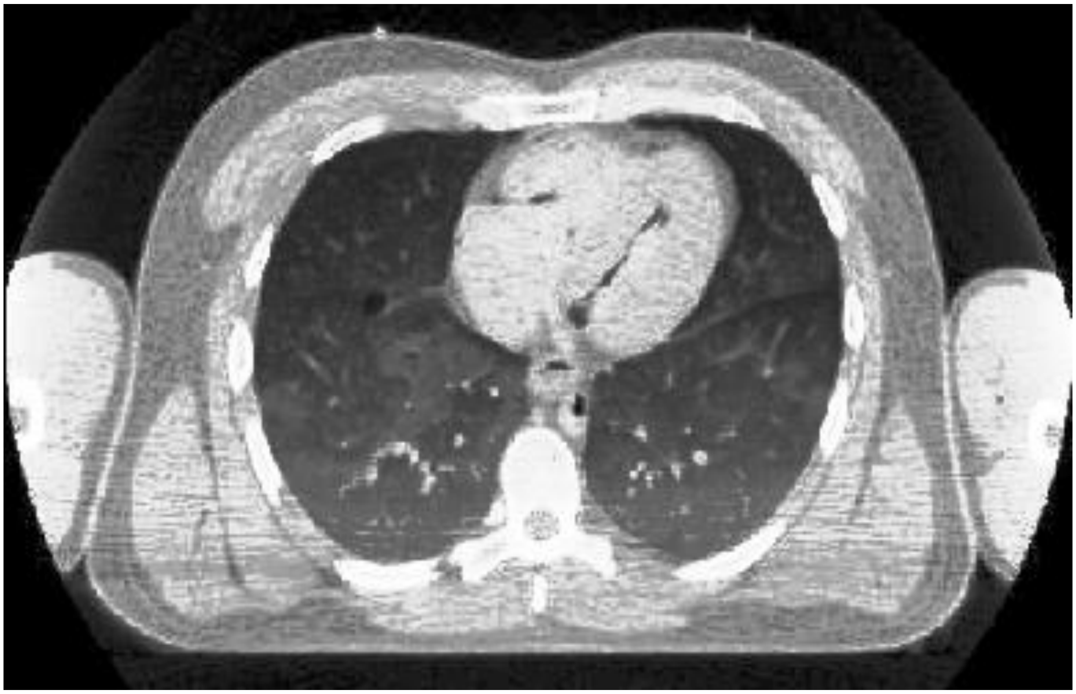
Original

My implementation

Original

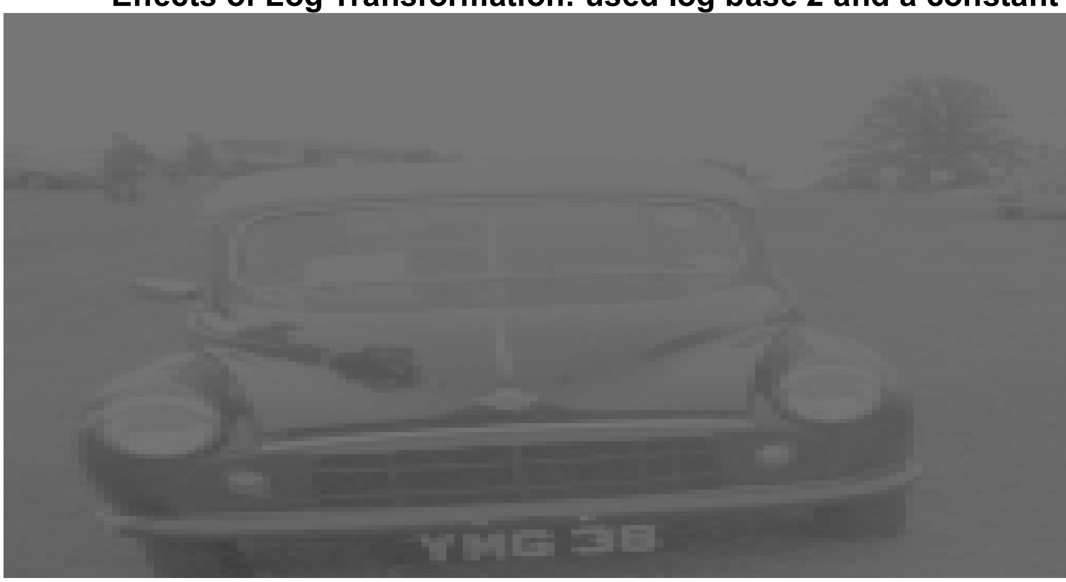
Standard package implementation


My implementation



Standard package implementation

Effects of Log Transformation: used log base 2 and a constant of 15



My implementation

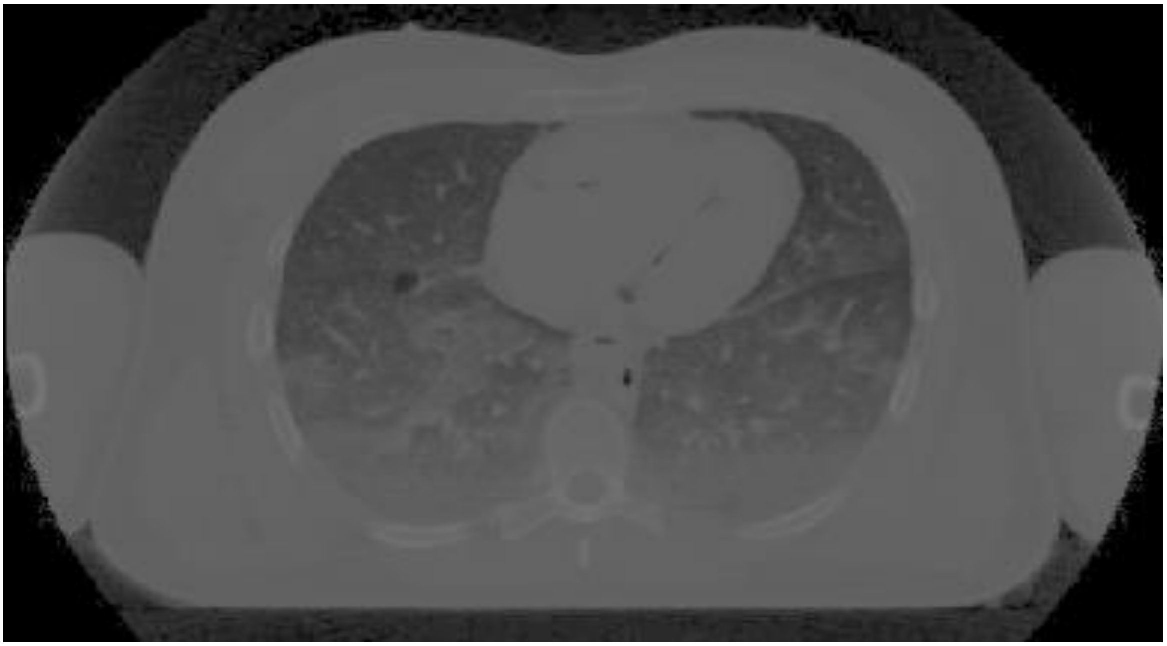

Standard package implementation


My implementation

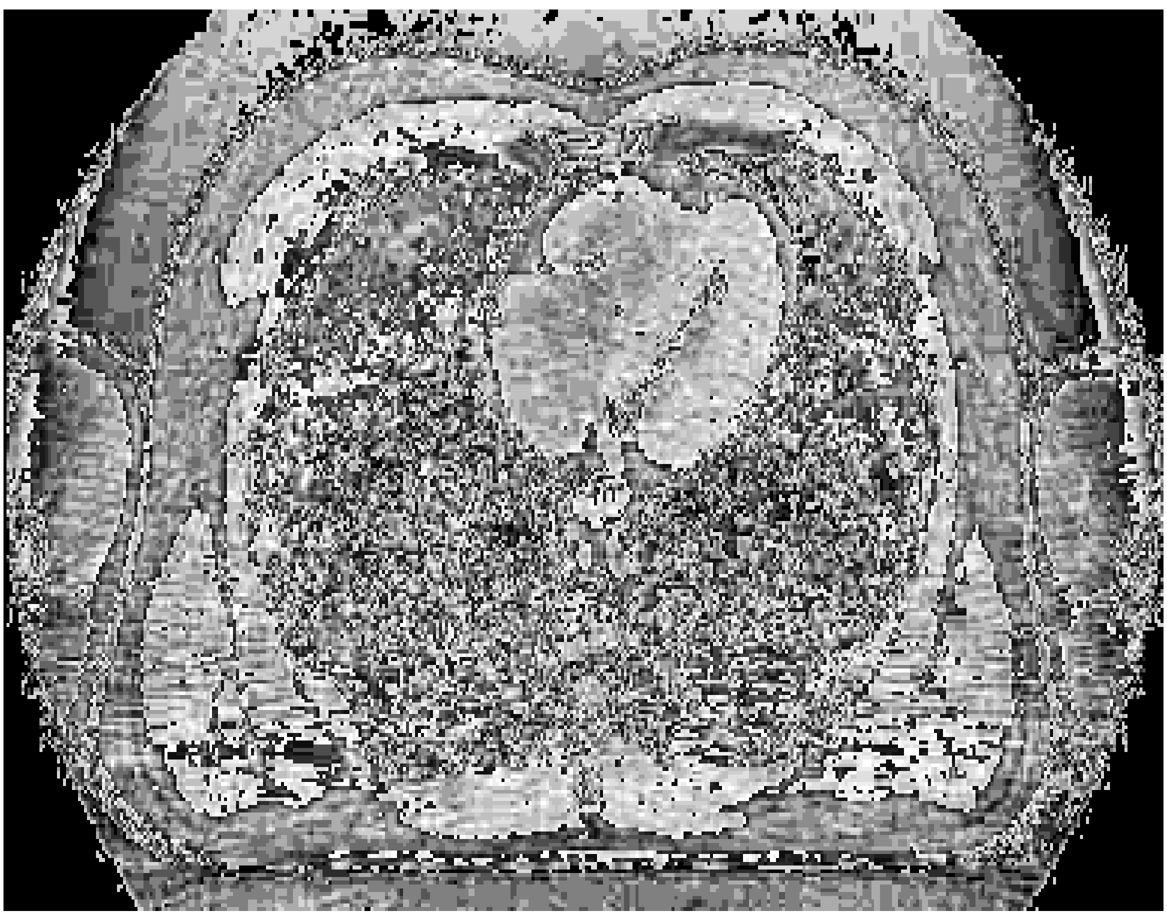

Standard package implementation

My implementation

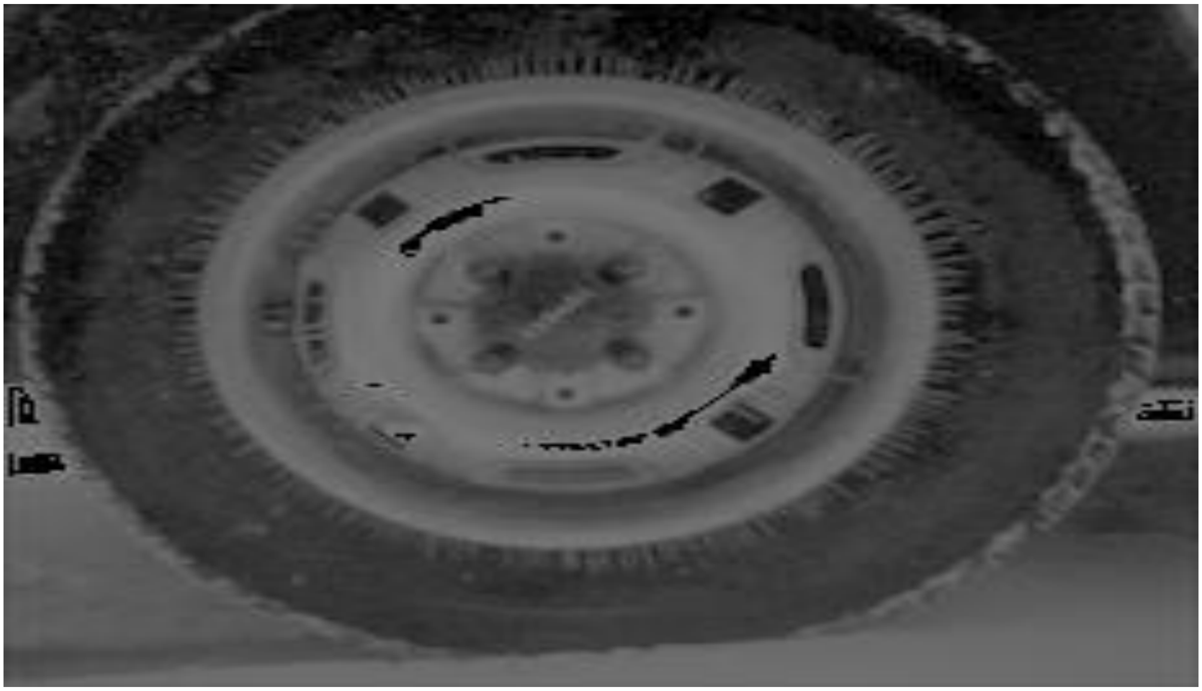
Standard package implementation



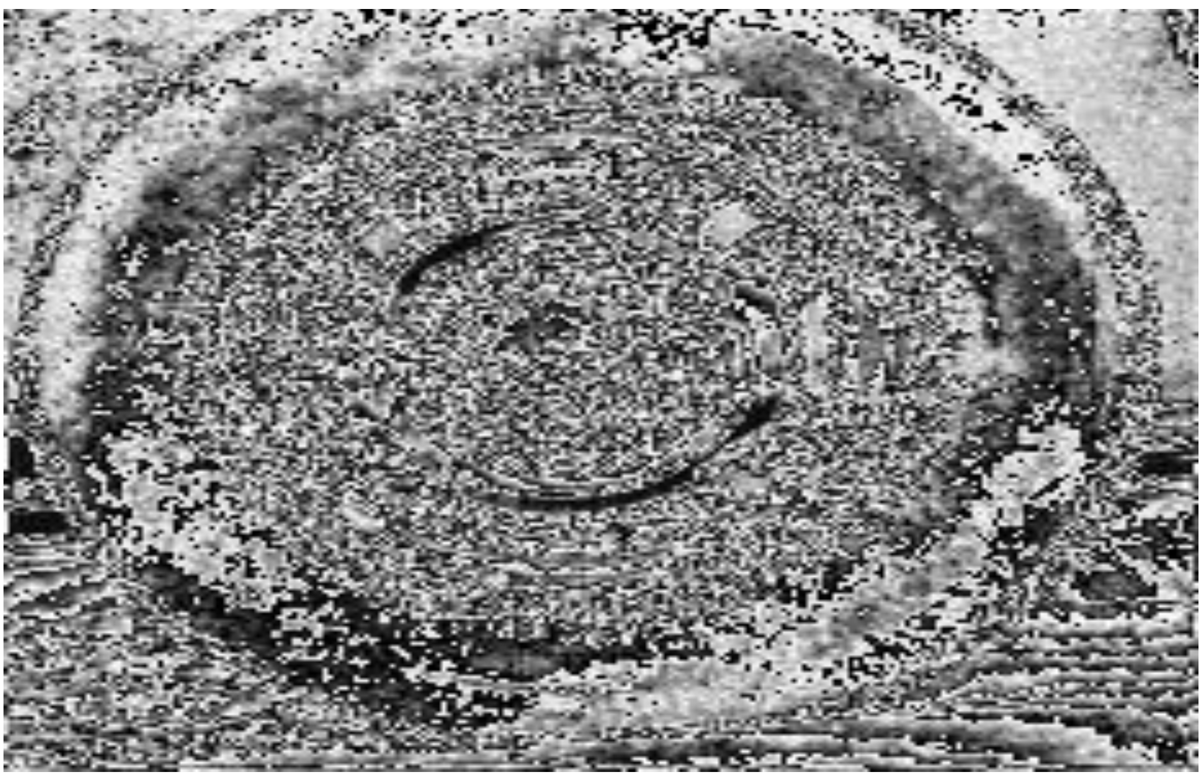
My implementation



Standard package Implementation



My implementation

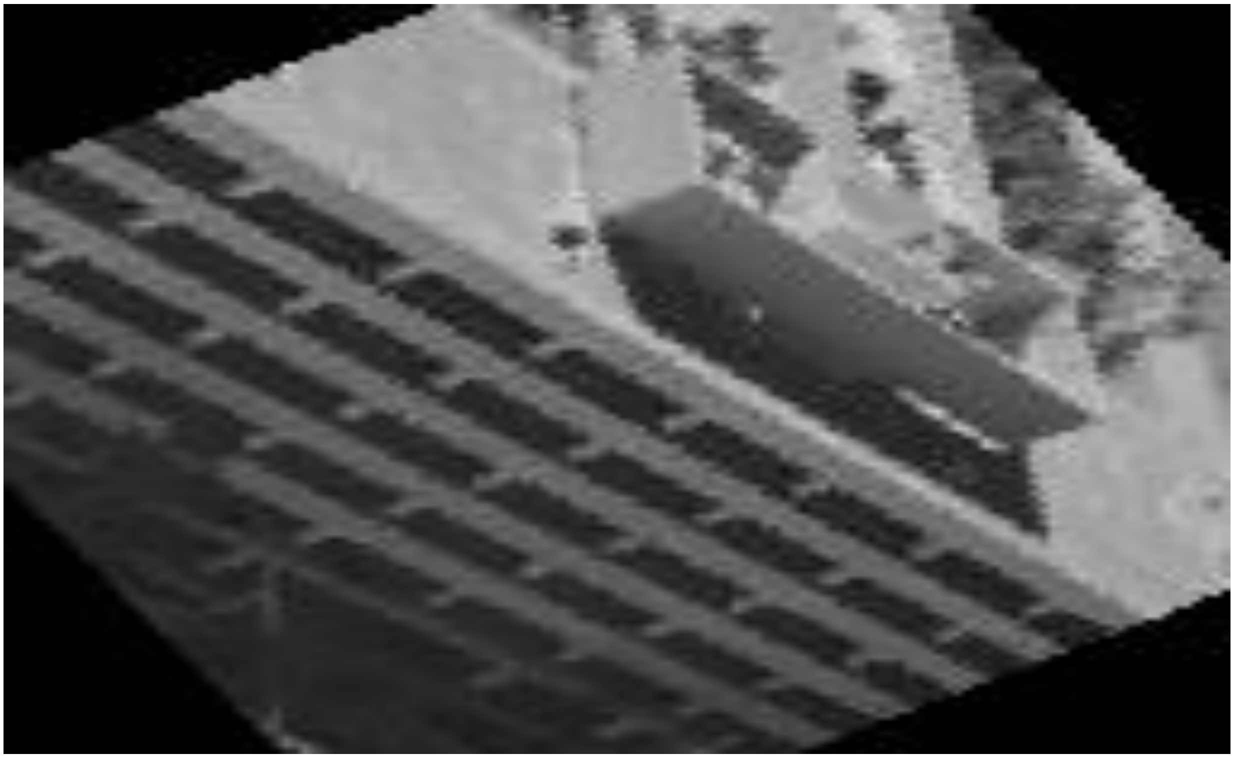


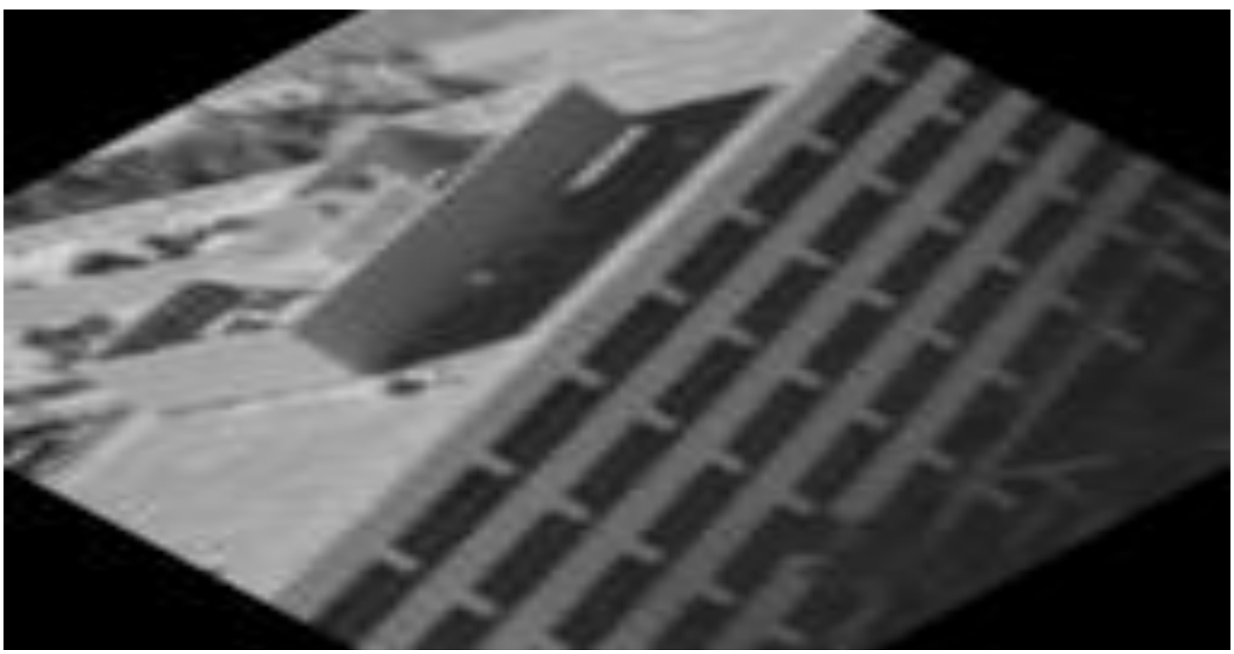
Standard package implementation

Rotation: Comparison of my implementation versus python packages (I rotated by 45 degrees)

My implementation

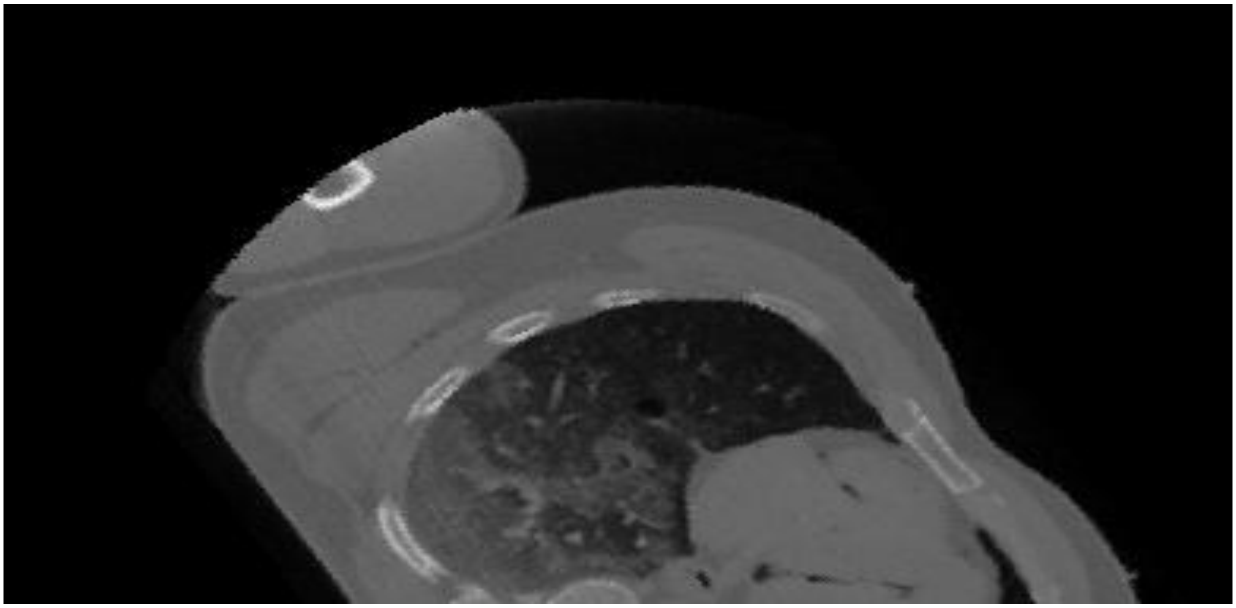
Python Package

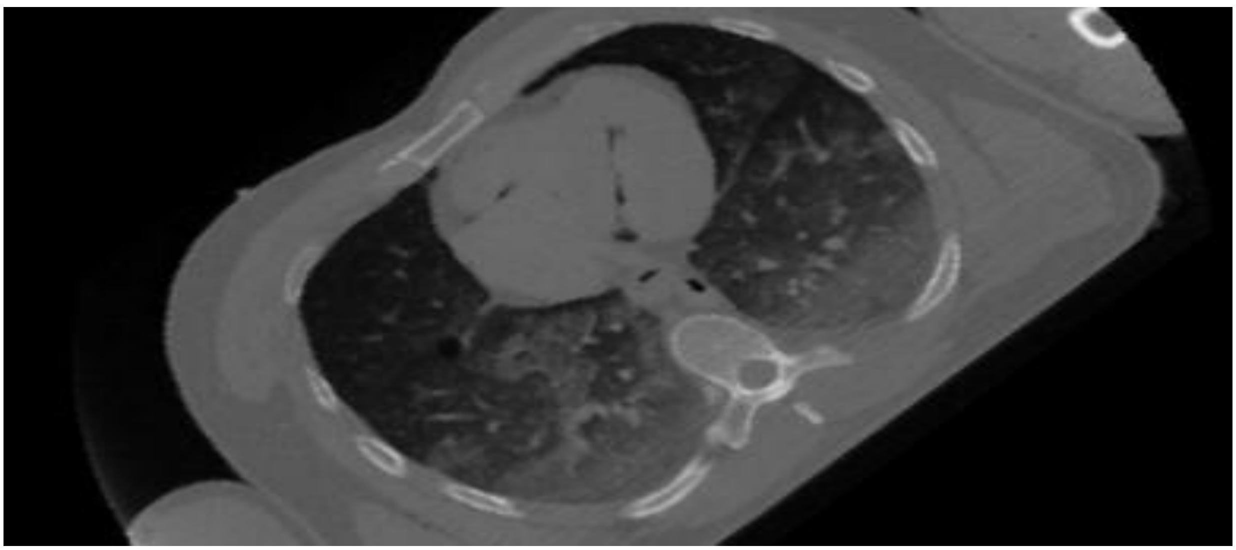

My implementation

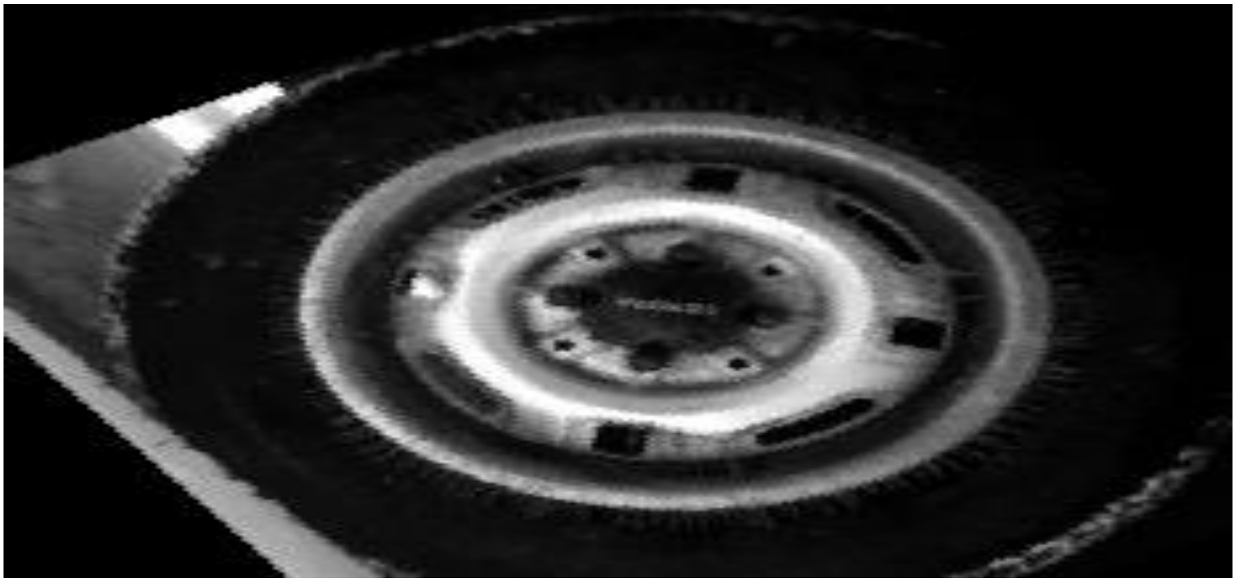

Python package

My implementation

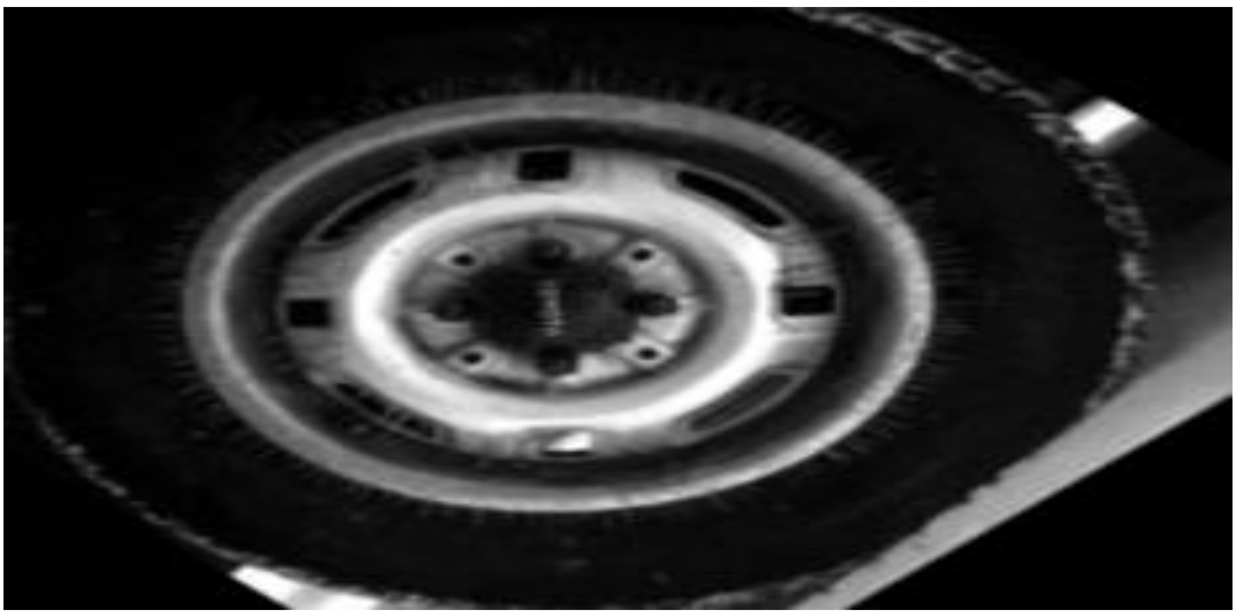
Python package


My implementation


Python Package



My Implementation



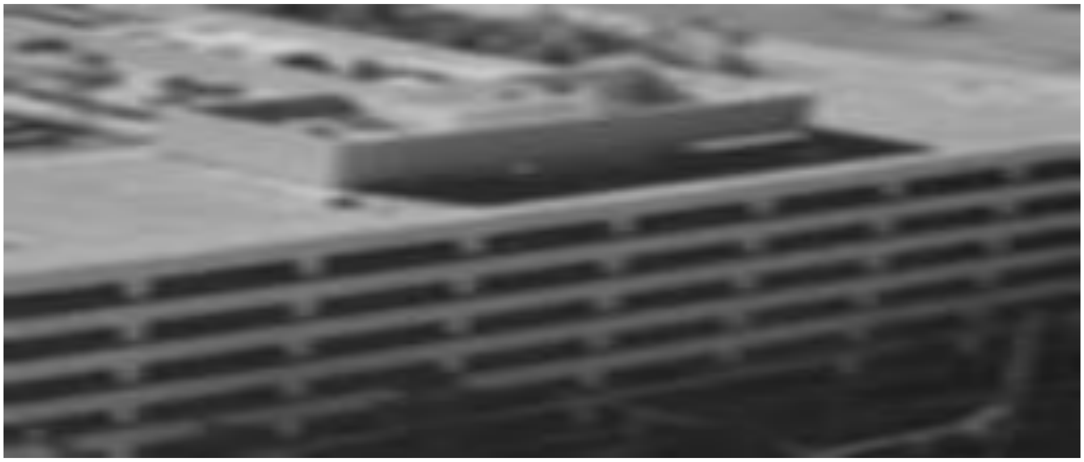
Python Package

Effects of Gaussian Blur: I compare my implementation versus python packages (I used a 3 by 3 kernel and sigma of 15)


My implementation


Python Package

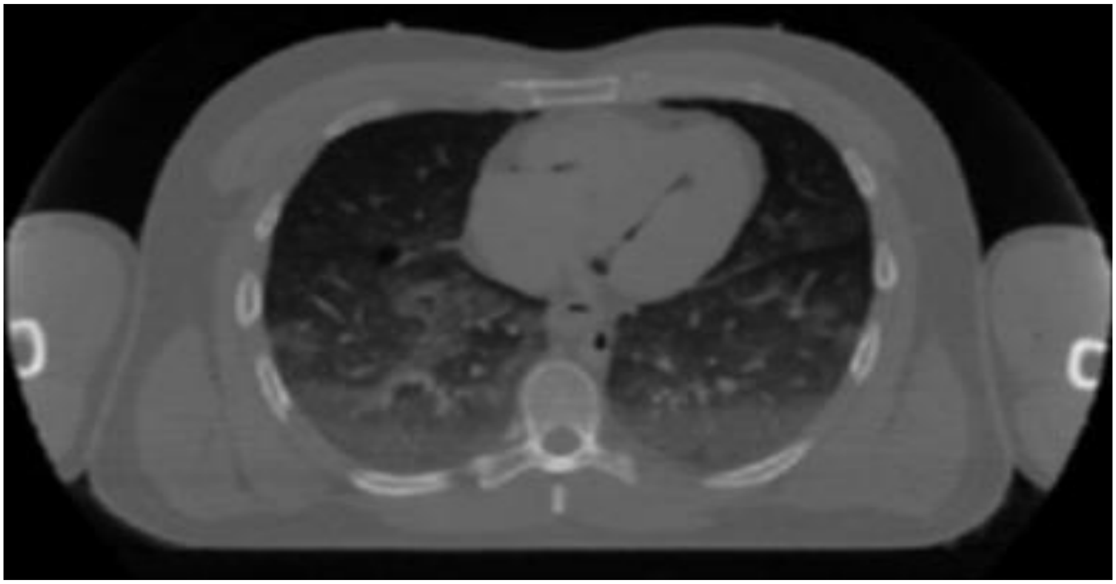

My implementation



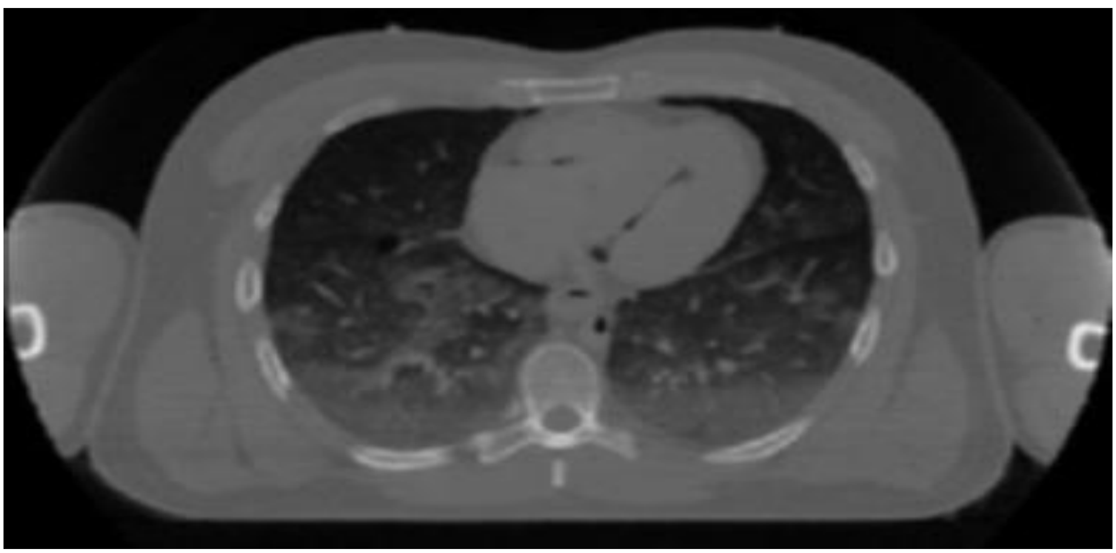
Python Package

My Implementation

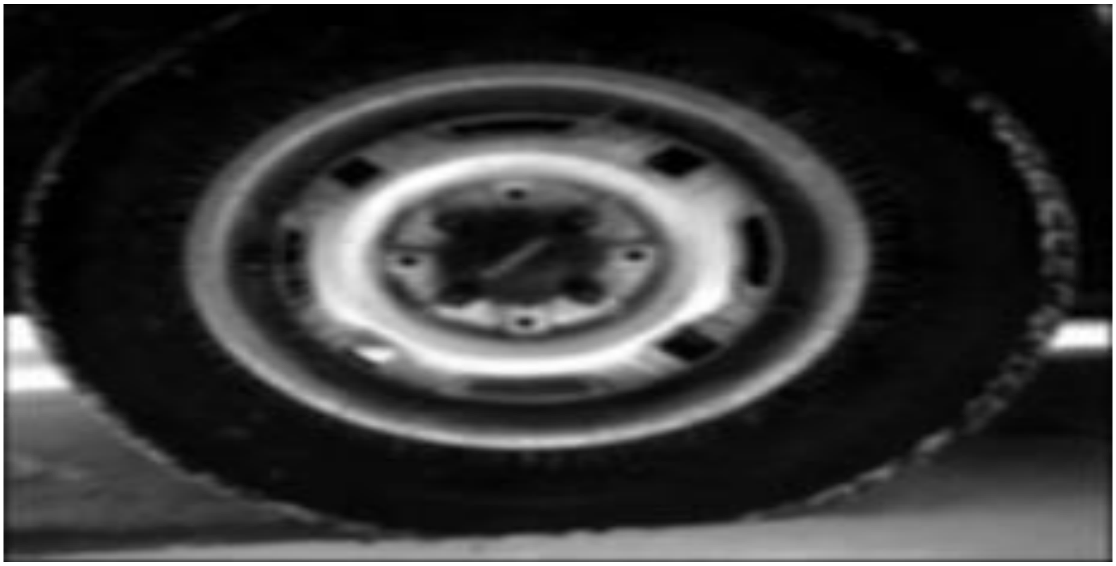
Python Package



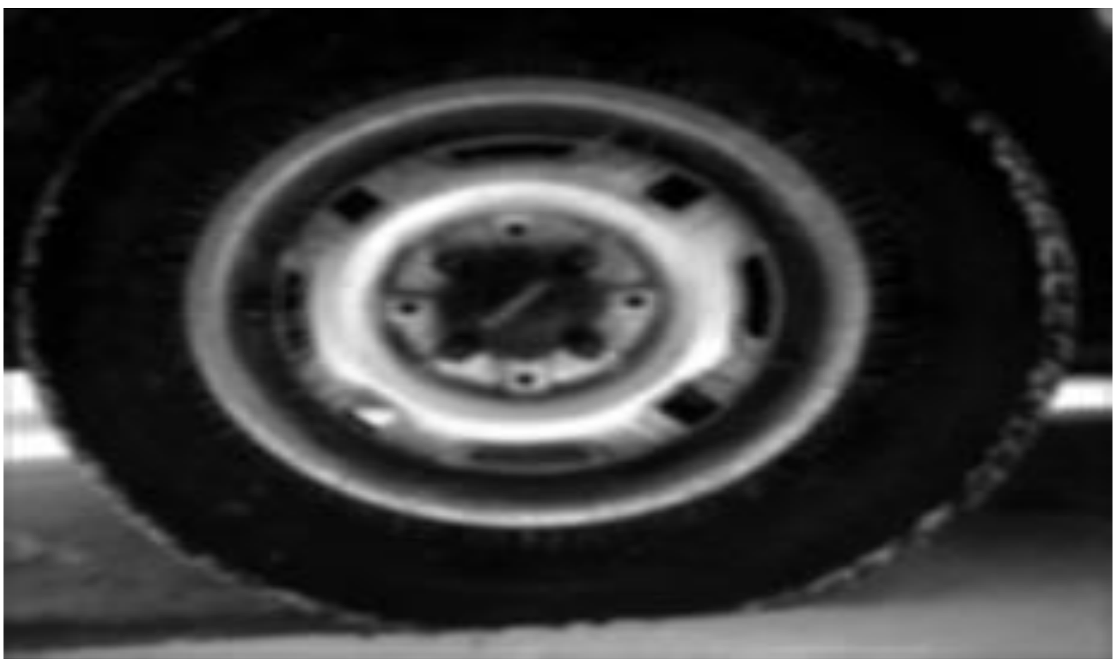
My Implementation



Python Package



My Implementation



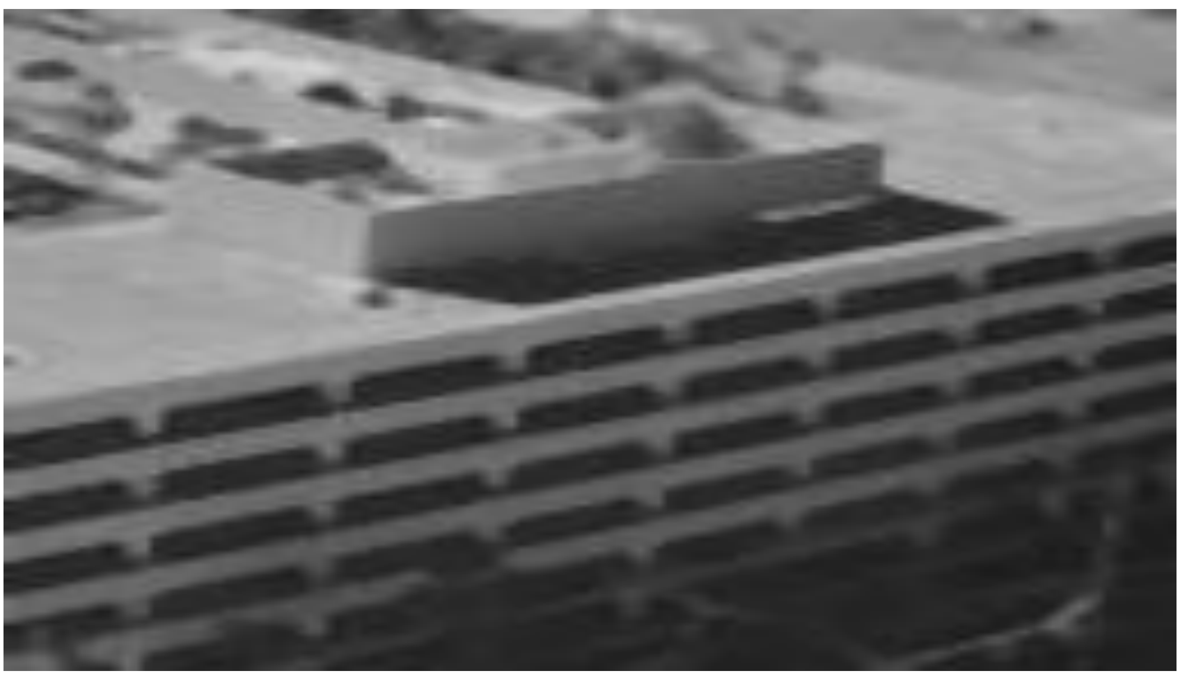
Python Package

Effects of Median Blur: I used a 3 by 3 window.

My Implementation

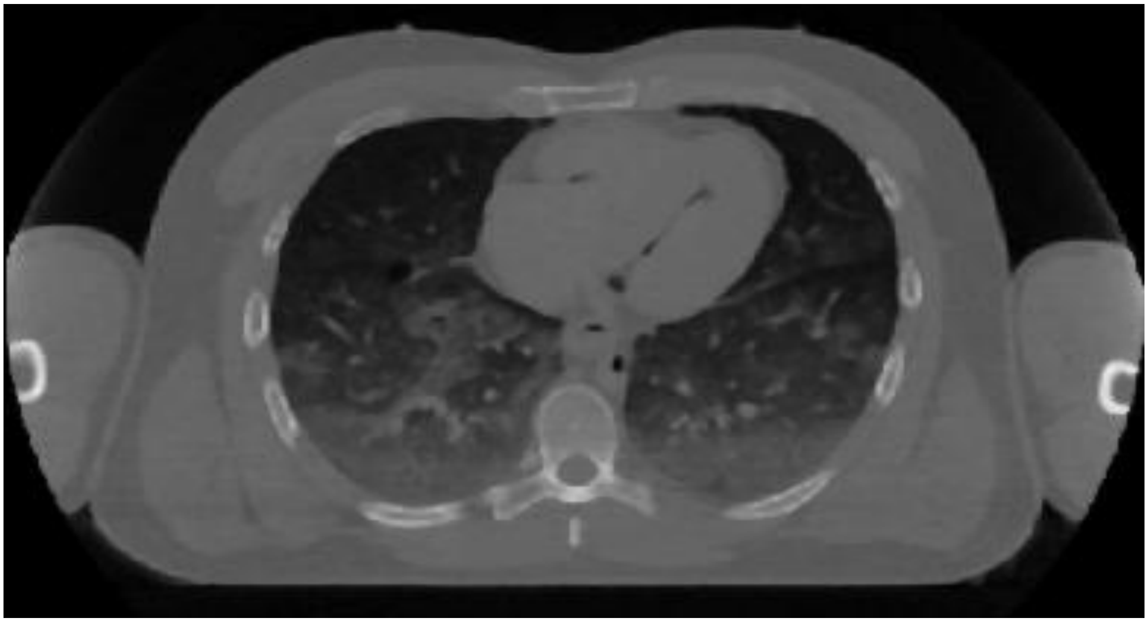
Python Package


My Implementation

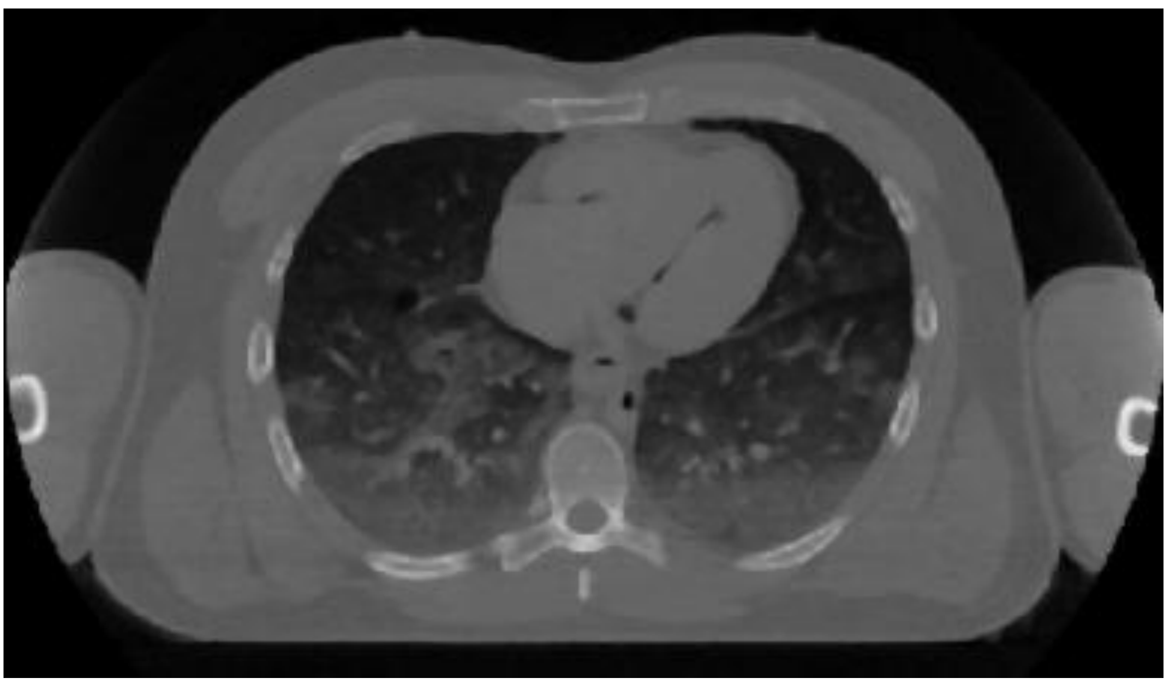

Python Package

My Implementation

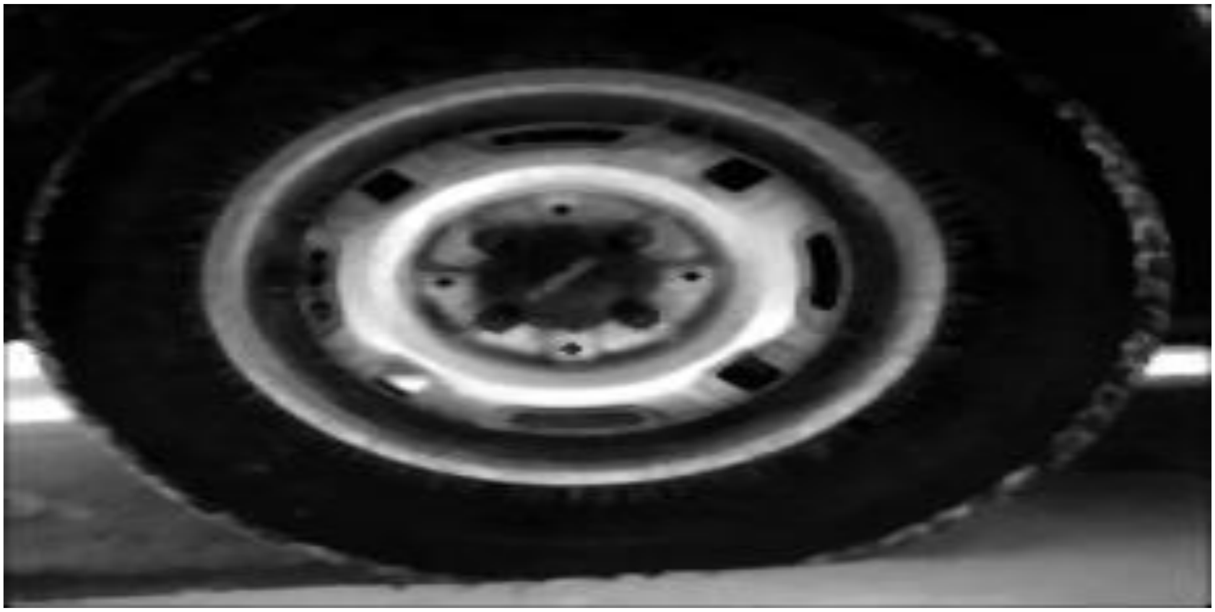
Python Package



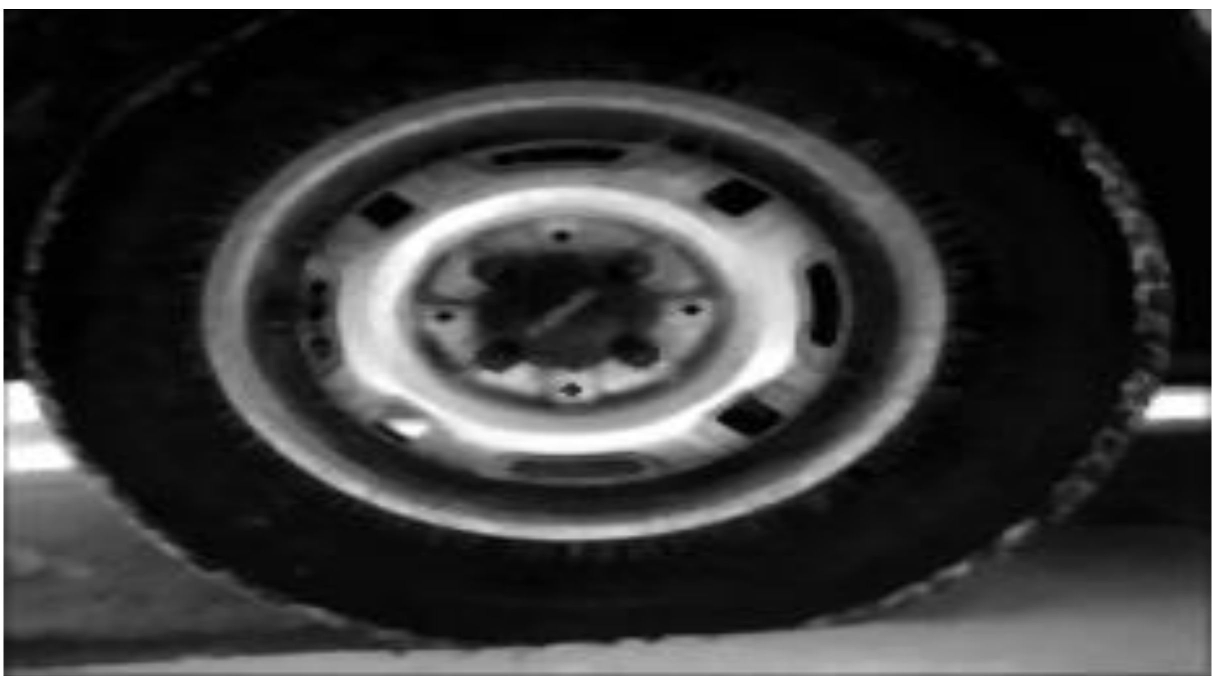
My Implementation



Python Package



My Implementation



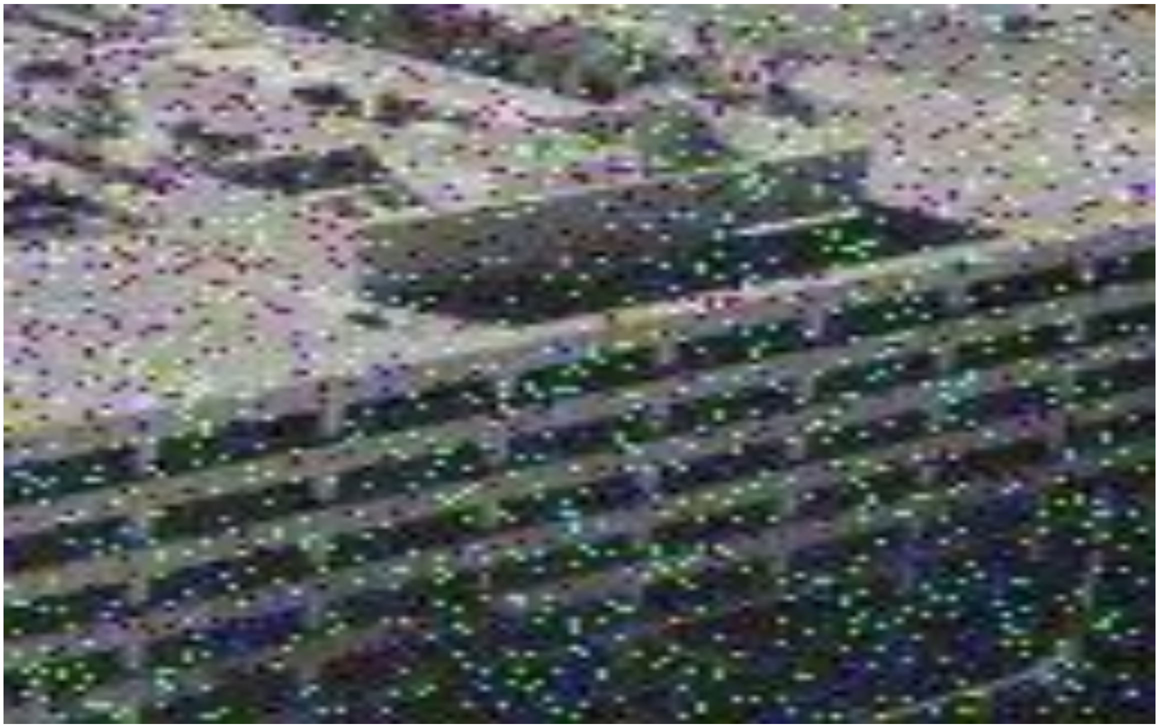
Python Package

Experiment with the gaussian blur and median blur.

I chose the building picture (building.pnm) and I added the salt and pepper noise. I then used the gaussian technique to remove the degradation. Overall, the techniques were somewhat effective or very effective. Below are the results.

Salt Probability (salt_prob): This parameter specifies the probability of adding salt noise to each pixel in the image. Salt noise occurs when pixels are randomly set to white (maximum intensity value, typically 255).

Pepper Probability (pepper_prob): This parameter specifies the probability of adding pepper noise to each pixel in the image. Pepper noise occurs when pixels are randomly set to black (minimum intensity value, typically 0).
Here, they were both set to 0.5 and 0.5
I used for 3 by 3 kernel and sigma of 15 for the gaussian blur and a 3 by 3 window. I found that this to be acceptable and better compared to other parameters that I chose.



Degraded Image: salt and pepper

Gaussian blur for restoration/enhancement


Median blur for restoration/enhancement

Application of the techniques: Where it is useful and Where Not:

Histogram equalization, log transformation, rotation of image, Gaussian blur, and median blur are fundamental image processing techniques with diverse applications and specific advantages. Histogram equalization is particularly useful for enhancing image contrast and improving the visual appearance of images with uneven illumination or limited dynamic range. It redistributes pixel intensities to maximize the use of available intensity levels, making it beneficial for improving the visibility of details in medical imaging, satellite imagery, and surveillance footage. Log transformation is valuable for adjusting the brightness and contrast of images with non-linear intensity distributions. It is commonly used in medical imaging to enhance the visibility of structures in X-ray, MRI, and CT scans, where pixel intensities may be concentrated in specific ranges.

Rotation of image is essential for reorienting images to correct alignment errors, adjust orientation, or facilitate further analysis. It finds applications in image registration, panorama stitching, object recognition, and augmented reality systems.

Gaussian blur is effective for reducing noise and smoothing images while preserving important details. It is widely employed in photography, video processing, and digital image editing to improve the aesthetics of images and facilitate subsequent analysis tasks.

Median blur, on the other hand, excels at removing salt-and-pepper noise without excessively blurring edges and fine details. It is indispensable in image denoising applications, particularly in medical imaging, microscopy, and surveillance, where noise reduction is critical for accurate diagnosis and analysis.

While these techniques offer valuable capabilities for image enhancement and preprocessing, they may not always be suitable in certain contexts. For example, histogram equalization may amplify noise in low-quality images, and Gaussian blur may introduce artifacts in images with fine textures. Additionally, rotation of images may distort important features if not applied carefully. Therefore, it is essential to consider the specific characteristics of the image data and the requirements of the application when selecting and applying these techniques.

Conclusion

In this project, we explored several fundamental images processing techniques, including histogram equalization, log transformation, image rotation, Gaussian blur, and median blur. Each technique offers unique capabilities and serves specific purposes in enhancing and manipulating digital images.

- Histogram Equalization: We observed that histogram equalization is a powerful method for enhancing the contrast and brightness of images by redistributing pixel intensities. It effectively stretches the dynamic range of pixel values, resulting in improved visual clarity and detail.

- Log Transformation: The log transformation function provides a flexible way to adjust the brightness and contrast of images, particularly for images with a wide range of pixel intensities. By applying a logarithmic function to pixel values, we observed enhancements in both shadow and highlight regions, leading to improved overall image quality.
- Image Rotation: Implementing image rotation allowed us to reorient images by specified angles, facilitating tasks such as alignment, correction, and orientation adjustment. I observed that rotation transforms preserve image content while altering its spatial orientation, enabling versatile image manipulation.
- Gaussian Blur: Gaussian blur is an effective technique for reducing noise and smoothing images by applying a weighted average across neighboring pixels. Our implementation demonstrated the ability of Gaussian blur to suppress high-frequency noise while preserving important image features, making it suitable for various image enhancement and preprocessing tasks.
- Median Blur: The median blur technique offers robust noise reduction capabilities by replacing pixel values with the median value of neighboring pixels. I observed that median blur is particularly effective in removing salt-and-pepper noise and preserving image details, making it a valuable tool for denoising applications.

In conclusion, the implemented image processing techniques provide essential tools for enhancing, manipulating, and preprocessing digital images in various domains, including computer vision, medical imaging, and multimedia applications. By understanding the principles and applications of these techniques, we can leverage them to improve image quality, extract meaningful information, and enable advanced image analysis tasks.