Image Restoration – A Use Case

Thai Quang Phat

Supervisor: Prof. Nguyen An Khuong, PhD.Candidate Nguyen Hoang Minh

November 23, 2024

Outline

- Problem
- Comments
- Procedure
- Experimental Results
- Conclusion
- Limitations & Future Work

Problem Statement

Given a blurred image of a heart, the task is to recover the degraded function. It is known that, the bottom right corner crosshair image before degraded, is 3 pixels wide, 30 pixels long, and had an intensity of 255. Provide a step-by-step procedure indicating how you would use the information just given to obtain the blurring function $H(u,v)$.

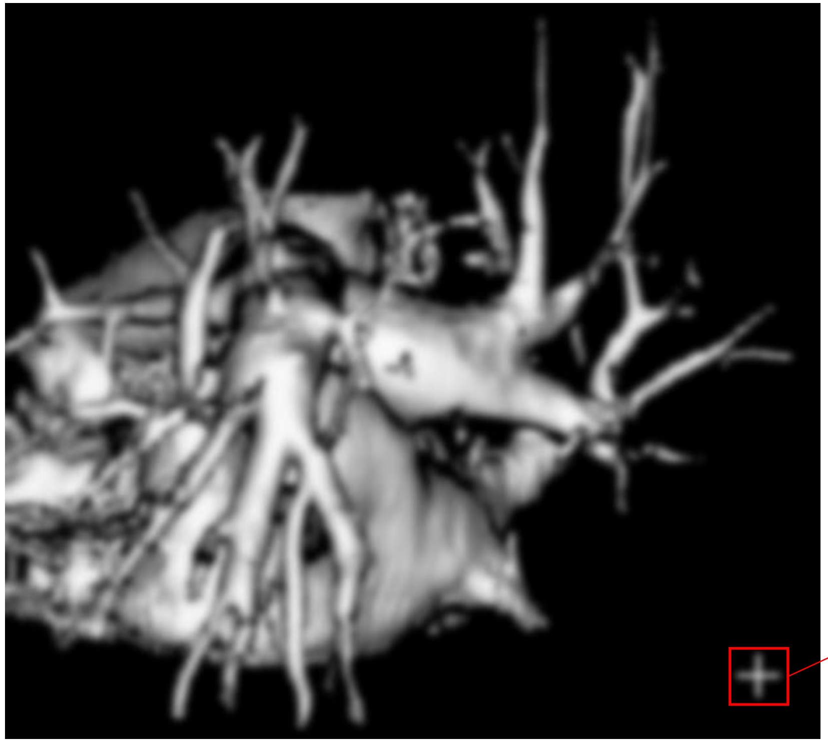


Figure. Blurred heart
(Original image courtesy of GE Medical Systems.)

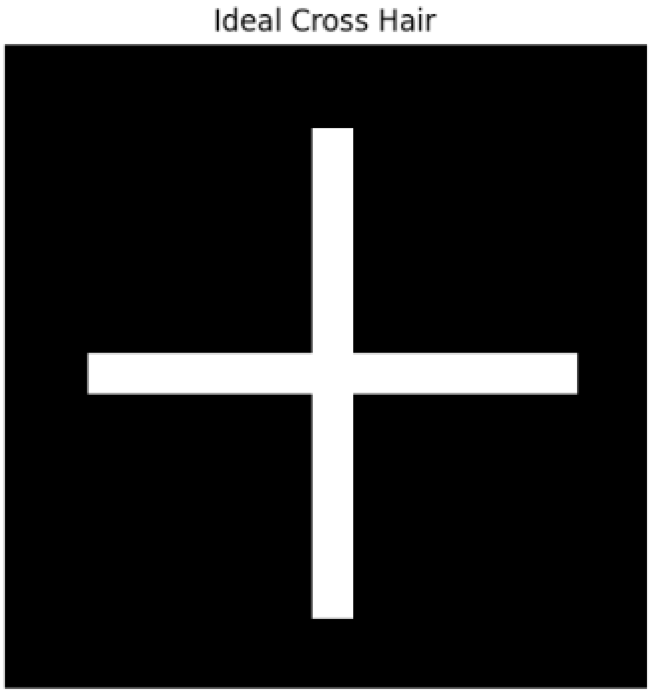


Figure. Crosshair before blur

Assumptions Problem

Upon solving the problem, we consider the following assumption

1. **No knowledge of the original heart image:** We will recover the degraded image as to verify the correctness of the degraded function, with not having the original heart image, thus at the restored result, we accept the result to our understanding.

Approach

We divide out approach into three main sections

Given some comments from observation and experimental

Calculating the estimated degraded function based on comments above

Re apply the degraded function to obtain the recovered image, and give conclusions about the correctness of the obtained function

Frequency Domain Encountered Approach

As far as we consider, blurring caused in the image is mostly affected in the frequency domain. With this comment, we approach to fully exploit the Fourier transform.

No Noise Corrupted Approach

We observe the histogram of a segment of the original image.

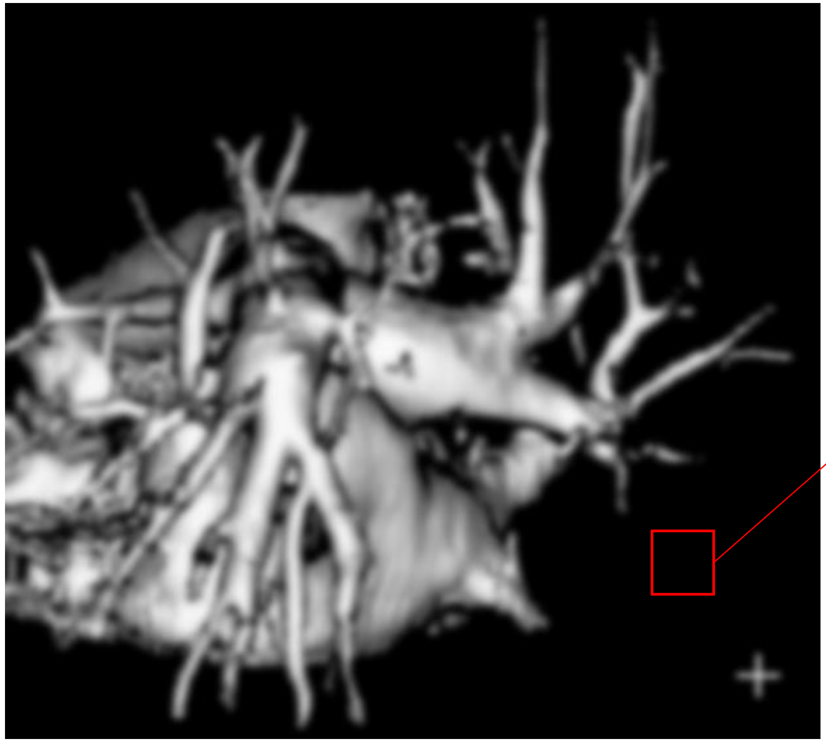


Figure. Input heart image.

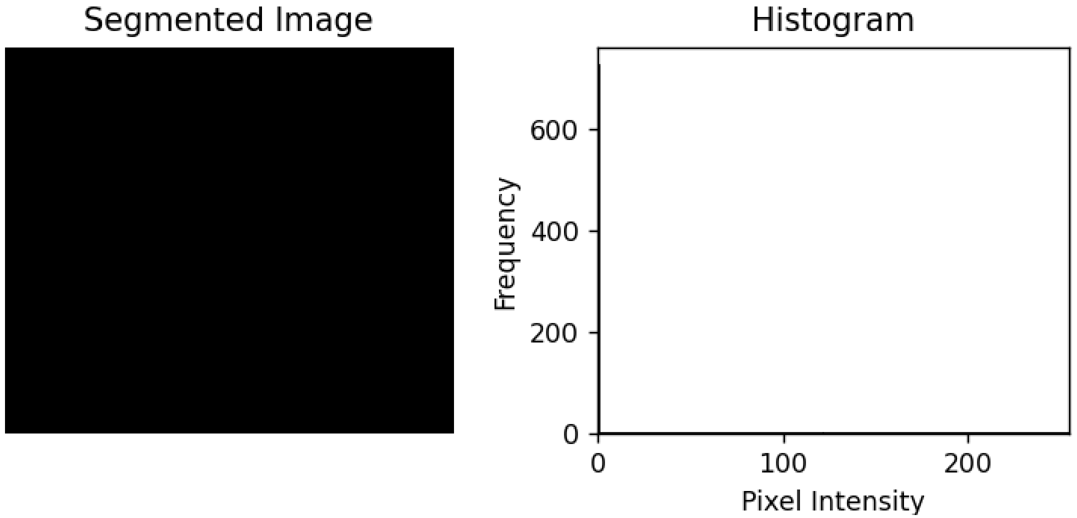


Figure. Histogram of segmented image.

Noise typically affects high-frequency components. Blurring the image attenuates these components, resulting in a smoother appearance. It can be seen in the histogram that, whether the original image have noise or not, after blurring, the appearance of noise is absence. In the absence of specific information about noise, it is reasonable to assume the image is not corrupted by noise.

Gaussian Filter Approach

We first denote some variables

- F_{blurred}: the Fourier transform of the blurred cropped crosshair image.
- F_{ideal}: the Fourier transform of the ideal crosshair image.
- H: the Fourier transform of the degraded function.

An approach to obtain H is by the following equation

$$F_{\text{blurred}} = F_{\text{ideal}} \cdot H \Leftrightarrow H = \frac{F_{\text{blurred}}}{F_{\text{ideal}}}$$

This technique is called the inverse filtering

Gaussian Filter Approach

In most cases, we cannot obtain directly H using the inverse filtering, since we miss knowledge of H itself. An alternative is to approximate H by the following

$$H \approx \frac{F_{\text{blurred}} \cdot \overline{F_{\text{ideal}}}}{\left|F_{\text{ideal}}\right|^2 + \varepsilon}$$

Where ε is the estimated constant. This is known as the Wiener filtering. Using this approximation, we can obtain the Fourier spectrum of the degraded function as below.

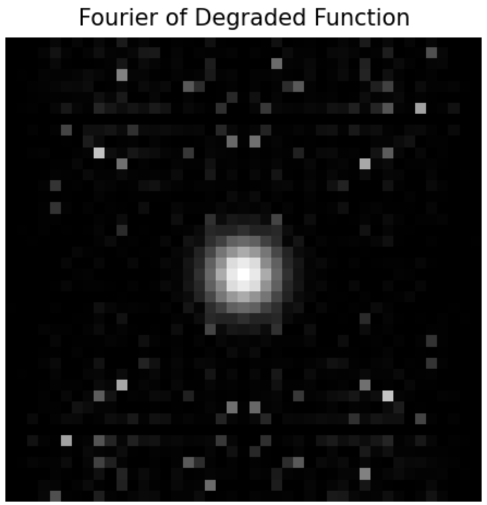


Figure. The Fourier spectrum of the degraded function.

Gaussian Filter Approach

We can ignore the noise dots, and focus on the white circle in the middle. This image resembles a lot of the Gaussian filter, which has the formula

$$G(u,v) = e^{\frac{-D(u,v)^2}{2D_0^2}}$$

Where D_0 is the cutoff frequency, $D(u,v)$ is the distance from each pixel to the center of the image, in this case the center of the Fourier Transformed image.

Here is the visualization plot of the Gaussian filter.

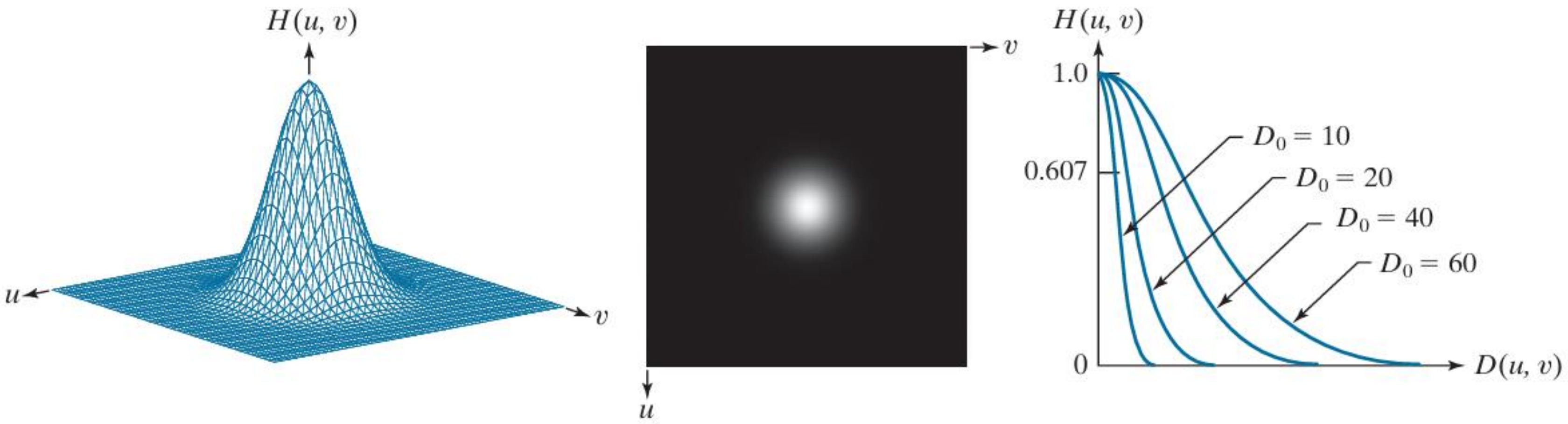


Figure. The Gaussian Lowpass Filter visualization and the cutoff frequency

Gaussian Filter Approach

Take a comparison of the Gaussian filter with cutoff frequency $D_0 = 2.407$ and the obtained degraded function.

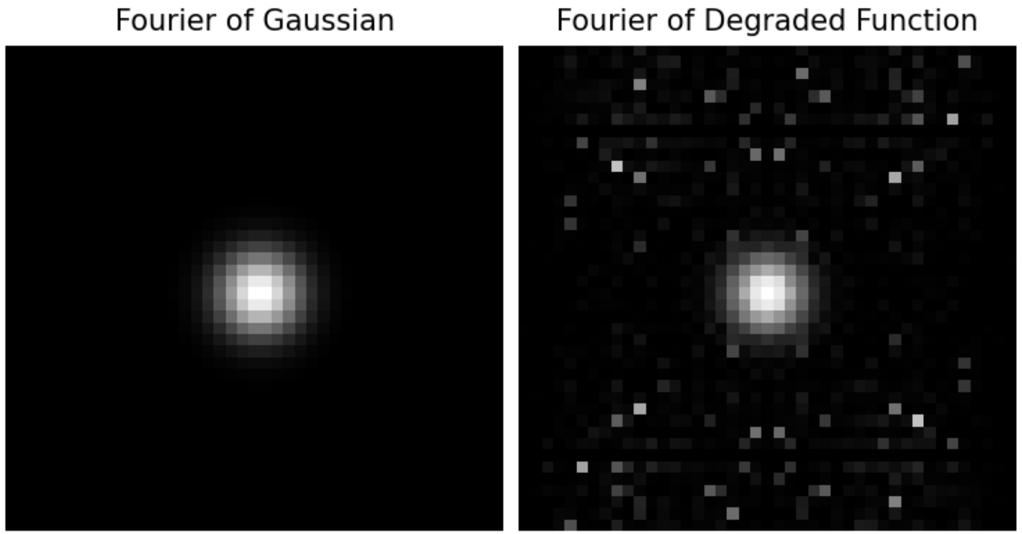


Figure. The Fourier spectrum of the Gaussian filter compared to the degraded function

This makes it reasonable for us to approximate our degraded function with the Gaussian filter. We will explain how we got the value 2.407 in the procedure below.

Comments Summary

We have a short summary of information before getting into the main procedure:

1. The image is not corrupted by noise.
2. The blurred is caused in the frequency domain.
3. The estimated degraded function is a Gaussian filter function.

This step is an important step, as it is the only information of the image that we have. We attempt by cropping out the blurred crosshair.

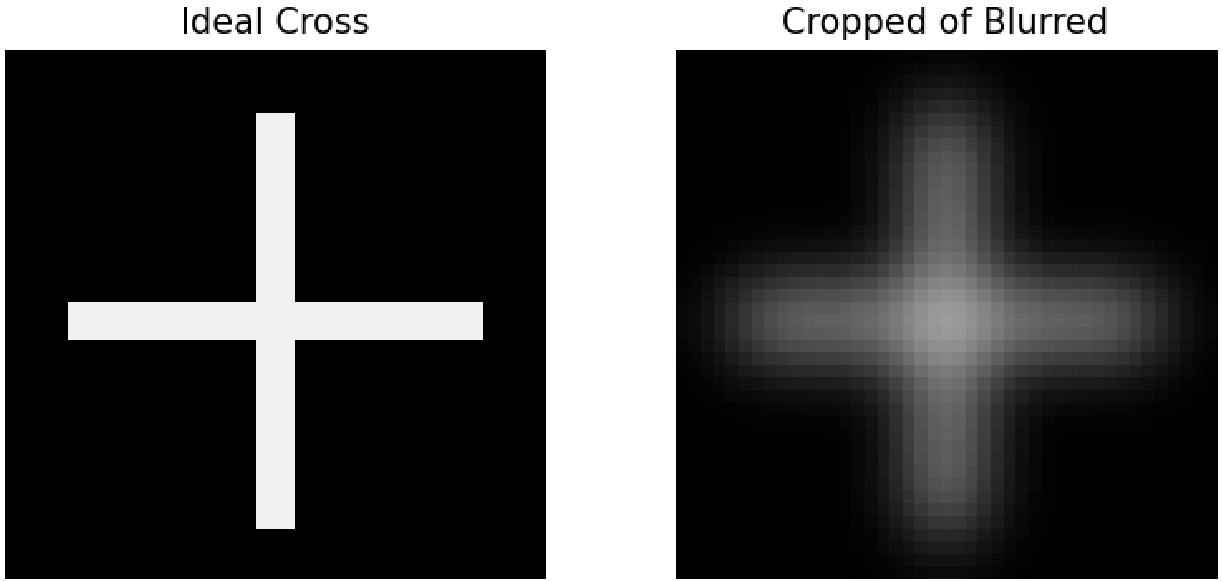


Figure. The cropped image compared to its original

Fourier Transformed Procedure

Since we are working on the frequency domain, it is crucial to apply Fourier transform before applying any filters.

Let f be any input image size M x N, we denote F as its Fourier transform, which is obtained by

$$F(u,v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) e^{-i2\pi(ux/M + vy/N)}$$

Fourier transform allows us to convert from the spatial domain, to the frequency domain. We do not discuss about why in this particular presentation.

Estimated Gaussian Filter Procedure

Previously we have stated that we can approximate the degraded function with the Gaussian filter function, which means, we want to have

$$H(u,v) \approx e^{\frac{-D(u,v)^2}{2D_0^{\,2}}}$$

This reduces the problem to find D_0 that fits best our assumption.

Recall that we have the following equivalent

$$H(u,v) \approx e^{\frac{-D(u,v)^2}{2D_0{}^2}} \Rightarrow D_0 \approx \sqrt{\left|\frac{-D(u,v)^2}{2\ln H(u,v)}\right|}$$

We take absolute value as to prevent negative values. To obtain the estimated cutoff frequency, the simplest approach is to take the average

$$\overline{D_0} \approx \frac{1}{M \cdot N} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} \sqrt{\left|\frac{-D(u,v)^2}{2\ln H(u,v)}\right|}$$

Estimated Gaussian Filter — Procedure

Some problem arises when using the approximation mentioned above

1. $H(u, v)$ is negative, makes it impossible to take natural logarithm.
2. $-D^2(u, v) / H(u, v)$ is negative, and its magnitude is large, taking the absolute value wrongly calculate D_0.

We provide a procedure to treat in the general case

1. Let $D_0(u, v)$ be the estimated cutoff frequency at position (u, v).
2. Value H(u, v) is first calculated as
   a. If $H(u, v) \leq 0$, let $H(u, v) = 1.0001$.
   b. Otherwise, do not change $H(u, v)$.
3. We then calculate $t = -D^2(u, v) / \ln(H(u, v))$
   a. If $t \leq 0$, let $t = 0$.
   b. Otherwise, do not change t.
4. Calculate $D_0(u, v) = \sqrt{t}$.

Estimated Gaussian Filter Procedure

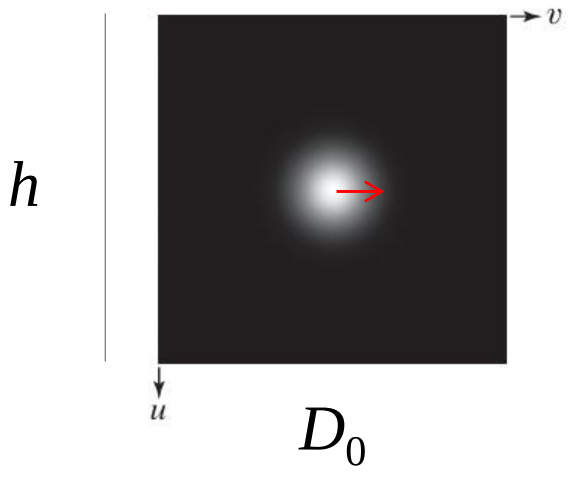
From here, we can estimate the cutoff frequency D_0 as

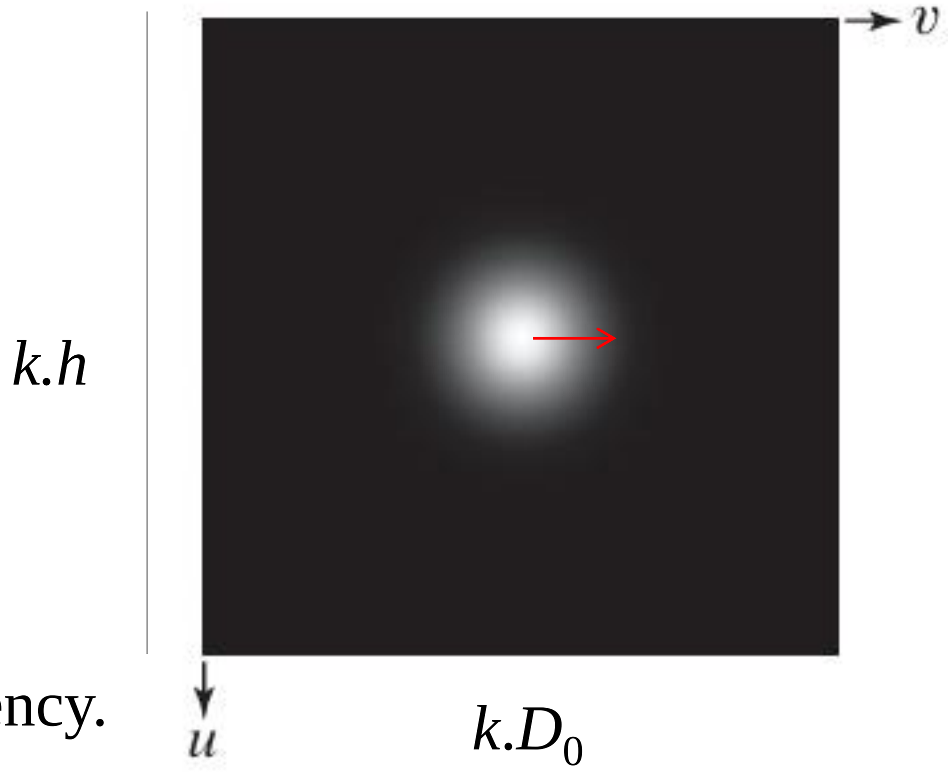
$$D_0 \approx \frac{1}{M \cdot N} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} D_0(u, v)$$

This is how we can obtain $D_0 = 2.407$ mentioned earlier.

Scale for Input Image

We take advantage of the cutoff frequency mentioned earlier, for a larger scale image, we would like to scale the cutoff frequency approximately as the ratio between the large and cropped image.





This encourage us to just scale the cutoff frequency.

Thus, the estimated *H'* Gaussian filter for the input heart image will have the following formula

$$H(u,v) \approx e^{\frac{-D(u,v)^2}{2D_0{}^2}} \xrightarrow{\text{Scaling}} H'(u,v) \approx e^{\frac{-D(u,v)^2}{2(kD_0)^2}}$$

Restored Image Procedure

This phase is not necessary for the problem itself, but it is a way to verify whether the obtained function is acceptable or not.

Restored Image Procedure

Having the estimated *H'*, obtained from scaling the cutoff frequency D_0 from *H*, we can obtain the restored image, using the Wiener filtering

$$F_{\text{restored}} = \frac{F_{\text{input}} \cdot \bar{H}'}{|H'|^2 + K}$$

F_{input}, F_{restored} respectively be the Fourier transform of the input and the restored image, *K* will be a constant being estimated.

Lastly, we use the inverse Fourier transform to bring the image back from the frequency domain to its spatial domain.

$$f_{\text{restored}}(x, y) = \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{n=0}^{N-1} F_{\text{restored}}(u, v) e^{i2\pi(ux/M + vy/N)}$$

Post Processing Procedure

We introduce two extra steps in enhancing the image quality: brightening and sharpening.

For brightening, we use thresholding, a technique which selects pixels at a certain level of intensity and add to its intensity.

$$f_{\text{brighten}} = \begin{cases} f_{\text{restored}} + \alpha & \text{if} \quad f_{\text{restored}} \geq t \\ f_{\text{restored}} & \text{otherwise} \end{cases}$$

Where *a* is the added intensity, *t* is the threshold level.

Post Processing Procedure

For sharpening, we opt to use the Laplacian sharpening kernel, and apply convolution to the restored image

$$L = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

Apply convolution to obtain the final sharpened image

$$f_{\text{sharpened}} = f_{\text{restored}} \star L$$

Here * denotes convolution.

Summary Procedure

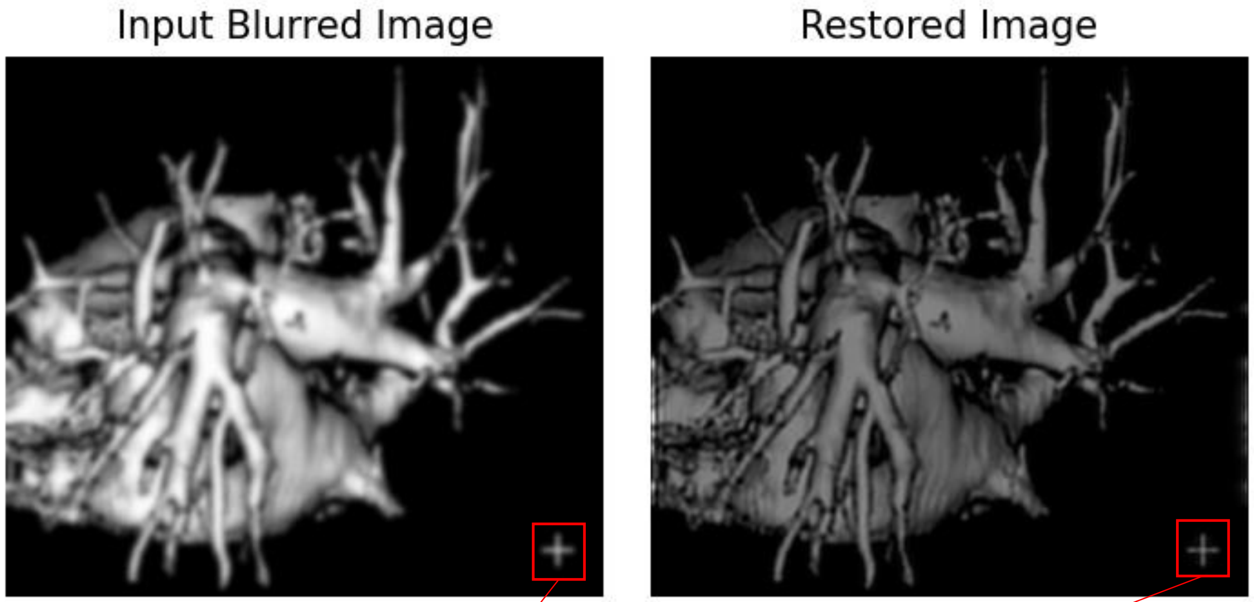
We have a short summary of the main procedure:

1. Crop out the blurred crosshair, as well as creating the ideal crosshair image.
2. Get the Fourier transform of images involved.
3. Use averaging to estimate the degraded function by Gaussian filter function.
4. Obtain the scaled degraded function for the heart image.
5. Use Wiener filtering to obtain the recover image.
6. (Optional). Brightening and sharpening the recovered image for visualization.

By explanations above, combine with some experiment, we show below the parameters for the problem.

Parameter	Meaning	Value
D_0	Cutoff frequency for the blurred crosshair	2.407
ε	The estimate constant for finding the degraded function	0.0001
K	The estimate constant for Wiener filter	0.005
k	The scaling coefficient	13.404
a	The added intensity	30.000
t	The threshold level	5.000

Crosshair Comparison



The restored image is said to be better than the blurred image, here are the reasons:

Crosshair restored: Observing the crosshair at the corner, it is seen that the recovered crosshair resembles more of the ideal crosshair (although darker, this is a result of the Wiener filter).

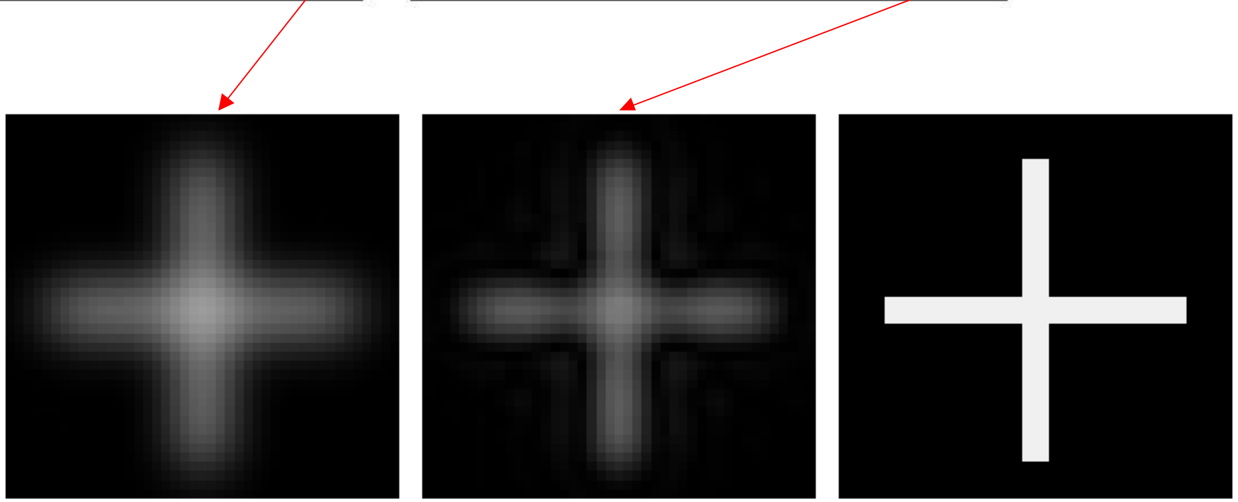
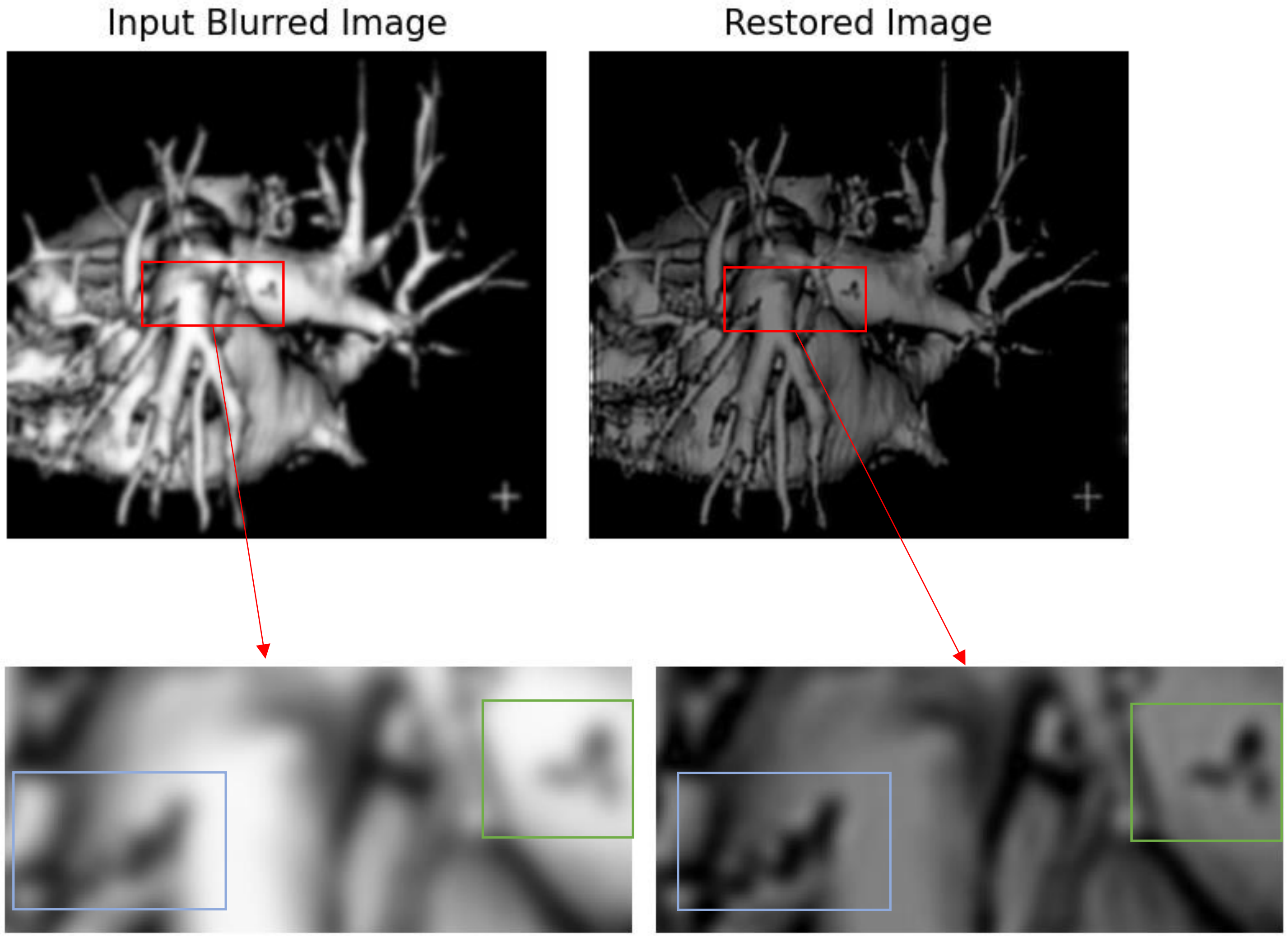


Figure. The blurred, restored and ideal crosshair image

Crack Comparison

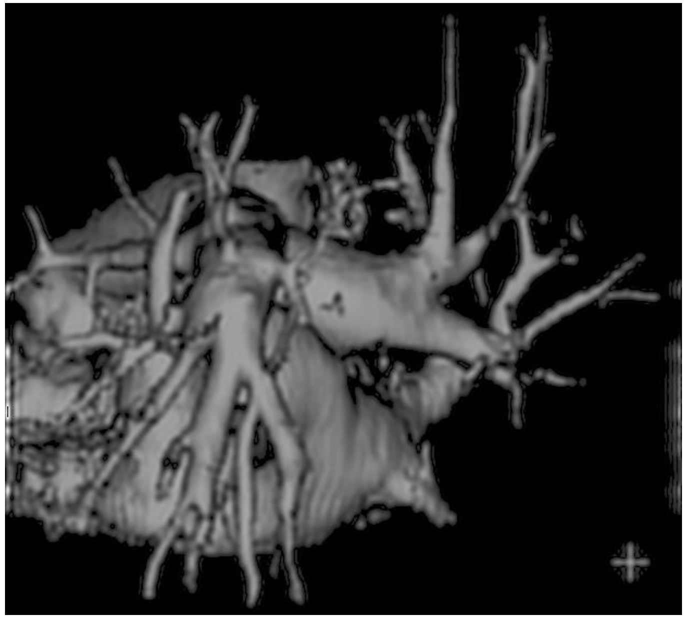


Information gained: take a look at the areas which appears to be a crack in the heart. It is seen in the blurred image, the shape of the triangle crack (in the green rectangle) is not clear, whether in the restored image, we can clearly see the shape of the crack, which resembles a triangle.

Another similar area lies in the blue rectangles. In the blurred image, we barely see the shape of the crack. But in the restored image, we can see details of it.

Image Results

For visualization purpose, we provide two resulted image using brightening and sharpening.



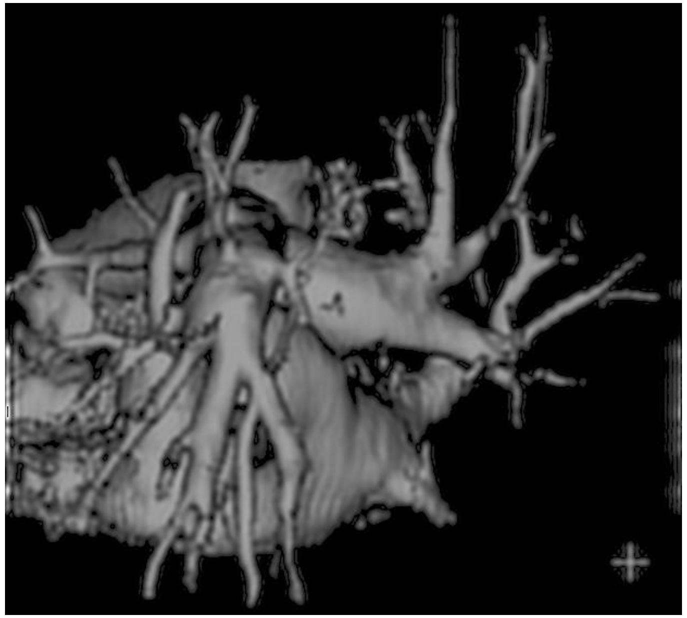


Figure. Brightened and sharpened image

Since these two images are optional steps to enhance the image quality, we only care about the difference between them and the restored image, other conclusions are similar to ones made for the restored image and the blur image.

Crack Comparison

We provide a side-by-side comparison image of the input image and three results, and observe its improvement.

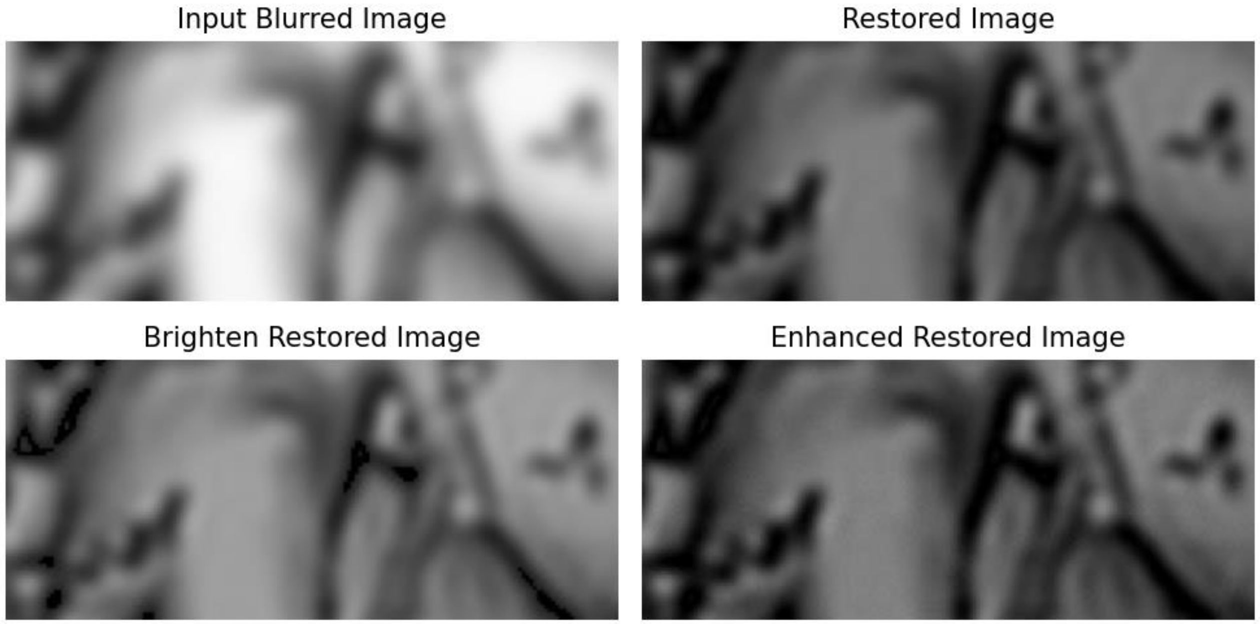


Figure. Crack areas in four images

With the brightened image, the cracks are shown more easier. We notice in the sharpened image, the details shows a more concise shape of how it suppose to be.

Image Results

We have a short summary of the image results:

1. The restored image have successfully restored the crosshair image at its best.
2. The restored heart image gives more information than the original blurred image. Shapes are more accurate.
3. The brightened image gives a brighter version of the restored image, while the sharpened image gives more sharp details, making the image more easily for visualization.

Conclusion

Throughout solving the problem, we could see that

- The estimated Gaussian function is good enough to predict the degraded function of the input heart image.
- Different cutoff radius is applied for different image size.
- Enhancement of an image does not give more information as it already shows. Such work is done for better visualization.

Limitations

1. **Limited filters:** Gaussian and Butterworth filters are good filters to estimate the original image in our scenario, there might be other filters which may give better results.

Future Work

1. **More filters:** explore other filters rather than Gaussian or Butterworth.
2. **Post processing methods:** as to enhance the image quality, better post processing methods can be encountered.

Reference

1. Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing, Fourth Edition*. Pearson, 2018.