Medical Image Restoration Techniques

Dom Carrillo[1] and Joshua Dublin[2]

[1,2]Department of Computer Science, Southern Connecticut State University, New Haven, CT, USA

***Abstract*—Image restoration aims to reconstruct or recover an image affected by noise, blur, or low resolution. This process is essential in medical imaging, where high quality images are essential for proper diagnosis of injury or illness. Being able to fix imperfect images with medical image restoration can save money or lives. The Inverse Filter, Wiener Filter, and Richardson-Lucy algorithm are common methods used in medical image restoration. This paper reviews the methodology of these three algorithms and presents an experiment of these algorithms with PSNR and SSIM results. This study will provide readers with a strong understanding of these algorithms.**

I. Introduction

CT scans, MRIs, X-Rays, microscopic images, and ultrasound images are crucial to diagnosis of injury or illness. Unfortunately, these images may suffer from additive noise and blurring during the acquisition process. With medical image restoration, blurred or imperfect images can be made usable, potentially saving money or lives [1].

There are many factors that affect image quality, these may be controllable, but mistakes can be made to reduce the quality of an image. These main factors include contrast, blur, noise, artifacts, and distortion [2]. In our experiment, different types of blurs and additive noises are added to an original image before the restoration filters are applied.

The Inverse Filter, Wiener Filter, and Richardson Lucy algorithm are three of the most common methods in image restoration [3]. A direct comparison of the effectiveness of these algorithms can be useful in finding out which is the best one in restoring medical images. Our experiments and results will provide a guideline for the basic ideas of image restoration, studying the Inverse Filter, Wiener Filter, and Richardson-Lucy Algorithm. We determine which of these three algorithms is the best comparing their PSNR (Peak Signal to Noise Ratio) and SSIM (Structural Similarity Index Measure) values.

Currently, it is known that out of the three algorithms, the Richardson-Lucy algorithm performs the best in restoring images [3]. The Inverse filter has the weakest performance [3]. The Wiener filter has stronger results than the Inverse filter [3]. Using the Wiener filter on images with low additive noise allows the image to look clearer. A benefit of the Richardson-Lucy algorithm is that it is iterative. This is beneficial because with each iteration, the Richardson-Lucy algorithm refines the restored image. The results get better with every iteration [3].

This problem is important to solve as it will clear up consensus as to which is the best algorithm. There may be a future researcher who is interested in medical image restoration. By reading our paper, they will gain important knowledge on all three algorithms. Through our experiment and results, they will learn about which algorithm is best of the three. By reading our paper, a researcher may find clarity on the topic or find a new direction. This paper will be useful for anyone who wants to learn about these three algorithms quickly, and then continue in their own direction of research.

In this article, there is a section on background of the problem and related work. The next section describes the methodology of the problem. The Degradation Model, Degradation Functions, Gaussian Blurring, Additive Noise Functions (Gaussian Noise, Poisson Noise) are discussed. The methodology of the restoration methods (Inverse Filter, Wiener Filter, Richardson Lucy Algorithm) are then discussed. The Evaluation Metrics (PSNR and SSIM) are then discussed. In the following sections, the experimental setup, results, conclusions and future work are discussed, with images and graphs provided. Finally, references are listed.

II. Background and Related Work

Advancements in technology have allowed for precise medical images. However, imperfect images are there are frequently taken. If a blurry, noisy, or imperfect image is taken, one may want to discard it. However, with advancements in medical image restoration, these imperfect images do not necessarily need to be thrown out. Using restoration techniques, these images can be restored, which can save lives, time, and money.

One way to restore images is through all-in-one restoration via task-adaptive routing. While single-task medical image restoration has had success, it has limited generalizability, which stands as an obstacle to wider application. All-in-one medical imaging addresses these concerns. However, a universal model often encounters task interference issues. A solution to this is task-adaptive routing strategy, which lessens task interference. This has shown strong performance in three tasks: MRI super-resolution, CT denoising, and PET synthesis [4].

Another method of medical image restoration is energy-efficient high-fidelity image reconstruction with memristor arrays for medical diagnosis. Image reconstruction algorithms have challenges in processing large amounts of data with high speed and quality. Memristrive image reconstructor (MIR) can greatly accelerate image reconstruction with the use of memristor arrays. High-fidelity MRI and CT image reconstructions achieve excellent results with MIR, a promising high fidelity image reconstruction platform for future medical diagnoses [5] .

Another method is medical image reconstruction through deep learning. Handcrafted models are mathematical models designed by human knowledge. After these were established, data driven modeling emerged, which relies on a model of both human designs and observed data. Recently, deep learning models have become prevalent where models are mostly based on learning with minimal human designs. Deep learning models can be used to deliver reconstructed medical images with minimal cost and risk to patients. Typical handcrafted models are well interpretable with solid theoretical supports on the robustness, recoverability, and complexity. Deep learning models are more flexible and effective in extracting information from large data sets than handcrafted models. A combination of handcrafted modeling with deep learning modeling has great potential. Medical image restoration will benefit from deep learning [6].

Our work differs from these research objectives. We are interested in determining whether the Inverse Filter, Wiener Filter, or Richardson-Lucy Algorithm is most effective in restoring images. These three algorithms are some of the most popular for medical image restoration. Through our experiment, we will find insight on these popular algorithms. We hope to come up with findings that will help researchers of medical image restoration. Our work will build upon the work that is already done in this field.

III. The Problem: Image Degradation

Figure 1 shows the degradation model that is used in this research. The degraded image g(x,y) in the spatial domain is represented in equation 1. This equation represents the spatial domain representations, where g(x,y) is the degraded image, f(x,y) is the original image, h(x,y) represents the blurring function, and n(x,y) is the additive noise. Convolution in the spatial domain is represented as (*) in the equation. This is similar to multiplication in the frequency domain, which is represented in equation 2. Here, the functions are converted into Fourier Transforms with spatial frequency coordinates (u,v), with F being the Fourier transform of the original image, H as the Fourier Transform of the degradation function, N as the Fourier Transform of the additive noise function and G as the Fourier transform of the original image [7]. The degradation functions and the additive noise functions used in the experiment are further discussed in the next section.

$$g(x,y) = f(x,y) * h(x,y) + n(x,y) \tag{1}$$

$$G(u,v) = F(u,v)H(u,v) + N(u,v) \tag{2}$$

A. Motion Blurring

Motion blurring adds an effect to the image to appear as if the objects are moving, adding a blur in a specific direction. The motion is controlled by angle θ and/or distance L. The format fo the motion blurring function can be represented in equation 3 below [8].

$$h(x) = \begin{cases} \frac{1}{L} & \text{if } \sqrt{x^2+y^2} \le \frac{1}{L} \text{ and } \frac{x}{y} = -\tan(\theta) \\ 0 & \text{Otherwise} \end{cases} \tag{3}$$

B. Gaussian Blurring

Gaussian Blur uses a Gaussian Function to determine the transformation to each pixel in a two-dimensional image. σ represents the standard deviation of the Gaussian Distribution, as shown in equation 5. X represents the distance from the source in the horizontal axis, and y is the distance from the source in the vertical axis. The formula produces a surface with concentric circles with a Gaussian Distribution from the center point [3].

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{4}$$

C. Gaussian Noise

Gaussian noise is an additive noise that uses a Gaussian distribution property and a probability distribution function, which is shown in equation 5 for two dimensional images. σ represents the standard deviation of the noise signal [8]. Gaussian noise can be caused by poor illumination, high temperature, and transmission. It can be minimized by using filtering techniques for noise removal [9].

$$\psi(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{5}$$

D. Poisson Noise

Poisson noise is an additive noise seen in images and is also known as quantum noise. It is a signal dependent noise, and is more challenging to filter than Gaussian noise. Equation 6 represents the degraded image with added Poisson noise, where f(i,j) is the original image, λ is the intensity, and g(i,j) is the degraded image. The Poisson function takes two different inputs, one is the mean of the Poisson distribution, and the other is the output is a Poisson random generation function to produce Poisson random numbers [3].

$$g(i,j) = \frac{1}{\lambda}\text{Poisson}(\lambda f(i,j)) \tag{6}$$

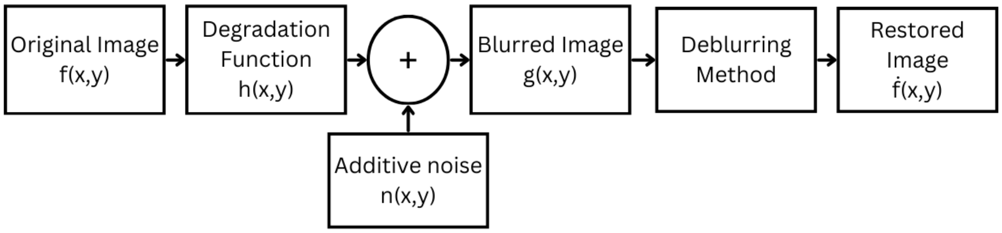


Fig. 1: Degradation Model

IV. Restoration Methods

A. Inverse Filter

One of the simplest ways to restore images is the Inverse filter. Inverse filtering is a deterministic and direct method for image restoration. The images involved must be lexicographically ordered, which means that an image is converted to a column vector by pasting the rows one by one after converting them to columns [10].

The Fourier transform of a restored image can be taken by dividing it by the degradation function, as shown in equation 7.

$$\dot{F}(u,v) = \frac{G(u,v)}{H(u,v)} \tag{7}$$

$$G(u,v) = F(u,v)H(u,v) + N(u,v) \tag{8}$$

In equations 7 and 8, Ḟ(u,v) represents the estimated image. F(u,v) represents the original image. N(u,v) represents the additive noise. H(u,v) represents the degradation function. G(u,v) represents the degraded image. Equation 7 shows the technique of direct inverse filtering. The estimated image is taken by dividing the degraded image by the degradation function.

Substituting G(u,v) from equation 8 into the equation of Ḟ(u,v) results in equation 9.

$$\dot{F}(u,v) = \frac{F(u,v)H(u,v) + N(u,v)}{H(u,v)} \tag{9}$$

A simpler way of expressing the function Ḟ(u,v) is achieved by dividing both the numerator and denominator by H(u,v) and simplifying each term in the equation. It is expressed in equation 10.

$$\dot{F}(u,v) = F(u,v) + \frac{N(u,v)}{H(u,v)} \tag{10}$$

In these last equations, N(u,v) is unknown. It can be estimated, but the exact value is never known. A small value of H(u,v) allows N(u,v) / H(u,v) to dominate over F(u,v), further distorting the image. N(u,v) / H(u,v) will be much larger than F(u,v) if H(u,v) is very small.

B. Wiener Filter

Norbert Wiener proposed the concept of a Wiener Filter in 1940. The Wiener Filter is helpful when partial knowledge is applicable about a blurring function. The Wiener Filter incorporates the degradation function and statistical characteristics of noise into the restoration process [11].

The goal of the Wiener function is to minimize the mean-squared error between the original and estimated image, as shown in equation 11 where F(u,v) represents the original image and Ḟ(u,v) represents the estimated image.

The mean squared error is computed by averaging the squared intensity differences of distorted and reference image pixels, along with the related quantity of peak signal-to-noise ratio (PSNR) [11].

Given two pictures A and B, sum the square of the difference between every pixel in A and the corresponding pixel in B and divide it by the number of pixels.

The Wiener Filter removes additive noise and inverts the blurring. Wiener filter is also known as the minimum mean square error or least square error filter. The Wiener Filter is optimal in minimizing the mean squared error between the original and restored image.

$$e^2 = E\left[(F(u,v) - \dot{F}(u,v))^2\right] \tag{11}$$

Equation 12 shows the function for the estimated image with the Wiener Filter function where Ḟ(u,v) is the estimated image, W(u,v) is the Wiener Filter function, G(u,v) is the degradation function.

Equation 13 shows the Wiener Filter function where $P_s(u,v)$ is the power spectrum of the original image obtained by the Fourier transform, $P_n(u,v)$ is the power spectrum of the additive noise obtained by the Fourier transform, H(u,v) is the degraded image, and $H^*(u,v)$ is the complex conjugate of H(u,v).

$$\dot{F}(u,v) = W(u,v)G(u,v) \tag{12}$$

$$W(u,v) = \frac{H^*(u,v)P_s(u,v)}{|H(u,v)|^2 P_s(u,v) + P_n(u,v)} \tag{13}$$

$$W(u,v) = \frac{H^*(u,v)}{|H(u,v)|^2 P_s(u,v) + \frac{P_n(u,v)}{P_s(u,v)}} \tag{14}$$

$$W(u,v) = \frac{P_s(u,v)}{P_s(u,v) + \sigma_n} \tag{15}$$

Equation 12 is a function for the minimized error. Given a degraded image, take the discrete Fourier transform to obtain G(u,v). The original image is estimated by taking the product of G(u,v) with the Wiener filter W(u,v). The inverse discrete Fourier transform can be used to take the image estimate from its spectrum.

Equation 14 shows the Wiener Filter function after dividing the numerator and denominator by $P_s(u,v)$. If there is a strong signal to noise ratio, the term $P_n(u,v)$ / $P_s(u,v)$ in equation 14 becomes very small, allowing the Wiener filter to simply become an inverse filter, $H^-1(u,v)$ for the point spread function. This mathematical connection shows that the Inverse and Wiener filters are related.

The point spread function is the three-dimensional diffraction pattern of light emitted from an infinitely small point source in the specimen and transmitted to the image plane through a high numerical aperture objective. It is considered to be the fundamental unit of an image in theoretical models of image formation. When light is emitted from such a point object, a fraction of it is collected by the objective and focused at a corresponding point in the image plane [12].

The point spread function represents how an imaging system responds to a single point of light, which describes the amount of blur introduced by the system. This is crucial for restoring a blurred image through a process called deconvolution. By knowing the point spread function, the blurring effect can be mathematically reversed to recover original image details.

The point spread function is a valid measure for the quality of an optical system as it reveals how points are blurred in the image. Because the point spread function is always normalized, it is easy to compare point spread functions of different systems and thereby compare their imaging qualities [13].

If there is additive white noise with no blurring signal, the Wiener Filter becomes equation 15, where $\sigma(n)$ represents the noise variance.

C. Richardson-Lucy Algorithm

The Richardson-Lucy algorithm is an iterative method to recover an image smoothed by a known point-spread function. It is proven to be robust in the presence of noise and blur. The algorithm was originally developed to solve positron tomography imaging problems. The Richardson-Lucy algorithm can be used effectively when the point spread function is known, but little or no information is available for the noise. The blurred and noisy image is restored by the iterative Richardson-Lucy algorithm [14].

The Bayes Theorem is shown in equation 16 below where P(y|x) is the conditional probability of an event y, given event x; P(x) is the probability of an event x; and P(x|y) is the conditional probability of an event x, given event y.

$$P(x|y) = \frac{P(y|x)P(x)}{\int P(y|x)P(x)\,dx} \tag{16}$$

The inverse relation permits the derivation of the iterative algorithm shown in equation 17, where i is the iteration number, g(y,x) is the known point spread function, c(y) is the degraded image or convolution, and $f_i(x)$ is the image at the ith iteration. In terms of convolutions, the equation can also be written as equation 18. Detail is added to the image in subsequent iterations because of the algorithm's form. Energy is conserved at every iteration.

$$f_{i+1}(x) = \frac{\int g(y,x)c(y)\,dy}{\int g(y,z)f_i(z)\,dz} f_i(x) \tag{17}$$

$$f_{i+1}(x) = \left\{ \left[\frac{c(x)}{f_i(x) \otimes g(x)} \right] \otimes g(-x) \right\} f_i(x) \tag{18}$$

In Equation 18, the cross is the convolution operation. The point spread function g(x) is known, so one finds the object f(x) by iterating. The inverse of equation 18 is shown as equation 19. This equation allows for the calculation of the point spread function of an object.

$$g^k_{i+1}(x) = \left\{ \left[\frac{c(x)}{g^k_i(x) \otimes f^{k-1}(-x)} \right] \otimes f^{k-1}(-x) \right\} g^k_i(x) \tag{19}$$

V. Evaluation Metrics

A. PSNR

The Peak Signal to Noise Ratio (PSNR) [15] is a ratio between the highest possible power of a signal, or an image, and the power of the noise that affects its representation. PSNR is usually expressed in decibel scale (dB). It is a rough estimation to human perception of reconstruction quality. A higher PSNR generally indicates that the reconstruction has a higher quality, but in some cases, such as edge detection PSNR, a lower PSNR indicates higher quality. The calculation of PSNR is shown in equation 20. R is the maximal variation in the input image data. If R is an 8-bit unsigned integer, it is 255. The calculation of MSE is shown in equation 21. Where I_1 is the is original image and I_2 is the estimated image. This value should be less when using image restoration. This value could be made higher to ensure it finds more edge points on the image [16].

$$PSNR = 10 \log_{10}(\frac{R^2}{MSE}) \tag{20}$$

$$MSE = \frac{\sum_{M,N}(I_1(m,n) - I_2(m,n))^2}{M,N} \tag{21}$$

B. SSIM

The Structural Similarity Index (SSIM) [17] calculation is shown in Equation 22 where $\vec{x}$ and $\vec{y}$ represent vectors with pixels from the true image x and the estimated image y. The SSIM operates in stages given by luminance, contrast, and structure. Luminance is calculated as shown in Equation 23. It is estimated using its mean intensity, where N is the number of pixels in $\vec{x}$. The luminance comparison function $l(\vec{x}, \vec{y})$ is used to compare the luminance of the true and estimated images, μ_x and μ_y. The standard deviations of the image intensities are also calculated, as shown in Equation 24. The contrasts comparison function $c(\vec{x}, \vec{y})$ is used to contrast signal of true image $\vec{x}$ and estimated image $\vec{y}$. These contrasts are normalized by dividing them by their standard deviations. The structure comparisons are performed on these signals given by $\frac{\vec{x}-\mu x}{\sigma_x}$ and $\frac{\vec{y}-\mu y}{\sigma_y}$ to obtain index $s(\vec{x}, \vec{y})$. These three components are used in the SSIM equation shown in equation 22 [18].

$$SSIM(\vec{x}, \vec{y}) = f(l(\vec{x}, \vec{y}), c(\vec{x}, \vec{y}), s(\vec{x}, \vec{y})) \qquad (22)$$

$$\mu_x = \frac{1}{N} \sum_{i=1}^{N} x_i \qquad (23)$$

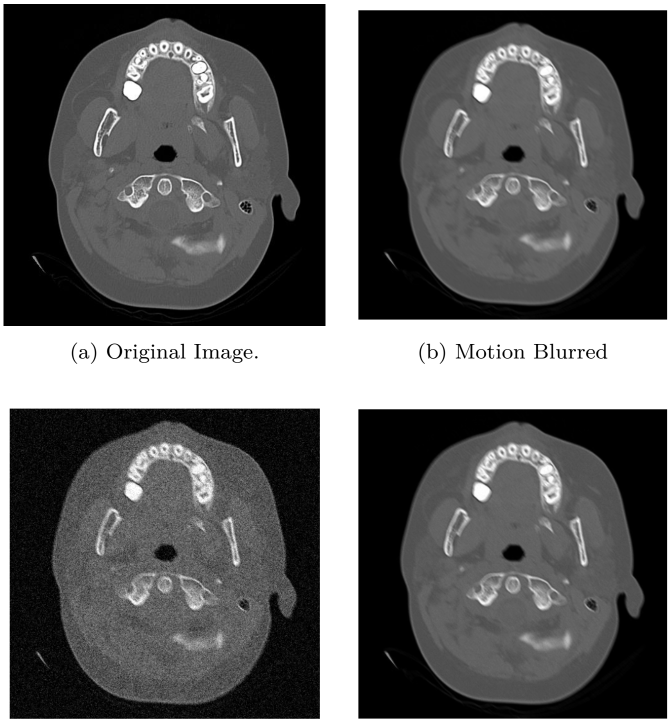
$$\sigma_x = (\frac{1}{N-1} \sum_{i=1}^{N} (x_i - \mu_x))^{1/2} \qquad (24)$$

VI. Experimental Setup

We developed a Python program that takes an image and applies the blurring functions, which includes Gaussian Blurring and Motion Blurring. At each blurred image, Gaussian Noise and Poisson Noise are added. After blurring the image, Inverse Filter, Wiener Filter, and the Richardson-Lucy Algorithm are used to deblur the image. PSNR and SSIM values are taken from each unblurred image and are used to formulate graphs to visualize the results. The steps used in the experimentation are outlined below.

1) The original image is taken
2) The blur is added (Gaussian Blur and Motion Blur)
3) Different additive noises are added to each blurred image (Gaussian Noise and Poisson Noise)
4) There exists four total images to unblur (Gaussian Blur + Gaussian Noise, Gaussian Blur + Poisson Noise, Motion Blur + Gaussian Noise, and Motion Blur + Poisson Noise)
5) The three restoration methods used in this experiment get applied to each of the four blurred and noisy images
6) PSNR and SSIM values are taken from each unblurred image.

The image used in this experiment is an image of a brain tumor taken from a CT scan out of a dataset with medical images. Figure 2 shows the original image and the motion-blurred images with different additive noises and figure 3 shows the Gaussian blurred images with different additive noises. Figure 6 shows the resulting deblurred images using the different algorithms that are used in the experiment.



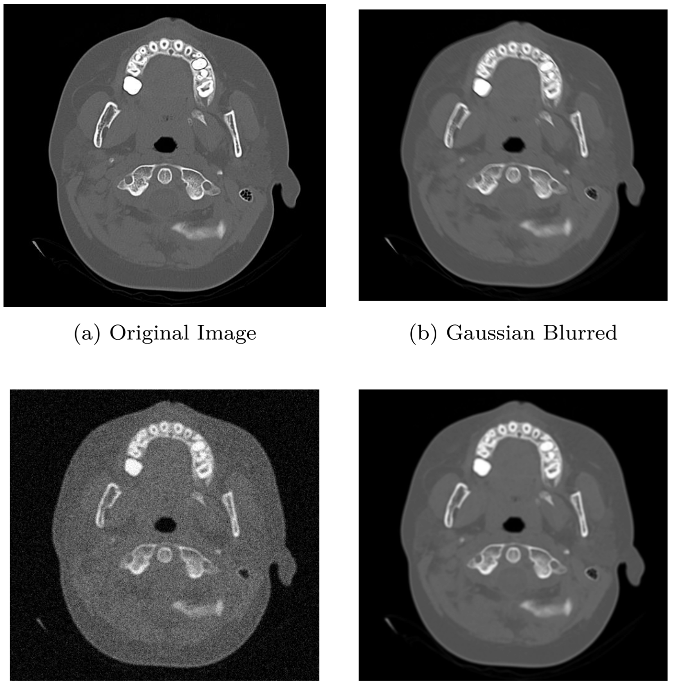
(a) Original Image.

(b) Motion Blurred

(c) Motion Blurred Image with Gaussian Noise.

(d) Motion Blurred Image with Poisson Noise.

Fig. 2: Motion Blurred Images.



(a) Original Image

(b) Gaussian Blurred

(c) Gaussian Blurred Image with Gaussian Noise

(d) Gaussian Blurred Image with Poisson Noise

Fig. 3: Gaussian Blurred Images.

VII. Experimental Results

The different filters were applied to the the blurred images with additive noise. The restored images using

Direct Inverse Filter are shown in Figure 6. The PSNR values of each restored image are shown in Table I. The SSIM values of each restored image are shown in Table II.

TABLE I: PSNR Values for Different Filtering Techniques

	Motion + Gaussian Noise	Motion + Poisson Noise	Gaussian + Gaussian Noise	Gaussian + Poisson Noise
Inverse	10.5024	10.4136	8.3143	10.5493
Wiener	10.38124	10.38124	10.38123	10.38124
RL Algorithm	10.414	10.414	10.414	10.414

TABLE II: SSIM Values for Different Filtering Techniques

	Motion + Gaussian Noise	Motion + Poisson Noise	Gaussian + Gaussian Noise	Gaussian + Poisson Noise
Inverse	0.30247	0.30247	0.01423	0.01423
Wiener	0.36136	0.36136	0.36136	0.36136
RL Algorithm	0.36461	0.36461	0.36459	0.36459

The mean PSNR and SSIM values are calculated and shown on table III. The data shown in each column of this table is visualized by graphing as shown on Figure 4 and Figure 5.

TABLE III: Mean PSNR and SSIM Values

	PSNR	SSIM
Inverse	9.9449	0.1584
Wiener	10.3812	0.3614
RL Algorithm	10.4139	0.3646

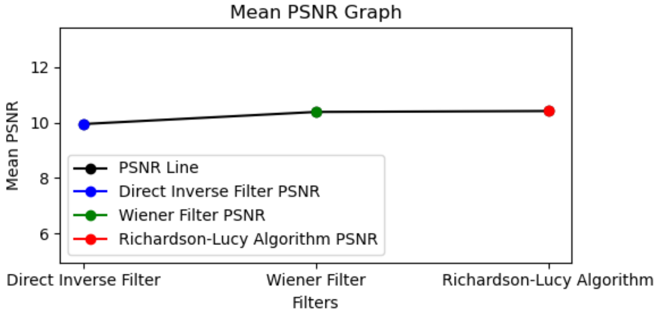


Fig. 4: Mean PSNR Graph

These graphs show that the Richardson-Lucy Algorithm has the best results in restoring images. It has slightly better results than the Wiener Filter, which also restores images very well, as shown in in the graphs and tables. The Inverse Filter does not restore images well, as noise gets amplified in the restoration of these images, further degrading the blurred images. The Inverse Filter is less suitable for restoring medical images.

Based on the PSNR values and the images, it can be deduced that PSNR values do not perceive human perception. Some of the restored images using Inverse Filter do not look clear compared to those that were restored using the Wiener Filter and the Richardson-Lucy Algorithm, even though they have higher PSNR values. SSIM helped us determine that the PSNR is not always an accurate measure for comparing restored images.

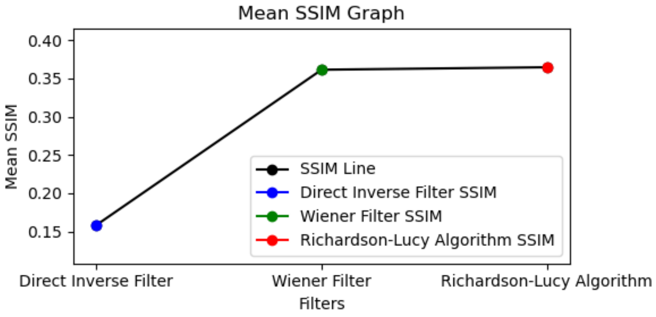


Fig. 5: Mean SSIM Graph

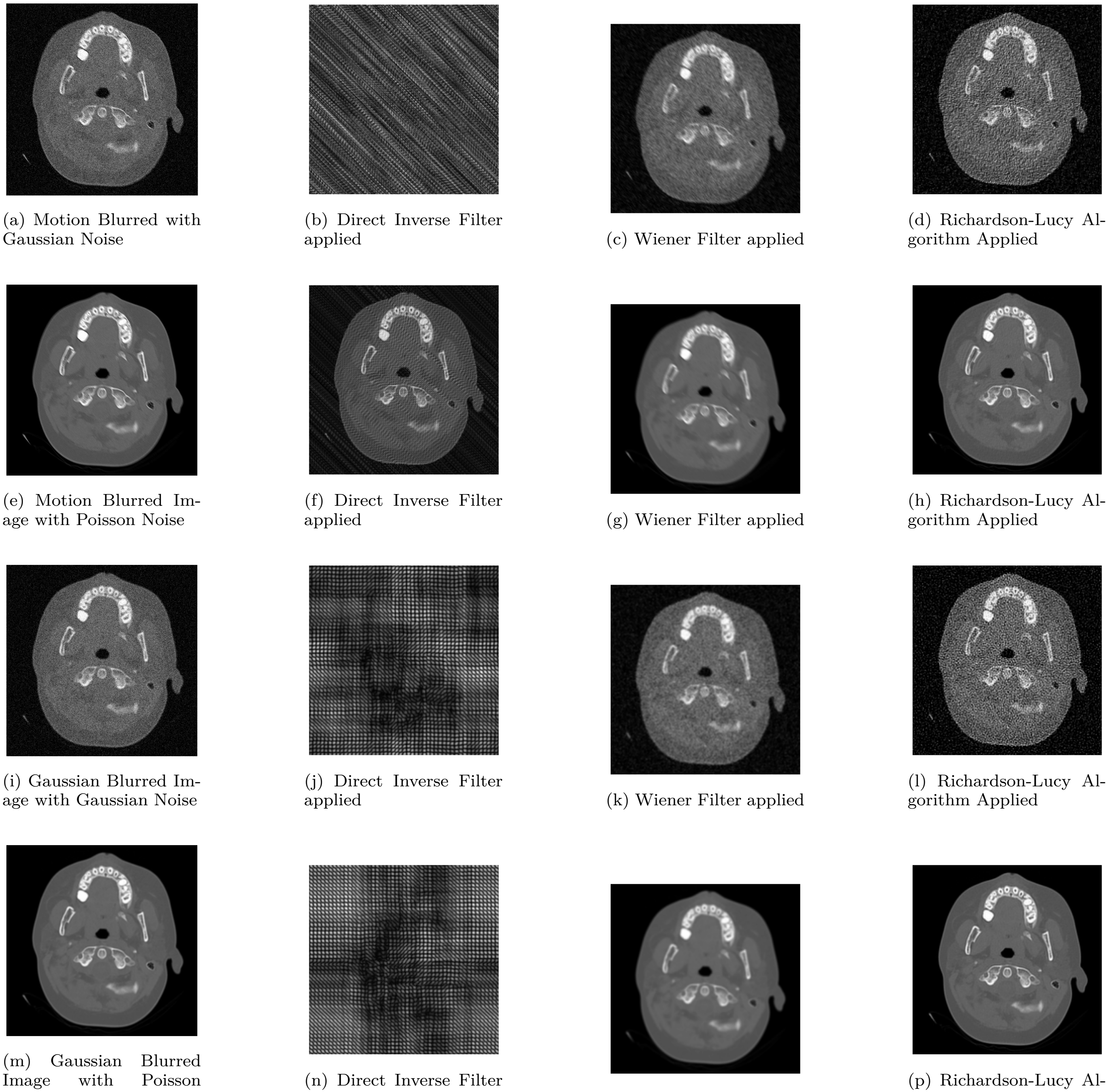
Based on the restored images and the PSNR and SSIM values, it shows that the Richardson-Lucy Algorithm performs the best compared to the Wiener Filter and the Inverse Filter, with the Wiener filter being a close second. This is because the Richardson-Lucy algorithm is an iterative method which makes it more capable of handling noise and blur with a known point spread function.

VIII. Conclusion and Future Work

Based on our results and data, we determined that the Richardson-Lucy Algorithm was the best option for restoring medical images due to having an iterative method that is more suitable for handling noise and blur. The Wiener Filter also restored the images very well, as it had slightly lower PSNR and SSIM values than the Richardson-Lucy Algorithm. We also found out that the PSNR evaluation metric is not always an accurate measure, as some of the restored images using Direct Inverse Filter do not look clear compared to those that were restored using the Wiener Filter or the Richardson-Lucy Algorithm. Readers of our work will come away with a strong understanding of the Inverse Filter, Wiener Filter, and Richardson-Lucy algorithm.

A limitation we had in this project is that we were only able to use two different evaluation metrics to evaluate our results, which are PSNR and SSIM. Due to time constraints, we were not able to make any improvements on the filters that were used in restoring the blurred and noisy images. Functions from libraries were used for the blurring and restoration methods.

Future work in restoring medical images include using modern approaches, such as machine learning methods and deep learning methods, as well as improving on the Wiener Filter and the Richardson-Lucy Algorithm. More evaluation metrics may also be used in future work, such as the VIF (Visual Information Fidelity) and the ERGAS



(a) Motion Blurred with Gaussian Noise

(b) Direct Inverse Filter applied

(c) Wiener Filter applied

(d) Richardson-Lucy Algorithm Applied

(e) Motion Blurred Image with Poisson Noise

(f) Direct Inverse Filter applied

(g) Wiener Filter applied

(h) Richardson-Lucy Algorithm Applied

(i) Gaussian Blurred Image with Gaussian Noise

(j) Direct Inverse Filter applied

(k) Wiener Filter applied

(l) Richardson-Lucy Algorithm Applied

(m) Gaussian Blurred Image with Poisson Noise

(n) Direct Inverse Filter applied

(o) Wiener Filter applied

(p) Richardson-Lucy Algorithm Applied

Fig. 6: Deblurring Results where first column is the blurred and noisy image, second column is where Inverse Filter is applied, third is where Wiener Filter is applied, and fourth is where Richardson-Lucy Algorithm is applied.

(Relative Dimensionless Global Error in Synthesis) for more accurate comparisons of restoring the images.

References

[1] E. J. Van Beek and E. A. Hoffman, "Functional imaging: Ct and mri," *Clinics in Chest Medicine*, vol. 29, no. 1, p. 195–216, Feb. 2008. [Online]. Available: https://pmc.ncbi.nlm.nih.gov/articles/PMC2435287/

[2] J. Nakashima and H. Duong, "Radiology, image production and evaluation," *StatPearls - NCBI Bookshelf*, Jul. 2023. [Online]. Available: https://www.ncbi.nlm.nih.gov/books/NBK553145/

[3] W. Pijal, I. Pineda, and M. E. Morocho-Cayamcela, "Comparative study of image degradation and restoration techniques," *Information and Communication Technologies*, pp. 253–265, 10 2022.

[4] Z. Yang, H. Chen, Z. Qian, Y. Yi, H. Zhang, D. Zhao, B. Wei, and Y. Xu, "All-in-one medical image restoration via task-adaptive routing," 2024. [Online]. Available: https:

//arxiv.org/abs/2405.19769

[5] H. Zhao, Z. Liu, J. Tang, B. Gao, Q. Qin, J. Li, Y. Zhou, P. Yao, Y. Xi, Y. Lin, H. Qian, and H. Wu, "Energy-efficient high-fidelity image reconstruction with memristor arrays for medical diagnosis," *Nature Communications*, vol. 14, no. 1, Apr. 2023. [Online]. Available: https://www.nature.com/articles/s41467-023-38021-7

[6] H.-M. Zhang and B. Dong, "A review on deep learning in medical image reconstruction," *Journal of the Operations Research Society of China*, vol. 8, no. 2, p. 311–340, Jan. 2020. [Online]. Available: http://dx.doi.org/10.1007/s40305-019-00287-4

[7] M. Lata, S. Ghosh, F. Bobi, and M. Yousuf, "Novel method to assess motion blur kernel parameters and comparative study of restoration techniques using different image layouts," *2016 International Conference on Informatics, Electronics and Vision (ICIEV)*, pp. 367–372, 05 2016.

[8] A. Sheta, "Restoration of medical images using genetic algorithms," *2017 IEEE Applied Imagery Pattern Recognition Workshop (AIPR)*, pp. 1–8, 10 2017.

[9] R. Altaie, "Restoration for blurred noisy images based on guided filtering and inverse filter," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 11, p. 1265, 04 2021.

[10] T. Stathaki, "Digital image processing: Image restoration - inverse filtering," https://www.commsp.ee.ic.ac.uk/~tania/teaching/DIP%202014/Image%20Restoration%20Pres.pdf [Accessed: (10/27/2024)].

[11] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, April 2004.

[12] M. D. Rudi Rottenfusser, Erin Wilson, "Education in microscopy and digital imaging," https://zeiss-campus.magnet.fsu.edu/articles/basics/psf.html [Accessed: (10/27/2024)].

[13] H. I. Academy, "Point spread function," https://svi.nl/Point-Spread-Function-(PSF) [Accessed: (10/27/2024)].

[14] MathWorks, "Deblurring images using the lucy-richardson algorithm," https://www.mathworks.com/help/images/deblurring-images-using-the-lucy-richardson-algorithm.html [Accessed: (10/27/2024)].

[15] A. Horé and D. Ziou, "Image quality metrics: Psnr vs. ssim," pp. 2366–2369, 2010.

[16] D. Poobathy and R. Chezian, "Edge detection operators: Peak signal to noise ratio based comparison," *International Journal of Image, Graphics and Signal Processing*, vol. 6, pp. 55–61, 09 2014.

[17] I. Bakurov, M. Buzzelli, R. Schettini, M. Castelli, and L. Vanneschi, "Structural similarity index (ssim) revisited: A data-driven approach," *Expert Systems with Applications*, vol. 189, p. 116087, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417421014238

[18] K. Seshadrinathan, H. Sheikh, Z. Wang, and A. Bovik, "Information theoretic approaches to image quality assessment," *Handbook of Image and Video Processing*, 12 2005.