Tomographic Image Reconstruction utilizing Maximum Likelihood Expectation-Maximation (ML-EM) Algorithm

Arabella Mae M. Aduviso
BSCS-NS-3A
Technological University of the Philippines
Ayala Boulevard, Ermita, Manila, Philippines
arabellamae.aduviso@tup.edu.ph

Juliana Anne M. Capoquian
BSCS-NS-3A
Technological University of the Philippines
Ayala Boulevard, Ermita, Manila, Philippines
simone.caturla@tup.edu.ph

Luis Pocholo Caducio
BSCS-NS-3A
Technological University of the Philippines
Ayala Boulevard, Ermita, Manila, Philippines
luispocholo.caducio@tup.edu.ph

Simone Arabella B. Caturla
BSCS-NS-3A
Technological University of the Philippines
Ayala Boulevard, Ermita, Manila, Philippines
julianaanne.capoquian@tup.edu.ph

Jieson R. Orain
BSCS-NS-3A
Technological University of the Philippines
Ayala Boulevard, Ermita, Manila, Philippines
jieson.orain@tup.edu.ph

***Abstract*—Computed tomography (CT) scans are essential in medical imaging, providing detailed cross-sectional images of the body that aid in diagnosing various conditions. Iterative reconstruction methods, especially the Maximum Likelihood - Expectation Maximization (ML-EM) algorithm, have significantly improved CT imaging by reducing noise, lowering radiation doses, enhancing image details, and speeding up reconstruction times. This thesis examines the application of the ML-EM algorithm in tomographic image reconstruction, focusing on its iterative process that refines image estimates until they closely match the true object. We implemented the ML-EM algorithm and tested it using the Shepp-Logan Phantom, a standard test image. Our results indicate that as the number of iterations increases, the reconstructed images become progressively clearer and more detailed, showcasing the effectiveness of the ML-EM algorithm. Moreover, the algorithm effectively minimizes discrepancies between the measured and estimated data, producing high-quality images suitable for medical diagnosis and treatment planning. Future research should focus on optimizing the number of iterations, enhancing the algorithm, leveraging hardware acceleration, and conducting comparative studies with other reconstruction algorithms.**

Keywords*— *Image Reconstruction, Iterative Reconstruction Methods, Maximum Likelihood Expectation Maximization.

I. INTRODUCTION

Computed tomography (CT) scans reconstruct x-ray attenuation data into an image which is frequently used to help medical practitioners to visualize pathology and anatomy [4]. Iterative reconstruction has been used to advance CT imaging using its capacity to reduce noise in the image that is frequently associated with filtered back projection, reduction in radiation dosage, enhancement of details, and reduced construction time. This sophisticated algorithm uses an initial image assumption and adjusts align with real-time measured values [5].

This paper will focus on iterative reconstruction using maximum likelihood - expectation maximization (ML-EM) algorithm, an algorithm that primarily focuses on finding an approximate solution that best fits the initial estimation of an image [6].

II. REVIEW OF RELATED LITERATURE

A. Tomographic Image Reconstruction

Tomographic image reconstruction is a mathematical process that produces cross-sectional pictures of the body that are used in CT scans, which offers more detailed information than regular x-rays. It is crucial in diagnosing diseases or injuries such as tumors, clots, bone fractures and more [3]. Initially, computer technology limited the capabilities of image reconstruction in early scanners, but advancements over the past decade have made it widely adopted in the industry. One of the most significant advantages of modern scanners is their ability to reduce noise in the final image, associated with filtered back projection, without high radiation dose to patients.

Tomographic image reconstruction uses various algorithms that are designed to handle large amounts of data generated during a CT scan and reconstruct images that help medical practitioners in diagnosis and treatment planning. Different algorithms produce different results in image quality and efficiency of reconstruction, making it an important component of CT technology [5].

B. Maximum Likelihood – Expectation Maximization (ML-EM)

Medical imaging and tomography use the ML-EM algorithm to reconstruct images using an iterative method that maximizes the likelihood of the observed data given a set of estimated images. From an initial guess, the algorithm will refine the image iteratively until it can resemble the true object using the sonogram data [2]. The formula for ML-EM algorithm is stated below:

$$x^{k+1} = \frac{x^k}{A^T 1} A^T \frac{m}{Ax^k}$$

To get the image at the next iteration (x^{k+1}), the formula divides the current estimate of the image (x^k) to the transpose of the system matrix (A^T) and multiply it to a vector with all elements set to one (*1*) to compute the sensitivity of the image. It is multiplied again the the transpose of the system matrix and multiplied with the measured data obtained from the imaging system (m) that is divided with the forward projection of the current estimate of the image (Ax) at it each iteration (k). Basically, the formula will continuously update the image estimate by comparing it to the actual data that you have, figuring out their differences, and correcting the image estimate little by little until it becomes a clearer image [1].

C. Forward and Back Projection

Forward and back projection uses data collected from multiple angles of the body to reconstruct an image. 2D cross-sectional images from 1D projections are obtained from transforming radiation measurements that involve complex algorithms and mathematical formulas. This technique is crucial for the accurate diagnosis and treatment planning in medical imaging.

Using both forward and backward projection is crucial for updating the reconstructed image in each iteration of the ML-EM algorithm. In each iteration of the algorithm, forward projecting the current estimate is continuously compared to the actual sonogram data until it can resemble the true object. Meanwhile, the correction image obtained from the backward projection of the ratio sonogram is used by the ML-EM algorithm to update the reconstructed image [4].

III. METHODOLOGY

```
from skimage.data import shepp_logan_phantom
from skimage.transform import radon, iradon, rescale
import matplotlib.pyplot as plt
import numpy as np
```

The early part of the system source code requires importing all the python libraries needed by the system. The following libraries are imported in accordance with their uses in the program.

1. Skimage: is a collection of algorithms for image processing and computer vision. This is used for generating the sample object image, the Shepp Logan Phantom, as well as manipulating its size. This library is mainly used to access functions for MLEM calculation, namely: randon and irandon.
2. Matplotlib: is known for its use in visualization. In the program, this is utilized for creating and updating plots to visualize the object, sinogram, forward projection, ratio sinogram, pack-projected ratio, and the reconstructed image at each iteration.
3. Numpy: is used for numerical operations, array manipulations, and initializing data structures needed for the reconstruction algorithm.

```
plt.ion()

activity_level = 0.1
true_object = shepp_logan_phantom()
true_object = rescale(activity_level * true_object, 0.5)

fig, axs = plt.subplots(2, 3, figsize=(20,10))
axs[0,0].imshow(true_object, cmap='Greys_r');   axs[0,0].set_title('Object')
```

These series of codes enable the interactivity mode of the plots, by which it dynamically updates the plot as changes are made to them. It also sets Shepp Logan Phantom, a standard test image used in image processing especially for simulating tomographic reconstruction, as the object image and rescaling it. By using pyplot in matplotlib library, the system creates a grid of subplots to show the visualizations generated by the program. The last line of codes displays the object image to be reconstructed at the first plot.

```
azi_angles = np.linspace(0.0, 180.0, 180, endpoint=False)
sinogram = radon(true_object, azi_angles, circle=False)


axs[0, 1].imshow(sinogram.T, cmap='Greys_r');   axs[0,1].set_title('Sinogram')
```

Generation of the sinogram, a representation of an image as a set of projections taken at different angles, is implemented through these codes. This happens with the use of *radon* function, which deconstructs and computes the radon transform of the object image at specified angles, resulting in a sinogram. The *azi-angles* is an array of 180 angles from 0 to 179 degrees for the Radon transform. These angles represent the directions from which projections (radon transforms) will be taken. The sinogram is then shown in the subplot.

```
mlem_rec = np.ones(true_object.shape)
sino_ones = np.ones(sinogram.shape)
sens_image = iradon(sino_ones, azi_angles, circle =False, filter_name=None)
```

The first line of this sequence of codes creates a 2D array (image) of ones with the same shape as the object image. The *mlem_rec* contains the initial guess for the reconstructed image, this will be iteratively updated to converge towards a more accurate reconstruction of the original object. This array, *sino_ones*, represents a sinogram where each projection is a line of ones. It is used to compute the sensitivity image. Subsequently, the *iradon* function from the *skimage* library performs the back-projection of the *sino-ones* array to create the sensitivity image, which is crucial for normalizing the updates in the ML-EM iterations.

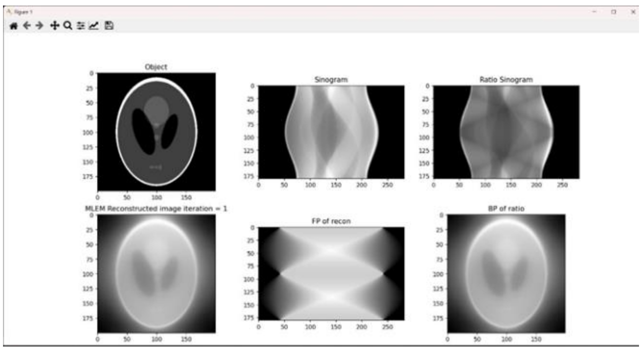
```
for iter in range(500):
    fp = radon(mlem_rec, azi_angles, circle=False)
    ratio = sinogram / (fp + 0.000001)
    correction = iradon(ratio, azi_angles, circle =False, filter_name=None) / sens_image

    axs[1,1].imshow(fp.T, cmap="Greys_r");  axs[1,1].set_title('FP of recon')
    axs[0,2].imshow(ratio.T, cmap="Greys_r");  axs[0,2].set_title('Ratio Sinogram')
    axs[1,2].imshow(correction, cmap="Greys_r");  axs[1,2].set_title('BP of ratio')

    mlem_rec = mlem_rec * correction
    axs[1, 0].imshow(mlem_rec, cmap='Greys_r');   axs[1,0].set_title('MLEM Reconstructed image iteration = %d' % (iter+1))
    plt.show()
    plt.pause(0.05)

plt.show(block=True)
```

These lines of code are the crucial parts of the source code. The for loop contains the whole iterative process of the ML-EM algorithm. The number of iterations can be set by the developers. Inside the loop, the computation of the forward projection of the current reconstructed image, *mlem_rec*, takes place. It then generates a new sinogram, *fp*, from the current estimate of the reconstructed image. Variable *ratio* is the ratio calculated between the measured object image's sinogram and the forward projection, *fp*. The *iradon* performs inverse Radon transform or back-projection of the *ratio* sinogram to calculate the correction image. These variables are essential to compute and generate updated reconstructed image in every iteration. Specifically, the codes '*mlem_rec* = *mlem_rec* * *correction*' updates the reconstructed image by multiplying it element-wise with the correction image. This step incorporates the correction derived from the ratio and back-projection to refine the estimate of the reconstructed image. Additionally, in each iteration, the forward projection, ratio sinogram, correction image or back-projection of ratio, and the reconstructed image are being updated and displayed in the subplots. The visualizations help to monitor the reconstruction process and understand how the image evolves with each iteration.

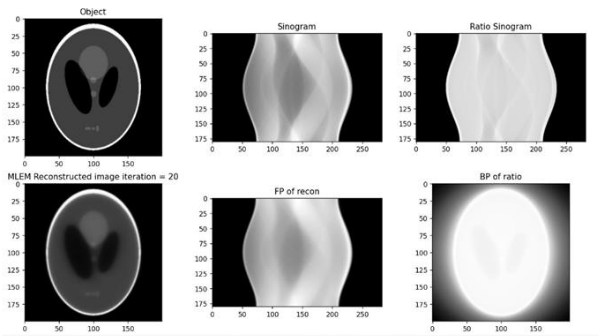
IV. IMPLEMENTATION

To test the effectivity of the Maximum Likelihood Expectation Maximization (ML-EM) algorithm in tomographic image reconstruction, below are the outputs of the reconstructed images generated by the system in different iterations.

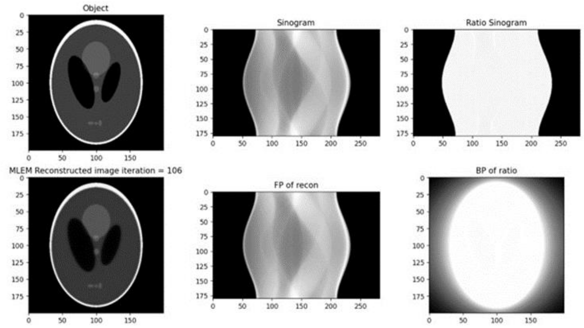
1st iteration:



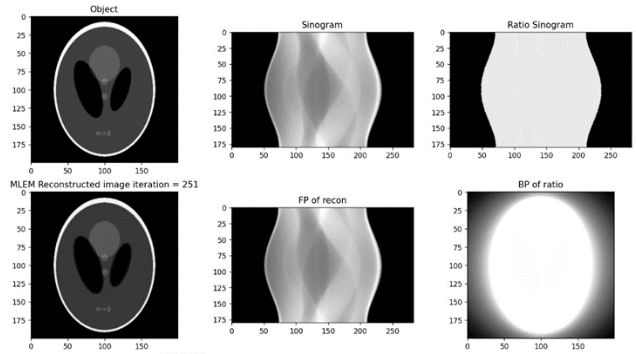
20th iteration:

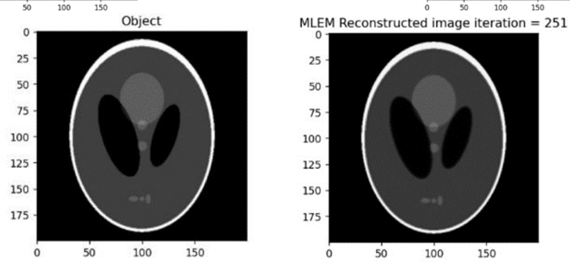


150th iteration:

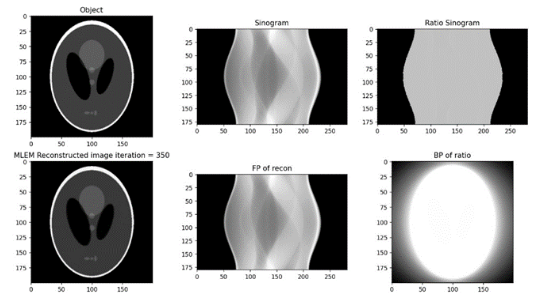


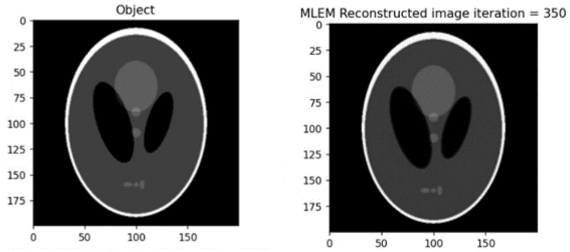
250th iteration:



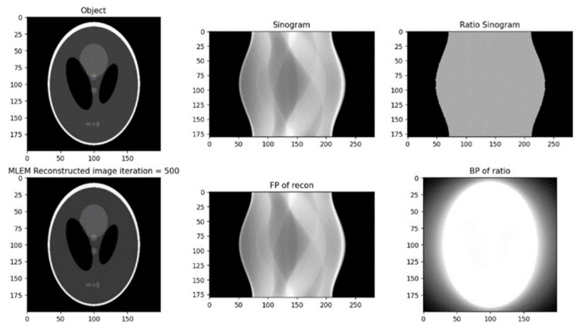


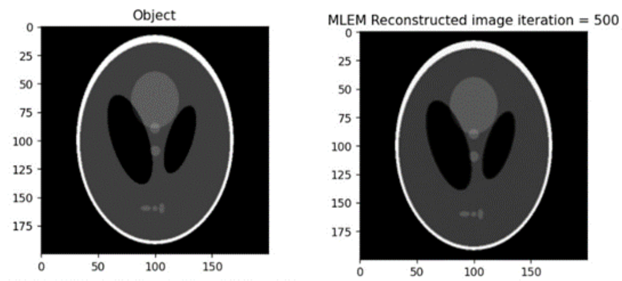
350th iteration:





500^{th} iteration:





[1] Considering these outputs, the reconstructed images are initially blurry than the object image, which appears in the first, 20^{th}, 150^{th}, and 250^{th} iterations of the implementation. Nonetheless, it is noted that noticeable improvements are seen throughout these intervals, particularly during the early 100 iterations. In the 350th and 500^{th} iterations, the reconstructed image are most likely identical to the object image, although minimal changes from the two reconstructed image are observed. Furthermore, it is observed that the more iterations the system goes through, the longer it takes for it to generate a new reconstructed image. Nonetheless, it creates new reconstructed images identical to the object image. As a result, the more iterations that are done, the more accurate the reconstructed image is to the original image.

Further observation also infers that the sinogram of the object image (Sinogram) and the forward projection or sinogram of the current reconstructed image (FP of recon) get similar in each iteration. The 'Ratio Sinogram' indicates that every projection line in the 'Sinogram' is also can be seen in the 'FP of recon'. Initially, the discrepancies between the FP of recon and the Sinogram are large, leading to a substantial correction. As the iterations progress, these discrepancies decrease, meaning the 'FP of recon' gets closer to the measured 'Sinogram' resulting in smaller ratios and corrections. The back projection of the ratio (BP of ratio) appears lighter in later iterations because the ratio values are smaller, as seen in the 'Ratio Sinogram', indicating fewer and smaller adjustments needed to refine the image. These concepts demonstrate the effectiveness of ML-EM as an iterative algorithm for tomographic image reconstruction.

V. CONCLUSION AND RECOMMENDATION

A. *Conclusion*

The Maximum Likelihood Expectation Maximization (ML-EM) algorithm has proven to be highly effective for tomographic image reconstruction, yielding accurate and high-quality images. Initially, the reconstructed images appear somewhat blurry, but as the number of iterations increases, the images become significantly clearer and more detailed. By the 350th and 500th iterations, the reconstructed images closely match the original object image, demonstrating the algorithm's ability to refine images iteratively to achieve high accuracy.

Additionally, the sinogram of the object image and the forward projection of the reconstructed image become increasingly similar with each iteration. This convergence shows that the ML-EM algorithm effectively reduces the discrepancies between the measured and estimated data, resulting in progressively smaller corrections and more precise reconstructions.

These findings confirm that the ML-EM algorithm is robust and reliable for tomographic image reconstruction. Its iterative process allows for continuous refinement and enhancement of the images, making it a valuable tool in medical imaging for accurate diagnosis and treatment planning.

B. *Recommendations*

- **Optimization of Iteration Number:** While increasing the number of iterations enhances image quality, it also leads to higher computational costs. Future research should aim to optimize the number of iterations to strike a balance between image accuracy and computational efficiency. Exploring techniques such as early stopping criteria based on convergence thresholds could be beneficial.
- **Algorithm Improvements:** Further improvements to the ML-EM algorithm could include incorporating advanced noise reduction techniques or developing hybrid models that combine ML-EM with other algorithms. These enhancements could potentially improve image quality and reduce reconstruction time.
- **Hardware Acceleration:** Utilizing hardware accelerators such as GPUs or TPUs to implement the ML-EM algorithm can greatly speed up the reconstruction process, making real-time or near-real-time applications feasible.

- **Comparative Studies:** Conducting comparative studies with other reconstruction algorithms would provide valuable insights into the strengths and weaknesses of ML-EM. This could guide future improvements and support wider adoption in the medical imaging community.

By addressing these recommendations, the ML-EM algorithm can be refined and optimized further, enhancing its utility in medical imaging and contributing to better patient outcomes through more accurate and detailed diagnostic images.

VI. REFERENCES

[2] Andrew Reader. (2021, December 21). Simple PYTHON code for iterative image reconstruction (also shown for a Jupyter notebook) [Video]. YouTube. https://www.youtube.com/watch?v=IhETD4nSJec

[3] Borman, S. (2009, January). The Expectation Maximization Algorithm: A short tutorial. Retrieved June 6, 2024, from https://people.csail.mit.edu/dsontag/courses/ml14/slides/EM_algorithm.pdf

[4] Computed Tomography (CT). (n.d.). National Institute of Biomedical Imaging and Bioengineering. https://www.nibib.nih.gov/science-education/science-topics/computed-tomography-ct#:~:text=The%20term%20%E2%80%9Ccomputed%20tomography%2C%E2%80%9D,images%2C%20or%

[5] Brooks, S. L. (1993). COMPUTED TOMOGRAPHY. Dental Clinics of North America/the Dental Clinics of North America, 37(4), 575–590. https://doi.org/10.1016/s0011-8532(22)00243-9

[6] Feger, J. (2024). Iterative reconstruction (CT). Radiopaedia.org. https://doi.org/10.53347/rid-51925

[7] Zeng, G. L. (2015). The ML-EM algorithm is not optimal for poisson noise. The ML-EM Algorithm Is Not Optimal for Poisson Noise. https://doi.org/10.1109/nssmic.2015.7582178