CS 736
Image Deblurring

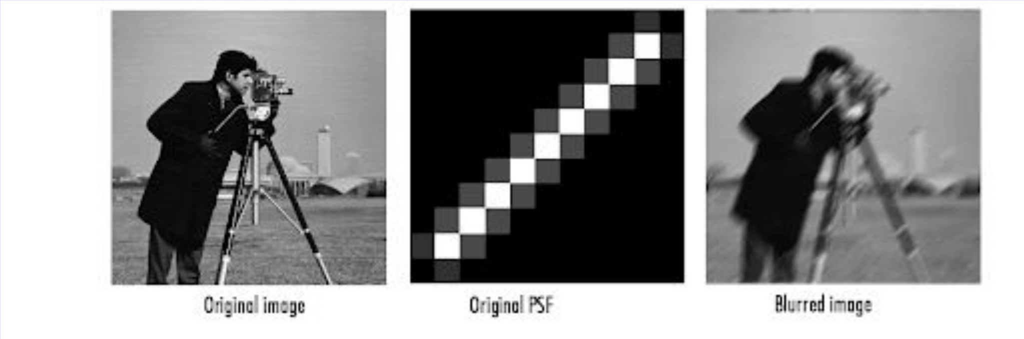
By:
Aadish Sethiya (210100001)
Harshit Agarwal (210020054)

Introduction

- Medical imaging is vital for accurate diagnosis and treatment planning.
- Image quality is often compromised by motion blur, noise, and hardware limitations.
- Deblurring algorithms help recover sharp, clear images from degraded inputs.
- This study explores and compares three key deblurring techniques for medical use.

Background

- Degraded images are modeled as the convolution of a sharp image with a Point Spread Function (PSF), plus noise.
- Deblurring involves reversing this process to estimate the original image.
- Deterministic deblurring methods provide stable and reproducible outcomes.
- Such methods are preferred in clinical settings for their reliability and interpretability.



Problem Description

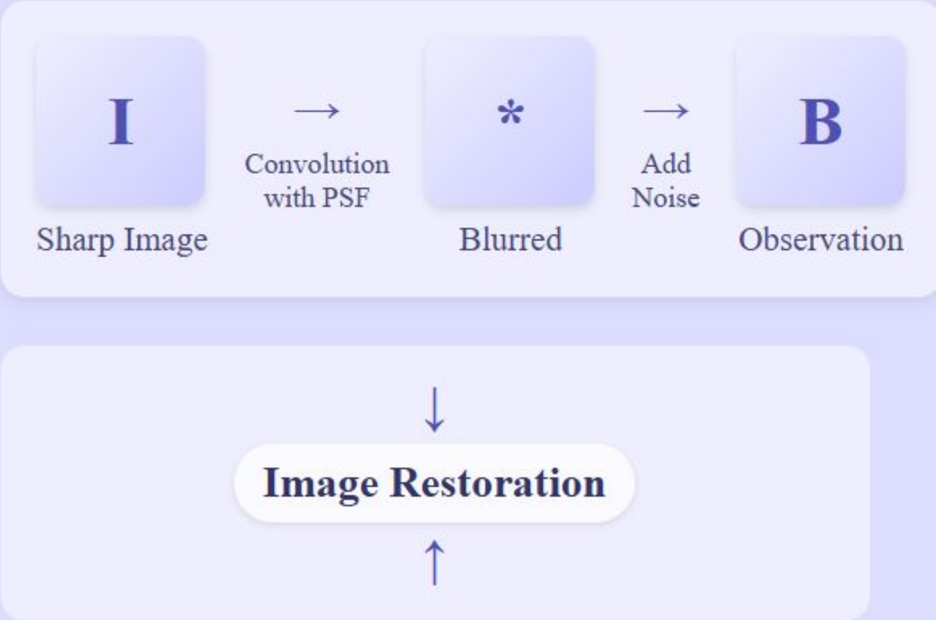
Image Deblurring Setup

- **Goal:** Restore a sharp image **I** from a blurred observation **B**, given the Point Spread Function (PSF) **P**

$$B = I * P + N$$

- $*$ denotes convolution operation
- **N** is additive noise (Gaussian or Poisson)

> **Challenge:** Invert the convolution process while effectively handling noise



Deblurring Algorithms

Richardson-Lucy Deconvolution

- Iterative algorithm for images with **Poisson noise**
- Based on **maximum likelihood estimation**

$$I^{(t+1)} = I^{(t)} \cdot [\ (B\ /\ (I^{(t)} * P + \epsilon)) * P^*\ ]$$

- $\mathbf{I^{(t)}}$: Current image estimate
- **B**: Observed blurred image
- **P**: Known PSF, **P***: 180° rotated PSF
- ϵ: Small constant to prevent division by zero

Note: Produces sharp results but sensitive to noise and iteration count

Algorithm Steps

1. Initialize with uniform or blurred image
2. Compute blurred estimate ($I^{(t)} * P$)
3. Calculate error ratio (B / blurred estimate)
4. Convolve with flipped PSF (P*)
5. Multiply result with current estimate
6. Update to new estimate $I^{(t+1)}$

↻ Repeat steps 2-6

Stop on convergence or fixed iteration count

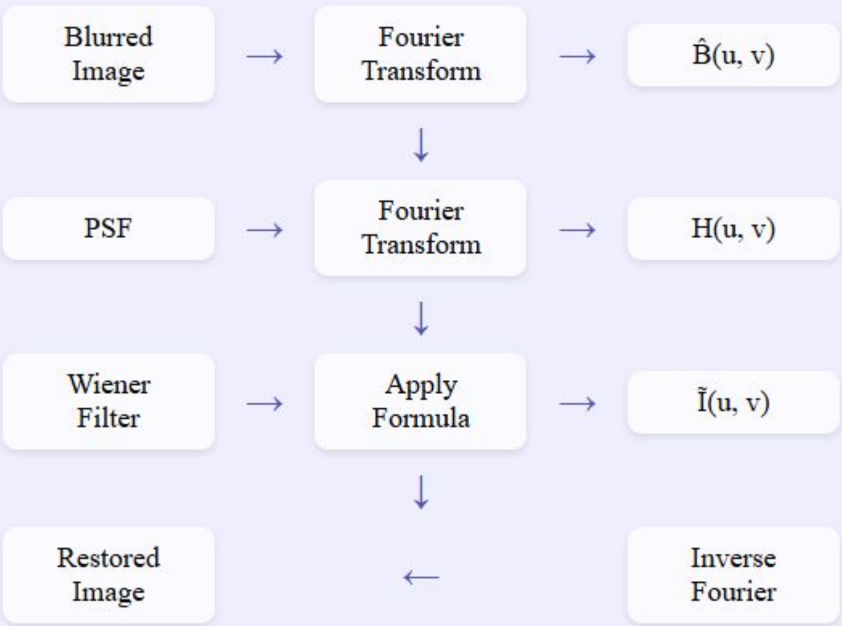
Wiener Deconvolution

- Operates in the **frequency domain** to restore blurred images
- Minimizes **mean square error** between original and estimated image

$$\tilde{I}(u, v) = [H^*(u, v) / (|H(u, v)|^2 + K)] \cdot \hat{B}(u, v)$$

- **$\tilde{I}(u, v)$, $\hat{B}(u, v)$**: Fourier transforms of restored/blurred images
- **$H(u, v)$**: Fourier transform of PSF
- **$H^*(u, v)$**: Complex conjugate of $H(u, v)$
- **K**: Noise-to-signal power ratio

- Final image obtained via **inverse Fourier transform**
- Effective for **Gaussian noise**; fast, non-iterative deblurring



Key Advantages

Fast	Optimal	Robust
Non-iterative	Minimizes MSE	Handles noise

Total Variation (TV) Regularization

- Solves an optimization problem:

$$\min_I \|I * P - B\|^2 + \lambda \cdot TV(I)$$

- Balances **data fidelity** with **edge-preserving regularization**
- Uses **iterative gradient descent** for optimization

$$I^{(t+1)} = I^{(t)} - \eta \cdot [\ (I^{(t)} * P - B) * P^* + \lambda \cdot \nabla TV(I^{(t)})\ ]$$

- **η**: Step size
- **λ**: Regularization parameter
- **∇TV(I)**: Gradient encouraging smooth regions while preserving edges

Optimization Process

1. Initialize with blurred image or estimate
2. Calculate data fidelity: $(I^{(t)} * P - B) * P^*$
3. Compute TV gradient: $\nabla TV(I^{(t)})$
4. Apply regularization weight λ
5. Update estimate with step size η

↻ Repeat steps 2-5

Until convergence or max iterations

Key Benefit: Preserves anatomical boundaries in medical images while reducing noise

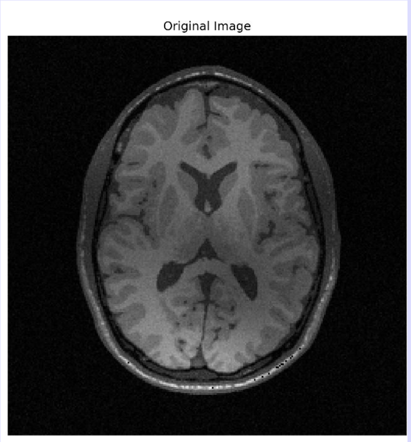
Results

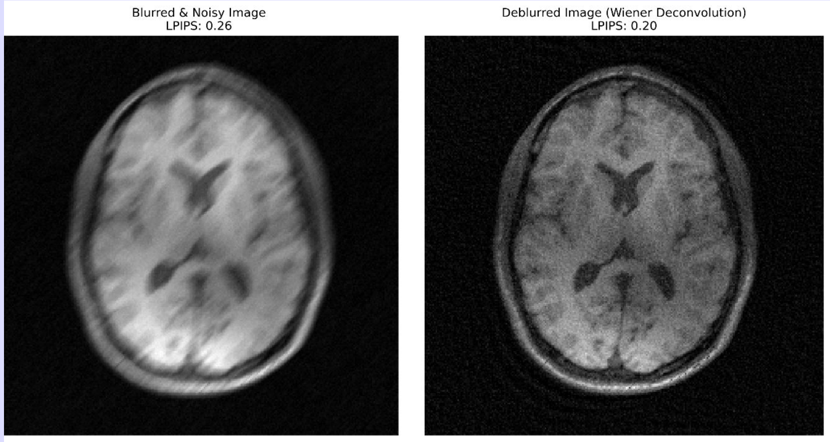
- Experiments conducted on three medical images:
  - X-Ray Lungs
  - Chest CT Scan
  - Brain MRI
- Images were synthetically blurred with a known PSF and corrupted with Gaussian noise.
- Applied three deblurring algorithms:
  - Richardson-Lucy Deconvolution
  - Wiener Deconvolution
  - TV Regularized Deblurring
- Evaluation methods:
  - Visual inspection
  - LPIPS metric to assess perceptual similarity

Brain MRI Image

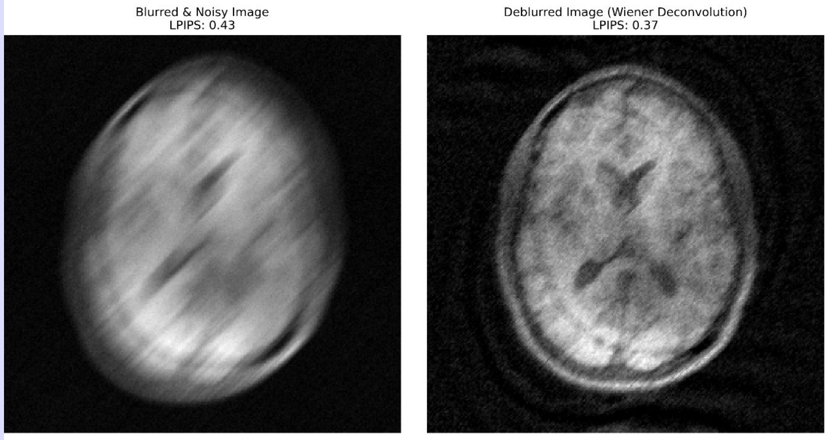
Wiener Deconvolution

Effect of Kernel Size

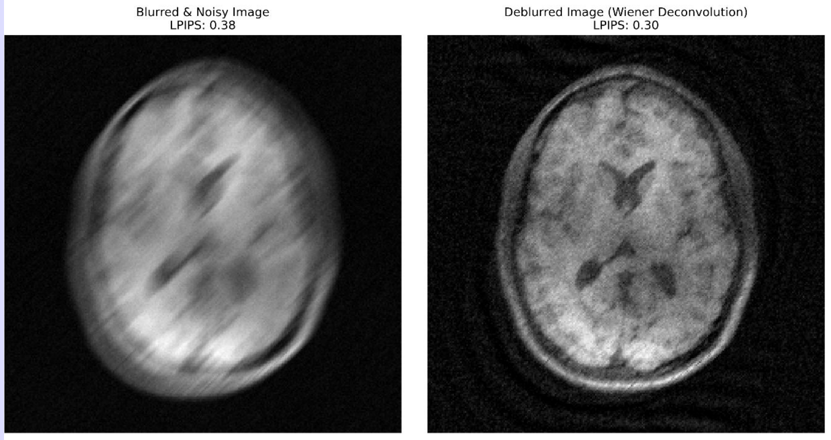




Kernel - 10, Angle - 45, Noise Sigma - 3



Kernel - 30, Angle - 45, Noise Sigma - 3

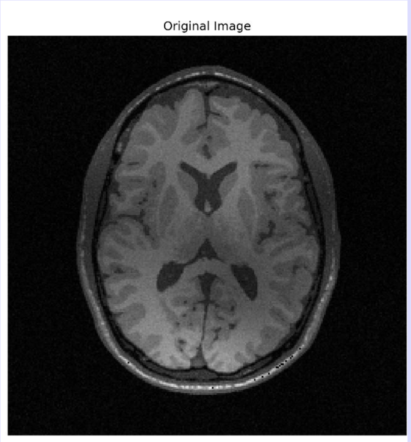


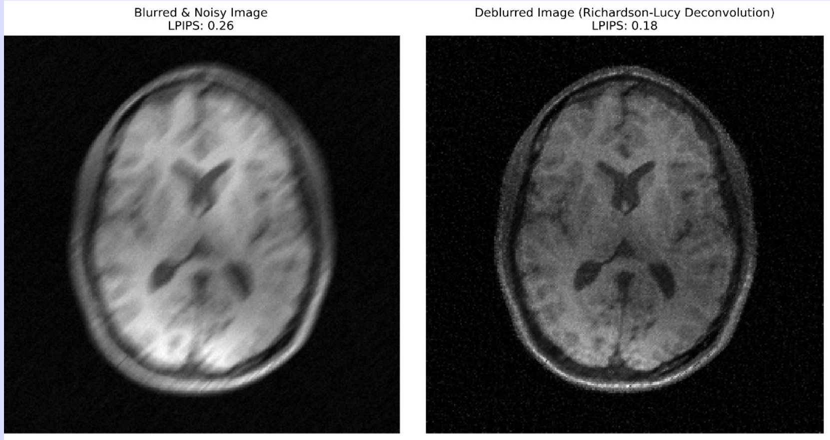
Kernel - 20, Angle - 45, Noise Sigma - 3

Brain MRI Image

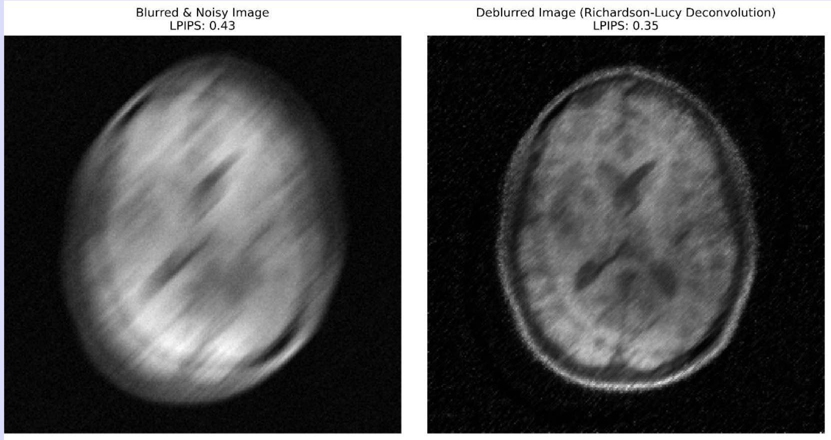
Richardson Lucy Deconvolution

Effect of Kernel Size

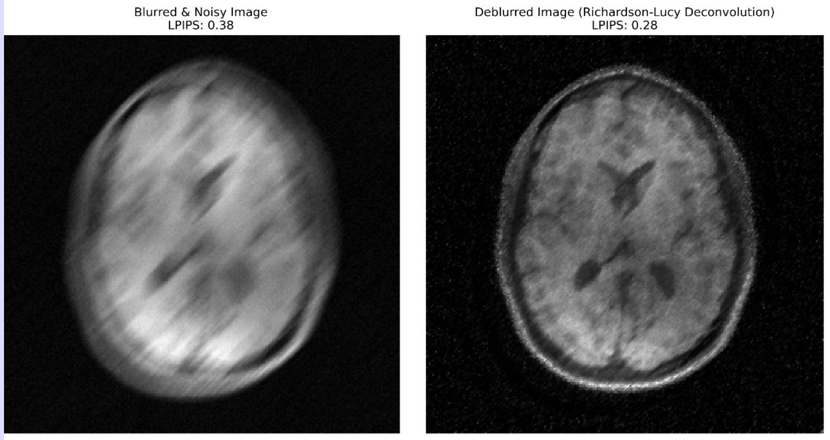




Kernel - 10, Angle - 45, Noise Sigma - 3



Kernel - 30, Angle - 45, Noise Sigma - 3

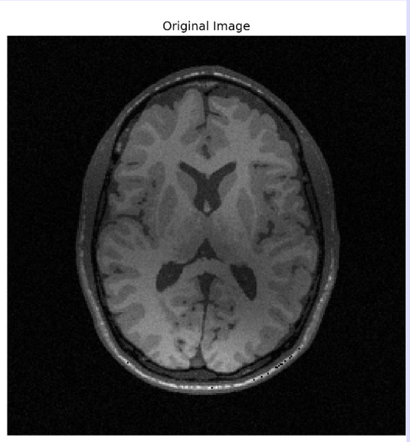


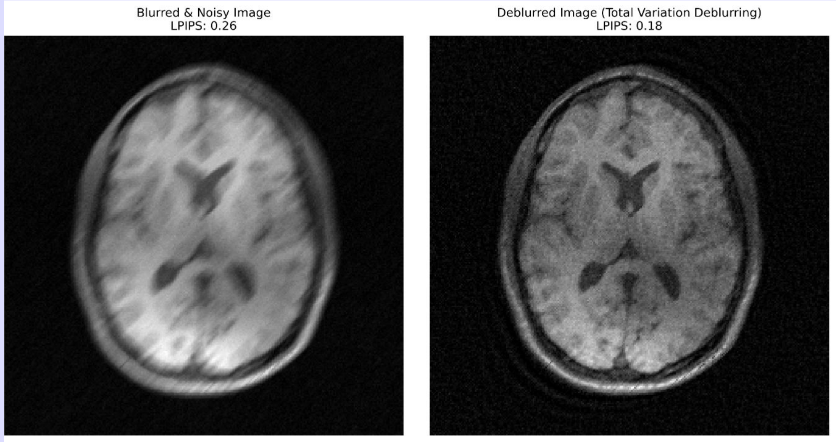
Kernel - 20, Angle - 45, Noise Sigma - 3

Brain MRI Image

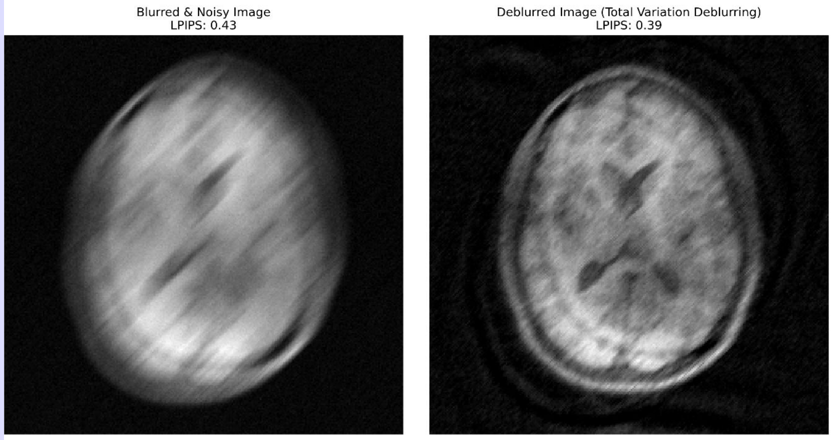
Total Variation Algorithm

Effect of Kernel Size

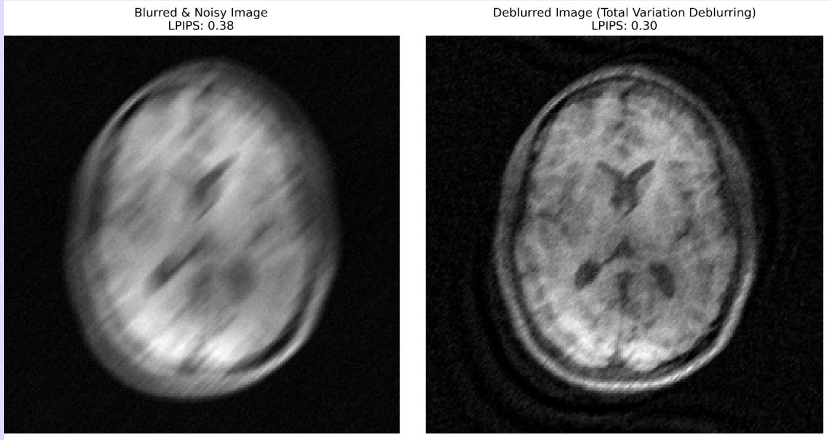




Kernel - 10, Angle - 45, Noise Sigma - 3



Kernel - 30, Angle - 45, Noise Sigma - 3

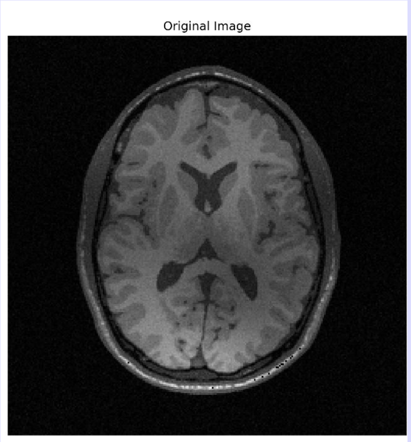


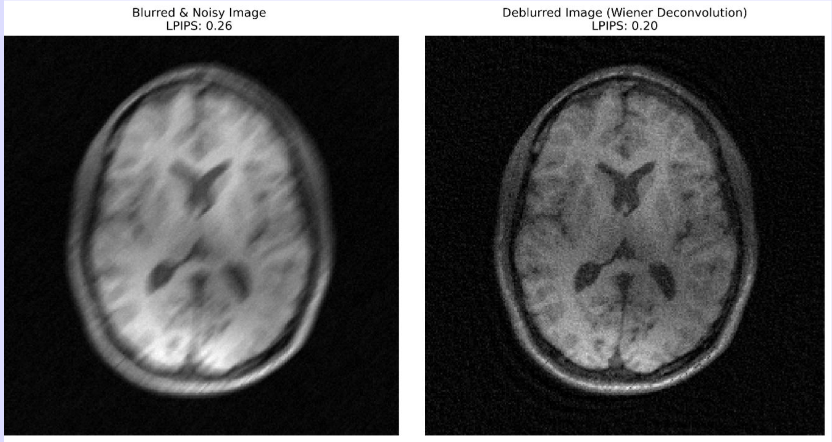
Kernel - 20, Angle - 45, Noise Sigma - 3

Brain MRI Image

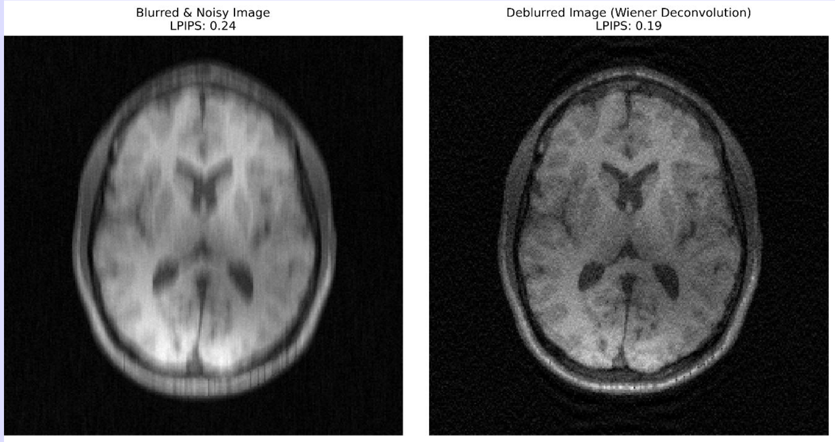
Wiener Deconvolution

Effect of Angle

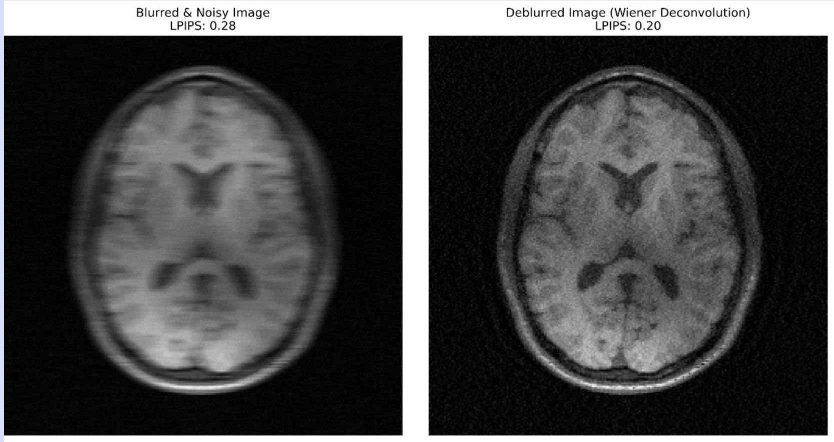




Kernel - 10, Angle - 45, Noise Sigma - 3



Kernel - 10, Angle - 90, Noise Sigma - 3

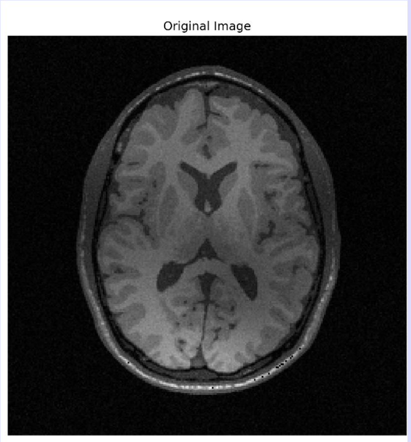


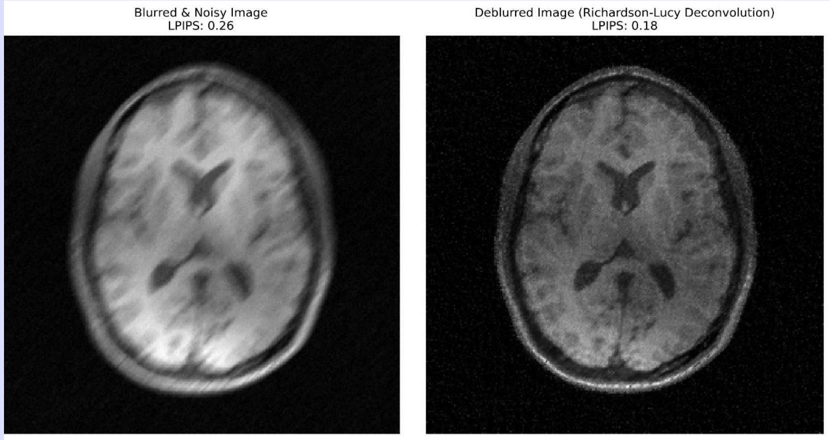
Kernel - 10, Angle - 0, Noise Sigma - 3

Brain MRI Image

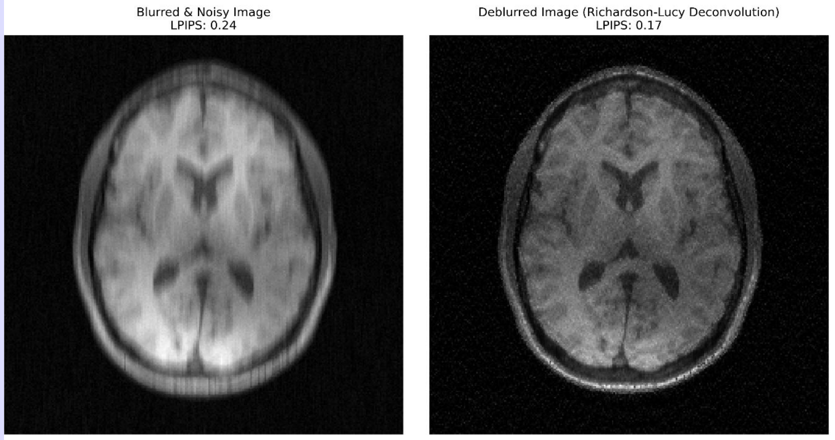
Richardson Lucy Deconvolution

Effect of Angle

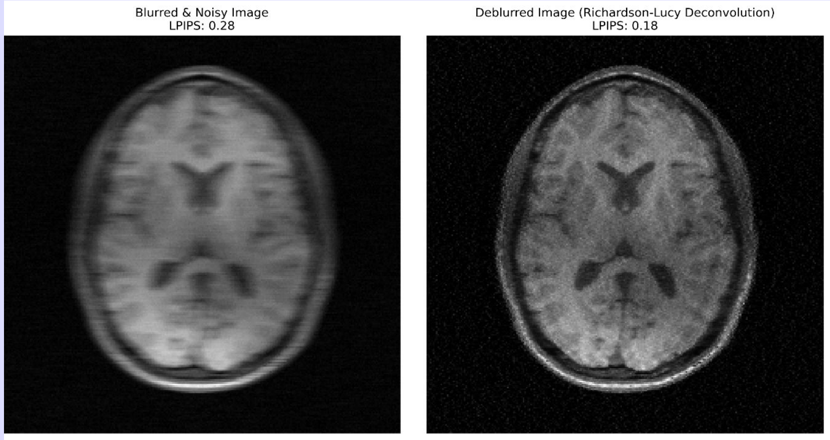




Kernel - 10, Angle - 45, Noise Sigma - 3



Kernel - 10, Angle - 90, Noise Sigma - 3

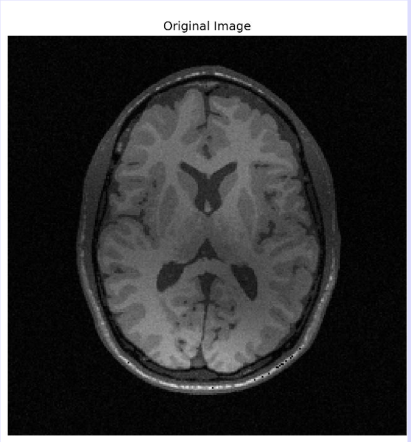


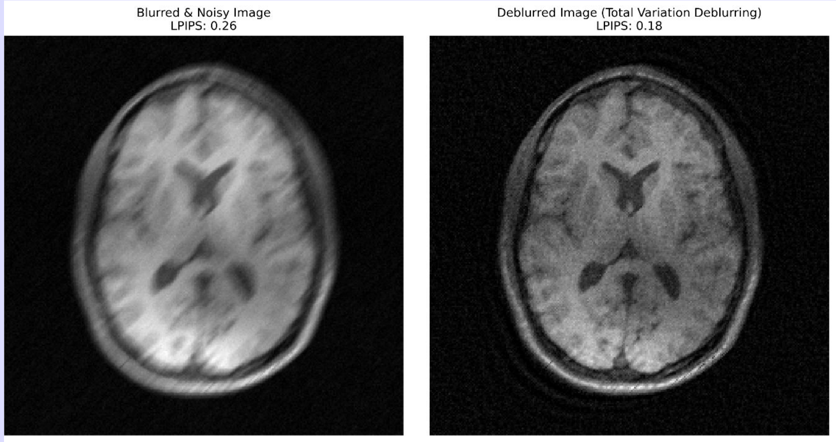
Kernel - 10, Angle - 0, Noise Sigma - 3

Brain MRI Image

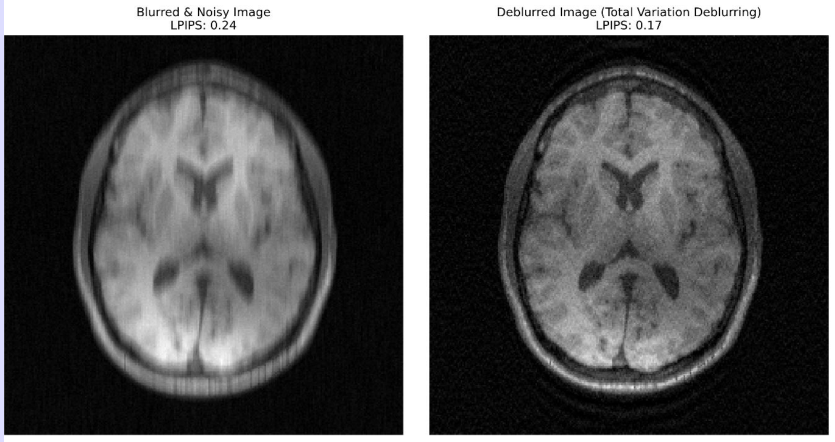
Total Variation Algorithm

Effect of Angle

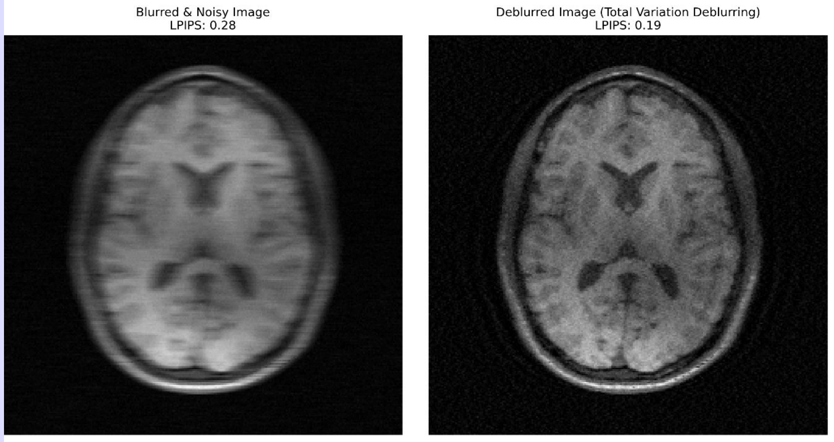




Kernel - 10, Angle - 45, Noise Sigma - 3



Kernel - 10, Angle - 90, Noise Sigma - 3

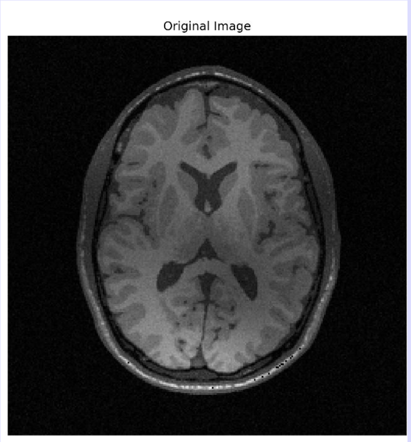


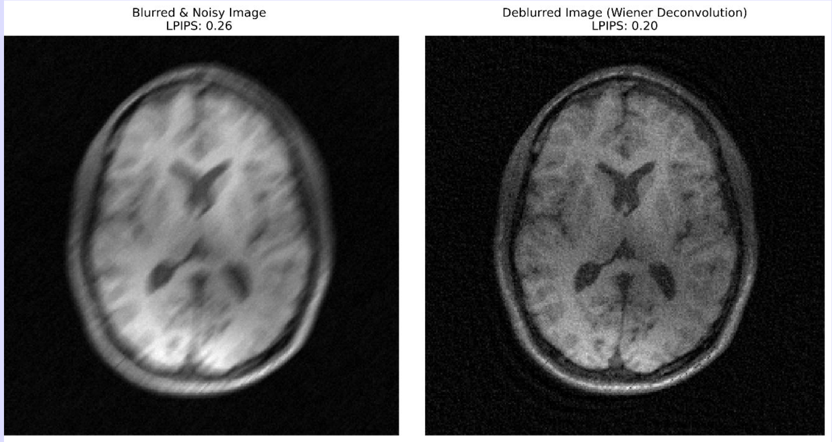
Kernel - 10, Angle - 0, Noise Sigma - 3

Brain MRI Image

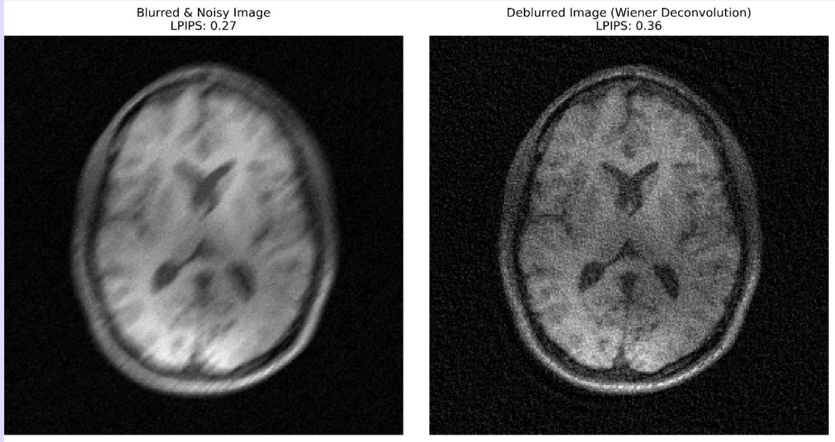
Wiener Deconvolution

Effect of Noise Sigma

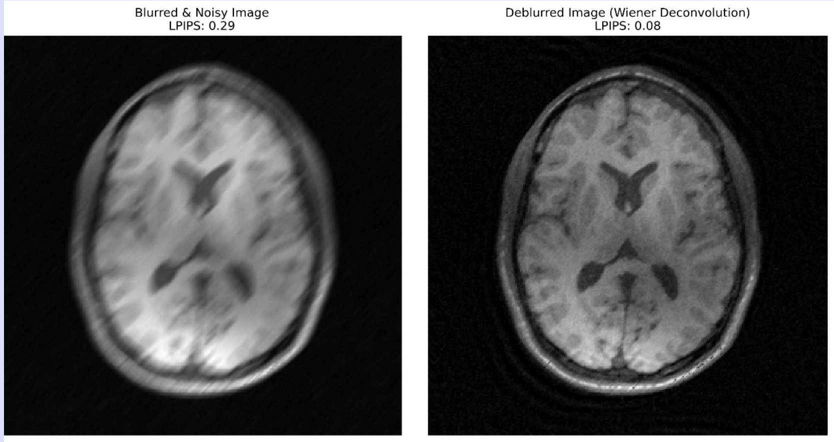




Kernel - 10, Angle - 45, Noise Sigma - 3



Kernel - 10, Angle - 45, Noise Sigma - 5

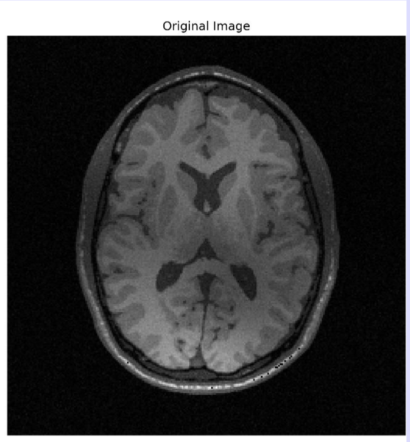


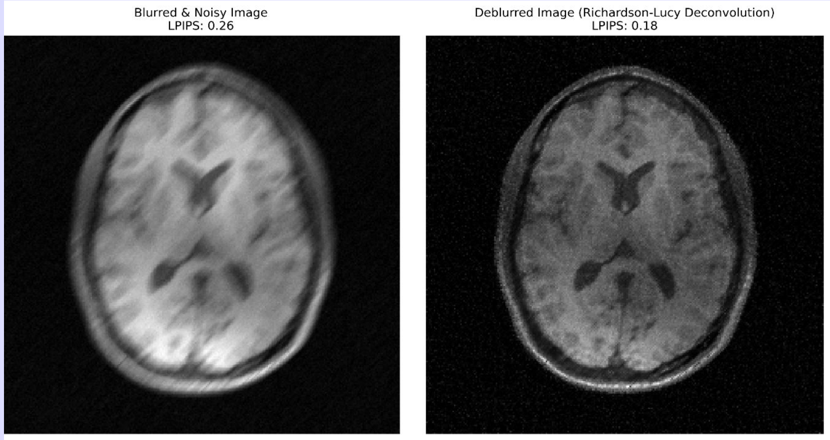
Kernel - 10, Angle - 45, Noise Sigma - 1

Brain MRI Image

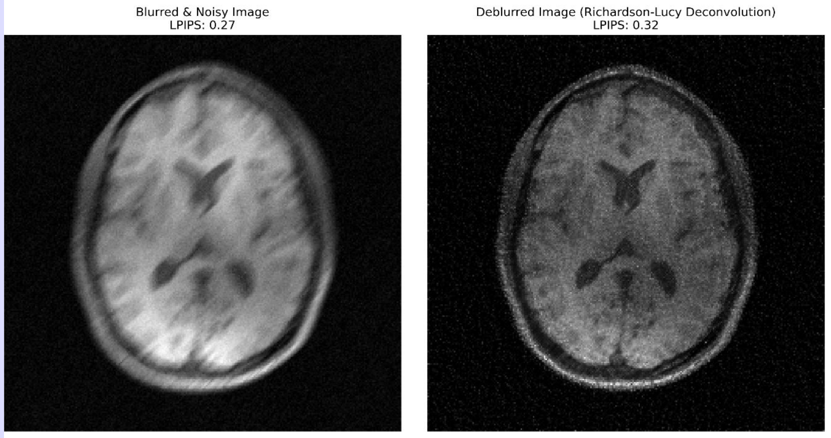
Richardson Lucy Deconvolution

Effect of Noise Sigma

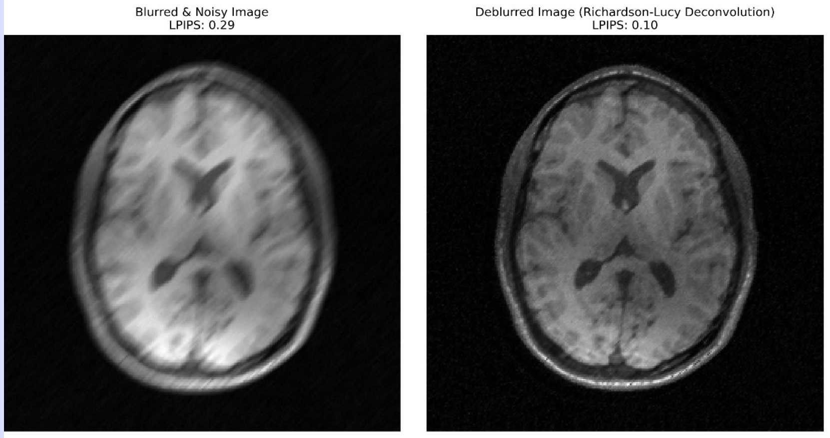




Kernel - 10, Angle - 45, Noise Sigma - 3



Kernel - 10, Angle - 45, Noise Sigma - 5

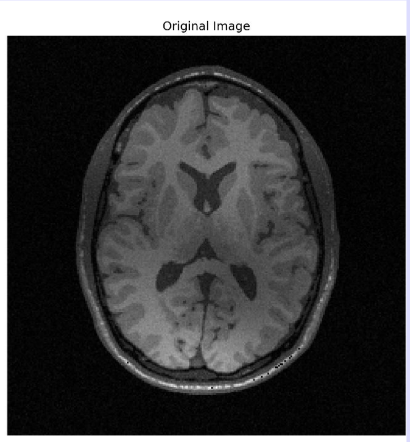


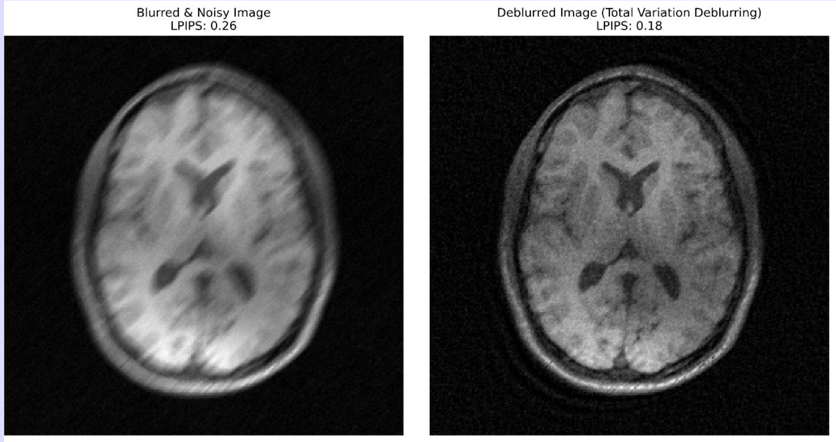
Kernel - 10, Angle - 45, Noise Sigma - 1

Brain MRI Image

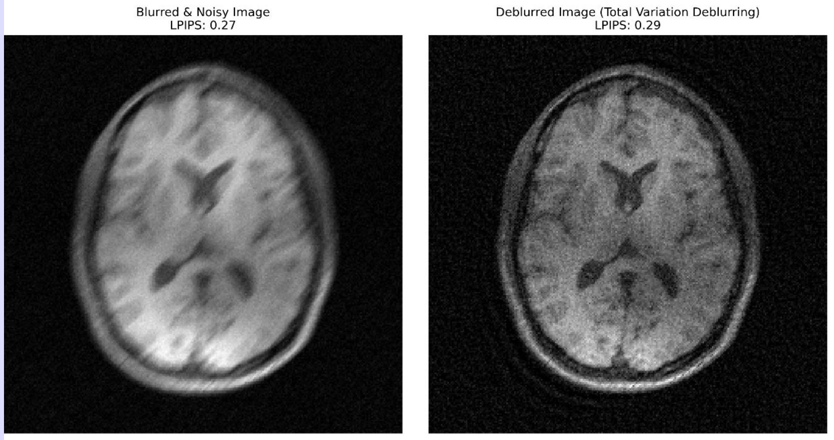
Total Variation Algorithm

Effect of Noise Sigma

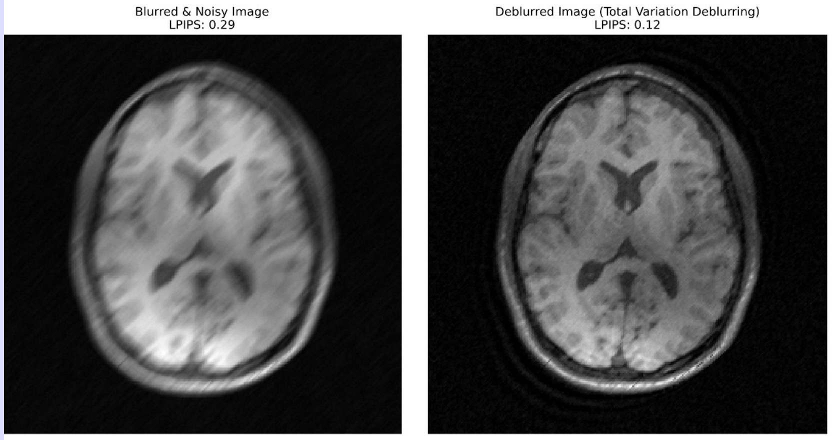




Kernel - 10, Angle - 45, Noise Sigma - 3



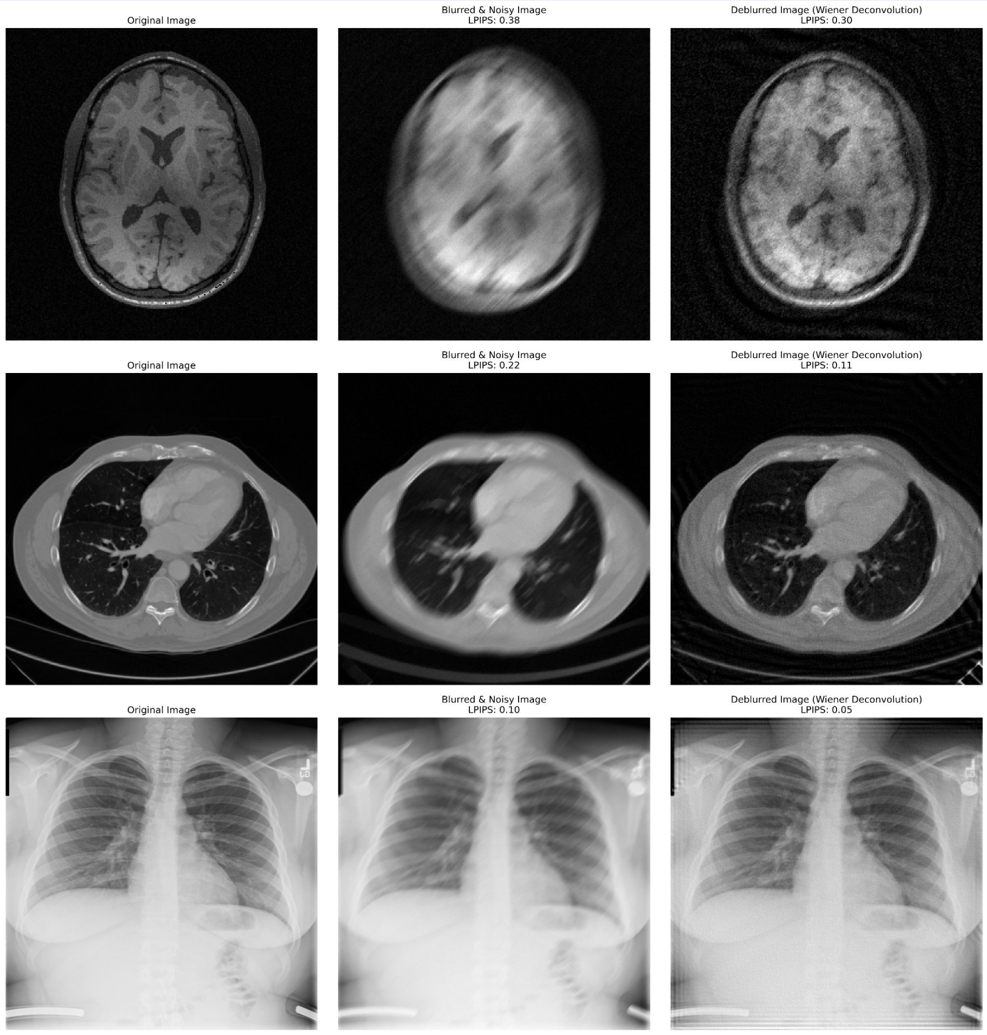
Kernel - 10, Angle - 45, Noise Sigma - 5



Kernel - 10, Angle - 45, Noise Sigma - 1

Wiener Deconvolution

Kernel Size: 20
Angle: 45
Noise Sigma: 3



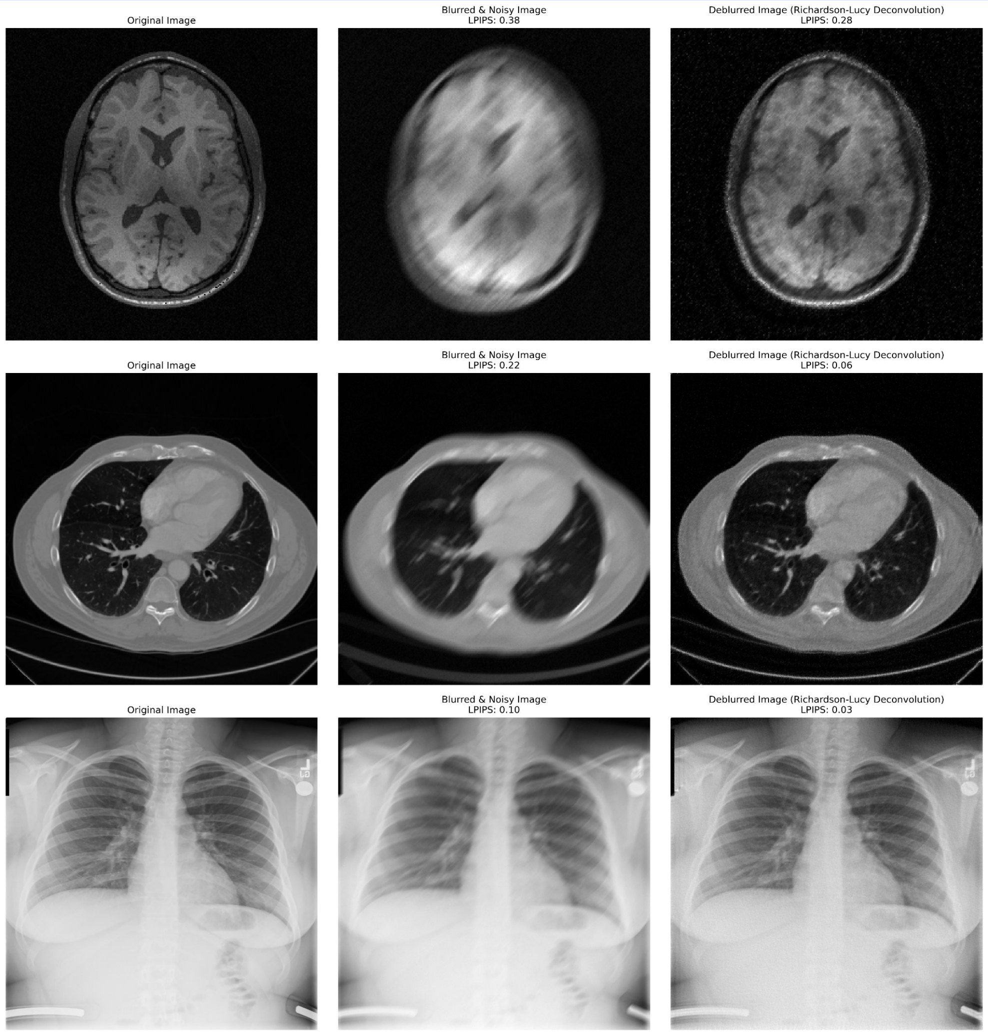
Brain MRI Image

Chest CT Image

Chest X-Ray Image

Richardson Lucy Deconvolution

Kernel Size: 20
Angle: 45
Noise Sigma: 3



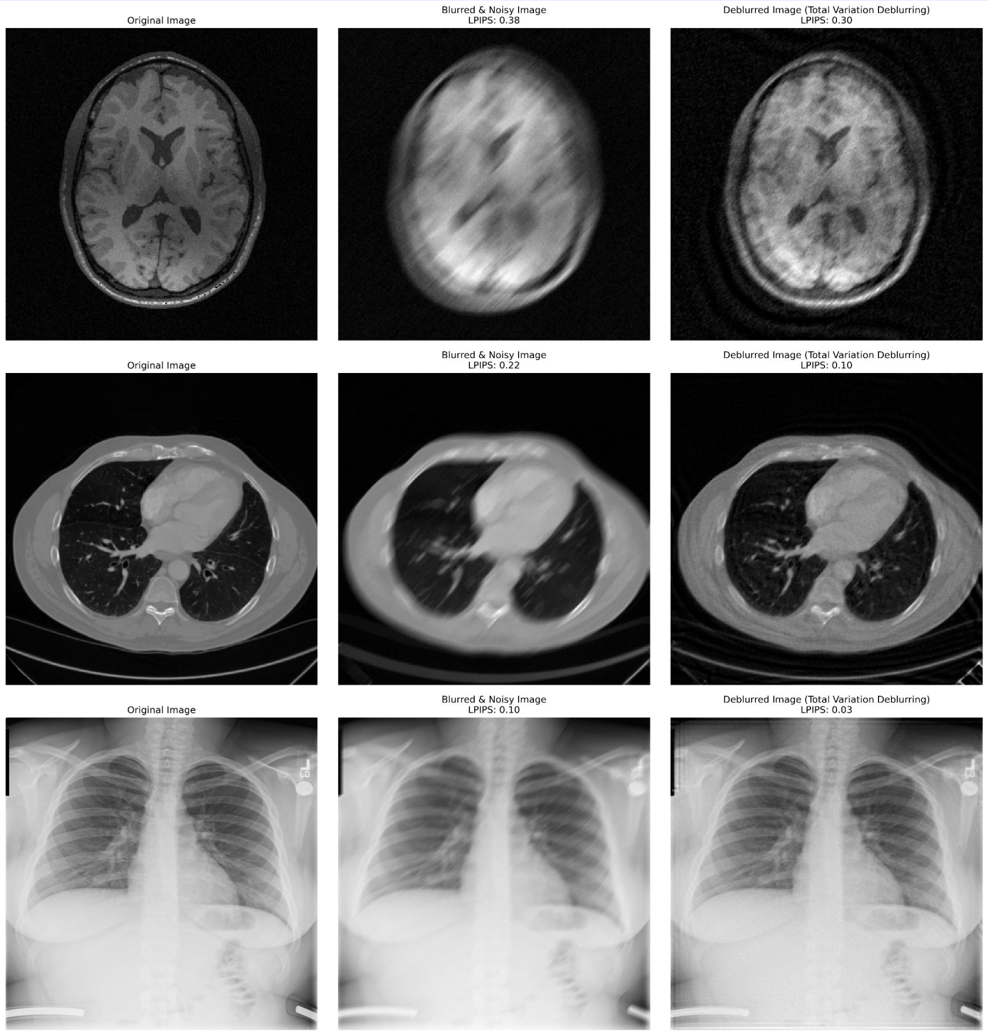
Brain MRI Image

Chest CT Image

Chest X-Ray Image

Total Variation Algorithm

Kernel Size: 20
Angle: 45
Noise Sigma: 3



Brain MRI Image

Chest CT Image

Chest X-Ray Image

Effect of Kernel Size on Deblurring

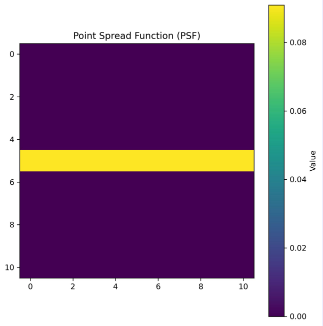
- Kernel sizes tested: k = 10, 20, 30
- Wiener Deconvolution
  - Performance drops with increasing kernel size
  - Susceptible to noise amplification due to weak frequency response
- Richardson-Lucy
  - Better with large kernels, but needs more iterations
  - More sensitive to noise as kernel size increases
- Total Variation
  - Most robust to large kernels
  - Preserves edge sharpness with slight smoothing trade-off

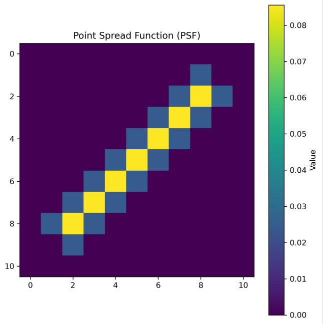
Effect of Noise Level on Deblurring

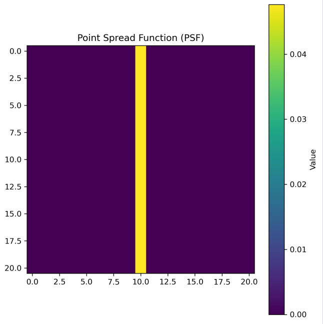
- Noise levels tested: $\sigma = 1, 3, 5$
- Wiener Deconvolution
  - Unstable with increasing noise (LPIPS: 0.06 to 0.36)
  - Amplifies noise, causing artifacts and detail loss.
- Richardson-Lucy
  - Amplifies noise over iterations (LPIPS: 0.10 to 0.32)
  - Requires early stopping or denoising, especially at $\sigma = 5$.
- Total Variation
  - Best in high-noise settings (LPIPS: 0.12 to 0.29)
  - Preserves structure with slight smoothing at $\sigma = 5$.

Effect of PSF Orientation on Deblurring

- PSF orientations tested: α = 0°, 45°, 90°
- Wiener Deconvolution
  - Largely isotropic in frequency domain
  - Minor sensitivity due to numerical artifacts
- Richardson-Lucy
  - Performs well across orientations
  - May introduce directional artifacts at higher angles
- Total Variation
  - Consistent across directions
  - Slight directional bias but more isotropic than RL







Observations

Visual Quality & Restoration

Kernel Size: 20, Angle: 45, Noise Sigma: 3

- Wiener Deconvolution
    - Partial blur reduction but limited detail recovery
    - Ringing artifacts and background distortions visible
    - LPIPS: 0.30 – moderate perceptual improvement
- Richardson-Lucy Deconvolution
    - Sharpest output with clear anatomical structures
    - Amplifies noise in flat/background regions
    - LPIPS: 0.28 – lowest, highest perceptual similarity
- Total Variation Deblurring
    - Best balance between noise suppression and edge preservation
    - Smooths noise while maintaining structure
    - Slight over smoothing in low-contrast areas
    - LPIPS: 0.30 – matches Wiener, but better qualitative sharpness

Computation Time & Efficiency

- Total Variation Deblurring
  - Slowest method due to iterative gradient descent and regularization
  - Requires more computation for convergence
- Richardson-Lucy Deconvolution
  - Moderate runtime
  - Iterative but reasonably efficient for a limited number of iterations
- Wiener Deconvolution
  - Fastest among all methods
  - Single-pass frequency-domain filtering enables quick processing

Overall Trade-offs

- Wiener Deconvolution
  - Best for speed and noise suppression
  - Suitable for real-time or low-resource scenarios
- Richardson-Lucy Deconvolution
  - Ideal for enhancing fine details
  - Trades off with higher noise sensitivity
- Total Variation Deblurring
  - Best balance for medical imaging
  - Combines edge preservation with noise reduction
  - Acceptable runtime for high-quality restoration

A detailed report along with all the results can be found at the link below:

https://github.com/AadishSethiya/CS-736-Project.git

Thank You!