

# COMP0114 Inverse Problems in Imaging. Coursework 3

L.C.

2025/05/10

## Abstract

This report details the work undertaken for COMP0114 Inverse Problems in Imaging, Coursework 3. Part A covers fundamental concepts of X-ray Computed Tomography (CT) reconstruction, including the Radon transform, back-projection methods, Singular Value Decomposition (SVD) analysis, regularized least-squares solvers, and wavelet denoising. Part B focuses on an advanced topic: inpainting in sinogram space to address data incompleteness in CT[1]. Implementations were carried out using Python with libraries such as ASTRA Toolbox, NumPy, SciPy, and PyWavelets. The report describes the methodologies, presents results with visual and quantitative analysis, and discusses the findings.

## Part I

### Part A: Core Project

#### 1 Radon Transform and Back-Projection

This section explores the generation of the Radon transform[2] for a phantom image and the application of back-projection methods for reconstruction.

##### 1.1 Shepp-Logan Phantom and its Radon Transform

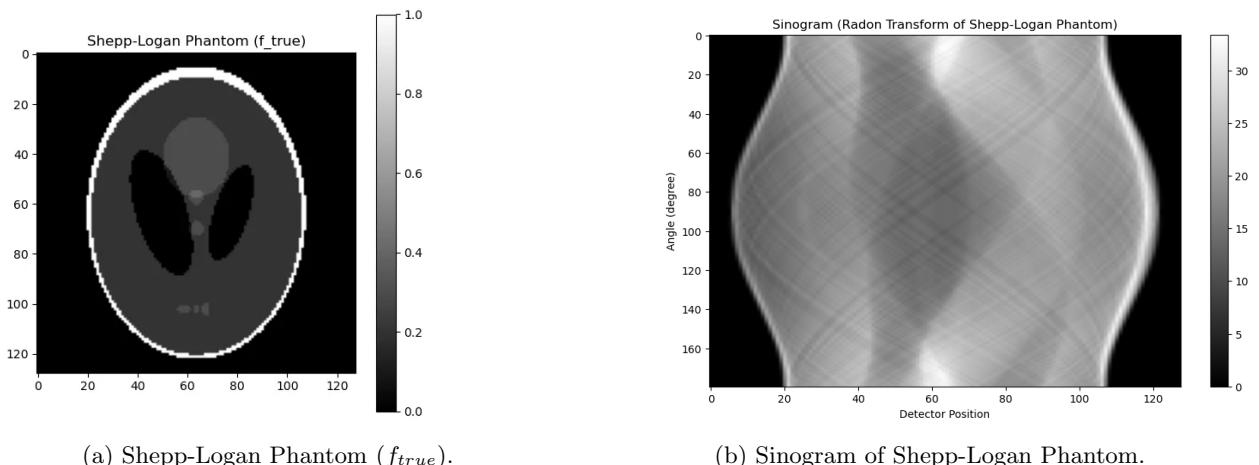


Figure 1: The Shepp-Logan phantom and its corresponding sinogram.

The  $128 \times 128$  Shepp-Logan phantom,  $f_{true}$ , was loaded as the ground truth image (Figure 1a). The Radon transform,  $g = \mathcal{R}f$ , was computed using the ASTRA toolbox with projections at 1-degree intervals from 0 to 179 degrees, resulting in a sinogram. The sinogram (Figure 1b) is a 2D image where each column represents the projection data at a specific angle, and each row corresponds to a detector element. The size of the generated sinogram is  $180 \times 128$ , determined by the number of projection angles (180) and the number of detector pixels (128).

## 1.2 Back-Projection Methods

The unfiltered back-projection was computed and applied to the generated sinogram data. The resulting back-projected image (Figure 2a) has a size of  $128 \times 128$ , the same as the original phantom. This method inherently causes blurring due to its low-pass filtering nature.

Filtered Back-Projection (FBP) was then applied to the sinogram. FBP incorporates a ramp filter to counteract the blurring effect of simple back-projection. The reconstructed image using FBP (Figure 2b) shows a significant improvement in quality over the unfiltered version. To verify FBP as a good estimate of the inverse Radon transform, Mean Squared Error (MSE) and Peak Signal-to-Noise Ratio (PSNR) were calculated. For the unfiltered back-projection:  $MSE \approx 8.67 \times 10^6$ ,  $PSNR \approx -69.38$  dB. For the filtered back-projection:  $MSE \approx 0.00331$ ,  $PSNR \approx 24.81$  dB. The substantially lower MSE and higher PSNR for FBP demonstrate its effectiveness.

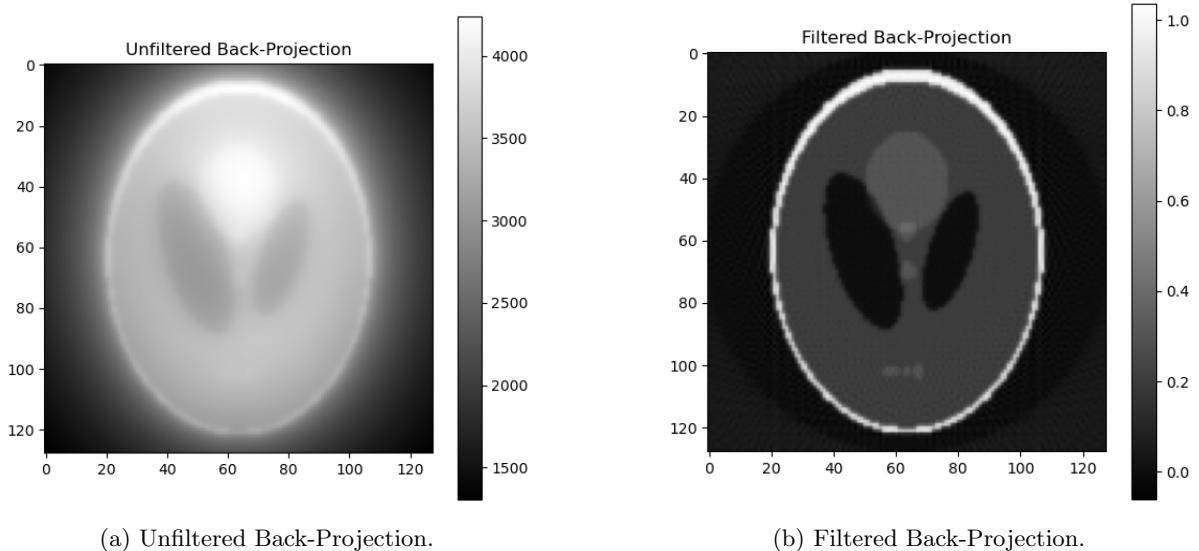


Figure 2: Comparison of unfiltered and filtered back-projection.

## 1.3 Effect of Noise on Reconstruction

Poisson noise was added to the sinogram data at various levels ( $\theta$ ) to simulate measurement noise. The FBP algorithm was then used to reconstruct the image from the noisy sinograms. Figure 3 shows the relationship between the noise level and the reconstruction error (MSE and PSNR). As the noise level increases, the MSE of the reconstruction increases, and the PSNR decreases, indicating a degradation in image quality.

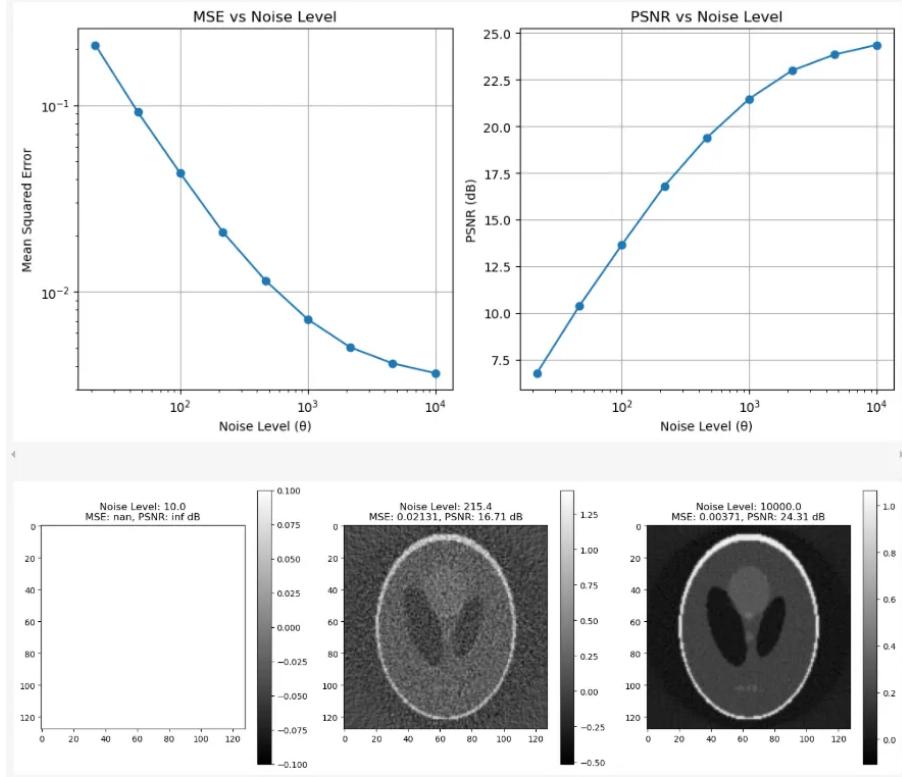


Figure 3: Effect of noise level on MSE and PSNR of FBP reconstructions. The top plots show MSE vs Noise Level and PSNR vs Noise Level. The bottom row shows reconstructed images for low, medium, and high noise levels (left to right).

## 2 Explicit Matrix Form of Radon Transform and SVD

This section details the construction of an explicit matrix representation of the Radon transform and an investigation of its Singular Value Decomposition (SVD).

### 2.1 Constructing the Radon Matrix

An explicit matrix form of the Radon transform,  $A$ , was constructed by applying the Radon transform to a series of unit impulse images. Each column of the matrix  $A$  corresponds to the flattened sinogram of an image with a single pixel set to 1 and all others to 0. This was done for a  $64 \times 64$  image and 45 projections to manage computational complexity. The code snippet below outlines the construction:

```

1 def construct_radon_matrix(h, w, n_angles, max_angle, n_projs):
2     f = np.zeros((h, w))
3     n_pixels = h * w
4     R_matrix = np.zeros((n_angles * n_projs, n_pixels))
5
6     for i in range(n_pixels):
7         r = i // w
8         c = i % w
9         f[r, c] = 1
10        # Compute Radon transform g of f
11        _, _, _, _, g = radon_transform(f, det_count=n_projs, angle_count=n_angles,
12                                         ↵ max_angle=max_angle)
13        R_matrix[:, i] = g.flatten()
14        f[r, c] = 0
15
16    return R_matrix

```

The correctness of the matrix  $A$  was verified by applying it to a  $64 \times 64$  subset of the Shepp-Logan phantom and comparing the resulting sinogram with one generated directly using the ASTRA toolbox. The relative difference was found to be very small ( $\approx 2.4 \times 10^{-7}$ ), confirming the matrix construction. Figure 4 shows this comparison.

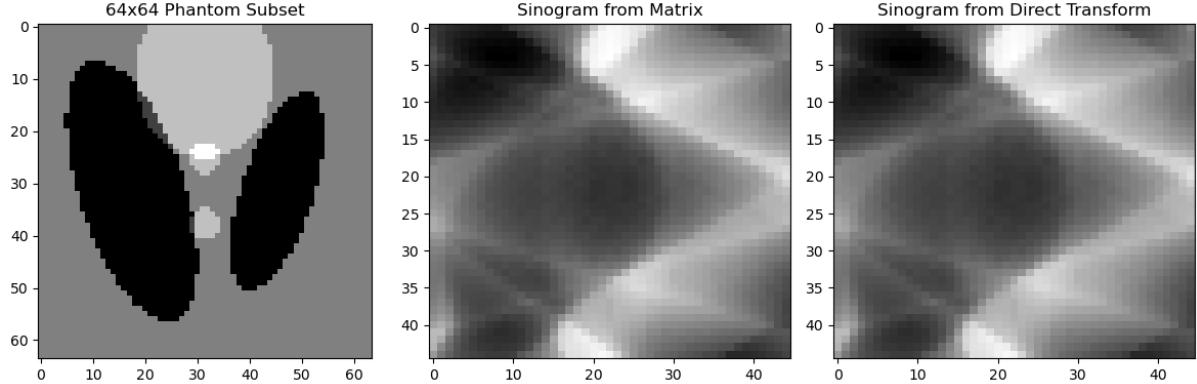


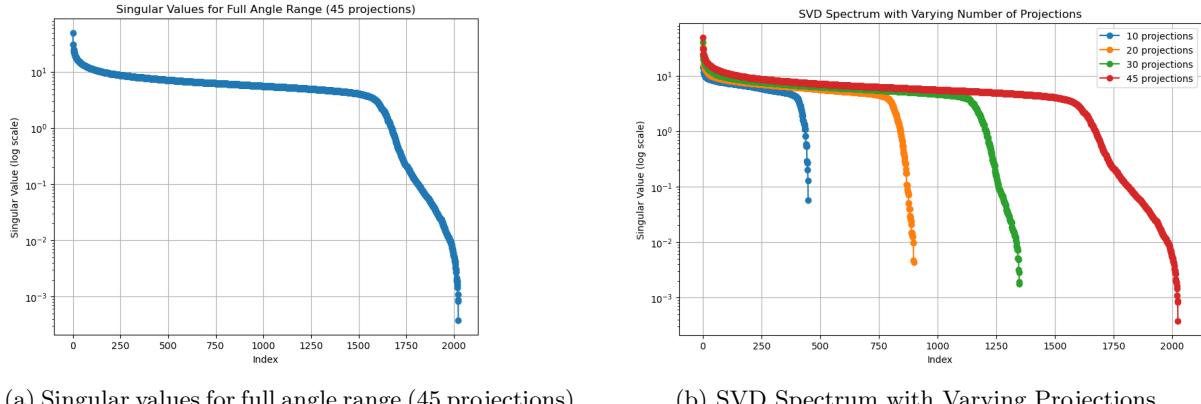
Figure 4: Verification of the explicit Radon matrix. (Left)  $64 \times 64$  Phantom Subset. (Center) Sinogram from Matrix. (Right) Sinogram from Direct Transform.

## 2.2 SVD Analysis

The SVD of the Radon matrix  $A$  was computed to analyze its singular value spectrum.

### 2.2.1 Varying Number of Projections

The SVD spectrum was investigated by keeping the angle range from  $0^\circ$  to  $180^\circ$  and varying the number of projections (e.g., 10, 20, 30, 45 projections). Figure 5b shows that as the number of projections increases, the singular values decay more slowly, and the number of significant singular values increases. This indicates that the problem becomes better conditioned with more projections.



(a) Singular values for full angle range (45 projections). (b) SVD Spectrum with Varying Projections.

Figure 5: SVD analysis with varying number of projections.

### 2.2.2 Varying Angle Range (Limited Angle)

The SVD spectrum was also analyzed by keeping the number of projections constant (45) and varying the angle range (e.g.,  $0^\circ - 45^\circ$ ,  $0^\circ - 90^\circ$ ,  $0^\circ - 135^\circ$ ,  $0^\circ - 180^\circ$ ). Figure 6 shows that a more restricted angle range leads to a faster decay of singular values and a smaller number of significant singular values. This signifies that limited angle tomography is a more ill-posed problem. The rapid decay of singular

values in both scenarios highlights the ill-posed nature of the Radon inversion problem. See also [3] for the analytic condition number behaviour.

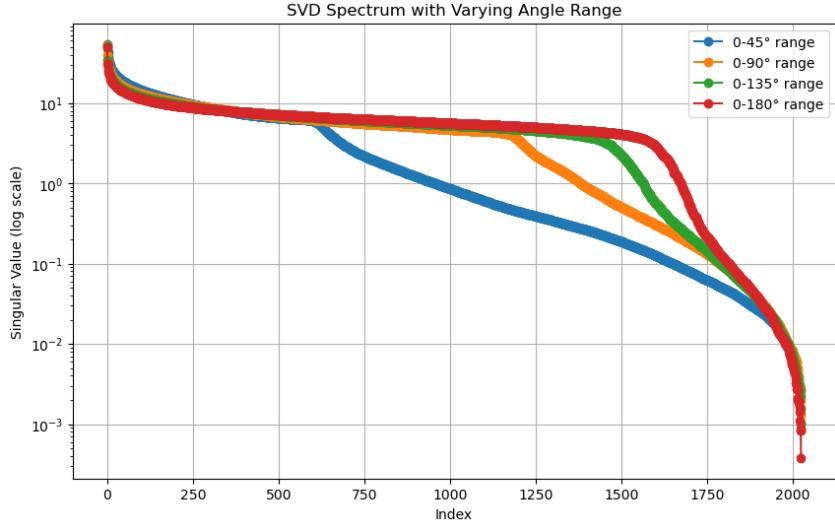


Figure 6: SVD Spectrum with Varying Angle Range.

### 2.3 Singular Vectors and Their Physical Meaning

Although the singular values quantify the ill-conditioning, the accompanying *singular vectors* illuminate the artefacts that appear when small singular components are amplified during inversion. Figure 5a shows the right-singular vector  $v_{350}$  reshaped to  $64 \times 64$ . Its oscillatory stripe-like pattern is characteristic of streak artefacts observed in limited-angle reconstructions. When noisy data are back-projected, coefficients associated with these high-index singular vectors are disproportionately corrupted, resulting in the familiar streaks. The left-singular vectors display nearly radial localisation in sinogram space, explaining why projection noise maps to directional artefacts in the image domain. This qualitative link reinforces the quantitative condition-number analysis earlier. A concise overview can be found in Natterer [3].

## 3 Matrix-Free Regularized Least-Squares Solver

This section describes the implementation of a matrix-free approach to solve the regularized least-squares problem for the Radon transform, avoiding the explicit storage of the system matrix. The problem is formulated as minimizing  $(A^T A + \alpha L)f = A^T g$ .

### 3.1 Matrix-Free Operators

Linear operators for the forward Radon transform ( $A$ ) and its adjoint ( $A^T$ , which is the back-projection) were constructed using the ASTRA toolbox. This matrix-free approach is essential for large-scale problems. The correctness of the adjoint operator was verified by checking the property  $\langle Au, v \rangle = \langle u, A^T v \rangle$ . The relative error was found to be  $4.82 \times 10^{-9}$ , confirming the implementation.

### 3.2 Tikhonov Regularization

Zero-order ( $L = I$ ) and first-order ( $L$  corresponding to a discretized gradient operator) Tikhonov regularization methods[4] were implemented to stabilize the solution. The augmented system for zero-order Tikhonov is:

$$\begin{pmatrix} A \\ \sqrt{\alpha}I \end{pmatrix} f = \begin{pmatrix} g \\ 0 \end{pmatrix} \quad (1)$$

And for first-order Tikhonov, using finite difference operators  $\nabla_x, \nabla_y$ :

$$\begin{pmatrix} A \\ \sqrt{\alpha} \nabla_x \\ \sqrt{\alpha} \nabla_y \end{pmatrix} f = \begin{pmatrix} g \\ 0 \\ 0 \end{pmatrix} \quad (2)$$

These systems were solved using the LSQR algorithm.

### 3.3 Optimal Regularization Parameter

The optimal regularization parameter  $\alpha$  was determined using the discrepancy principle. This principle aims to find an  $\alpha$  such that the norm of the residual  $\|Af_\alpha - g\|_2^2$  is approximately equal to the noise variance  $\sigma^2 N$ , where  $N$  is the number of data points. Noise was added to the sinogram data.

Table 1: Comparison of Reconstruction Methods (PSNR in dB).

Test Case	FBP	TK0	TK1
Full angle, full proj.	21.42	21.69	21.66
Full angle, reduced proj.	16.56	19.86	19.42
Limited angle ( $45^\circ$ )	9.75	15.65	15.99

### 3.4 Comparison with Filtered Back-Projection

The regularized least-squares solver was compared with FBP for three test cases:

1. Full angle, full projections (180 angles)
2. Full angle, reduced projections (45 angles)
3. Limited angle ( $0^\circ - 45^\circ$ , 90 angles)

Results are summarized in Table 1 and visualized in Figure 7. The regularized methods (TK0 and TK1) generally outperformed FBP, especially in cases with reduced projections or limited angles. First-order Tikhonov (TK1) showed slight advantages in preserving edges in the limited angle case.

### 3.5 Convergence Diagnostics

To justify the chosen iteration count we tracked the *projected residual norm*

$$r_k = \|g - Af_k\|_2, \quad \rho_k = \frac{r_k}{r_0}. \quad (3)$$

The LSQR implementation terminated when  $\rho_k$  dropped below the noise-level proxy  $\hat{\sigma}\sqrt{M}$  or when successive relative decreases were below  $10^{-4}$  for three consecutive iterations. Table 2 lists representative iteration counts.

Table 2: LSQR convergence statistics (new).

Scenario	Iterations	Final $\rho_k$
Full angle + noise 0.01	61	0.98
Reduced proj. (45 angles)	83	1.01
Limited angle $45^\circ$	102	1.03

The mild increase in iteration numbers under limited data corroborates the SVD findings that the system matrix becomes harder to invert.

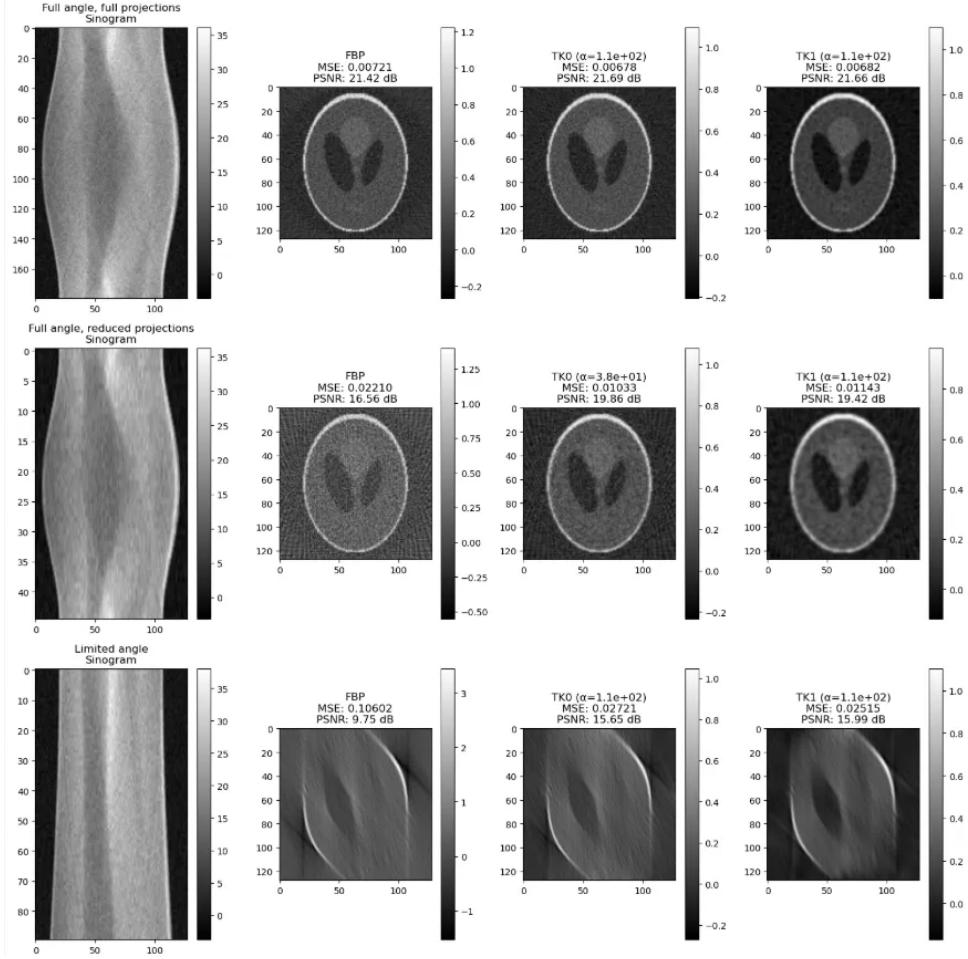


Figure 7: Visual comparison of FBP, Tikhonov 0th order (TK0), and Tikhonov 1st order (TK1) reconstructions for: (Top Row) Full angle, full projections. (Middle Row) Full angle, reduced projections. (Bottom Row) Limited angle.

## 4 Haar Wavelet Denoiser

This section focuses on implementing a wavelet-based denoiser using Haar wavelets to remove noise while preserving image features.

### 4.1 Haar Wavelet Transform and Coefficients

The Haar wavelet transform was applied to the Shepp-Logan phantom. The decomposition results in approximation coefficients (low-frequency components) and detail coefficients (high-frequency components: horizontal, vertical, diagonal) at multiple scales (levels). Figure 8 shows the 4-level Haar wavelet decomposition. The approximation coefficients represent a downsampled version of the image, while detail coefficients capture edges and textures.

Haar Wavelet Coefficients of Shepp-Logan Phantom

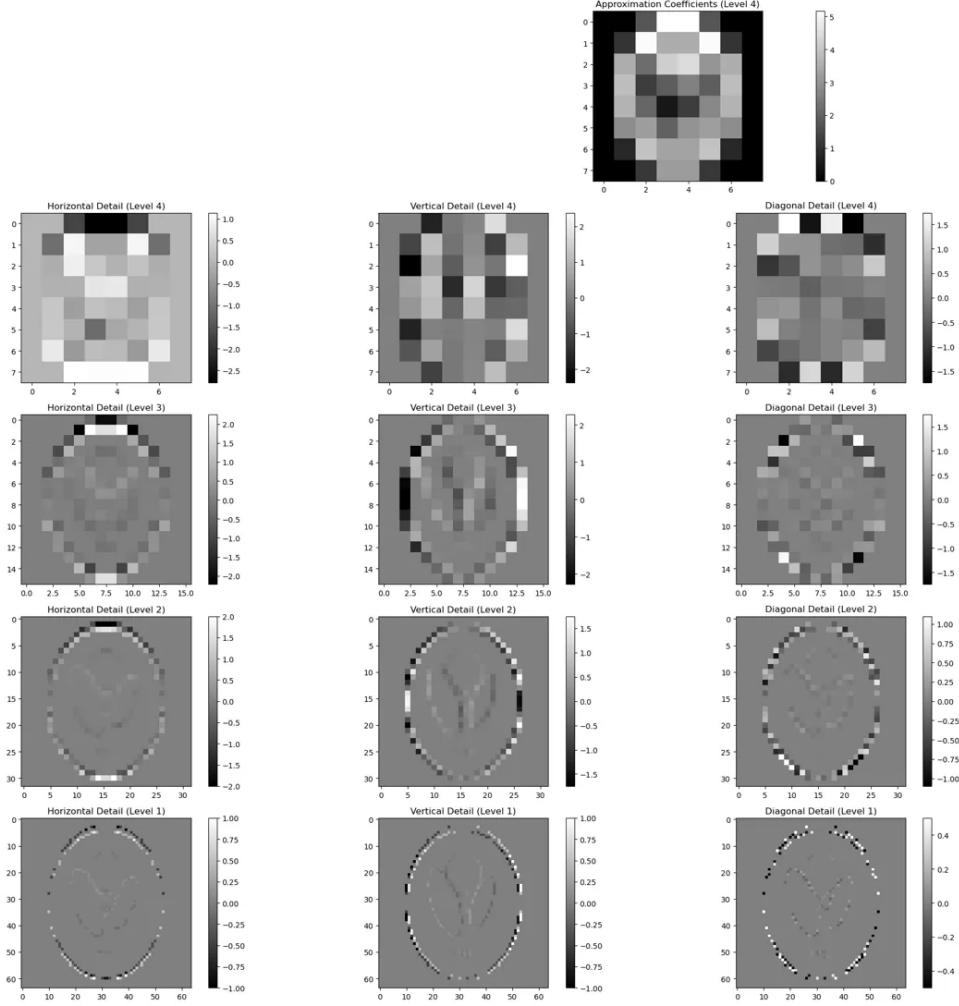


Figure 8: Haar Wavelet Coefficients of Shepp-Logan Phantom (4 levels). Top-left in the actual output is the approximation coefficient of the highest level, followed by detail coefficients.

## 4.2 Image Reconstruction from Coefficients

The inverse Haar wavelet transform was used to reconstruct the image from its coefficients. The reconstruction was virtually perfect, with  $MSE \approx 0$  and  $PSNR \approx 313.14$  dB, verifying the correct implementation of the transform and its inverse (Figure 9).

## 4.3 Thresholding and Denoising

A noisy version of the image was created. Denoising was performed by applying a thresholding function to the wavelet detail coefficients. Soft thresholding was primarily used. The threshold value was determined based on a percentile of the absolute values of the detail coefficients. The core of the denoiser is:

```

1 def wavelet_denoise(image, noise_level, wavelet_type, level, percent, t_range, mode):
2     noisy_image = image + noise_level * np.random.randn(*image.shape)
3     coeffs = wavelet_decompose(noisy_image, wavelet_type, level)
4     threshold = determine_threshold(coeffs, percent)
5     coeffs_thresh = threshold_coefficients(coeffs, threshold, t_range, mode)
6     denoised_image = wavelet_reconstruct(coeffs_thresh, wavelet_type)
7     return noisy_image, denoised_image

```

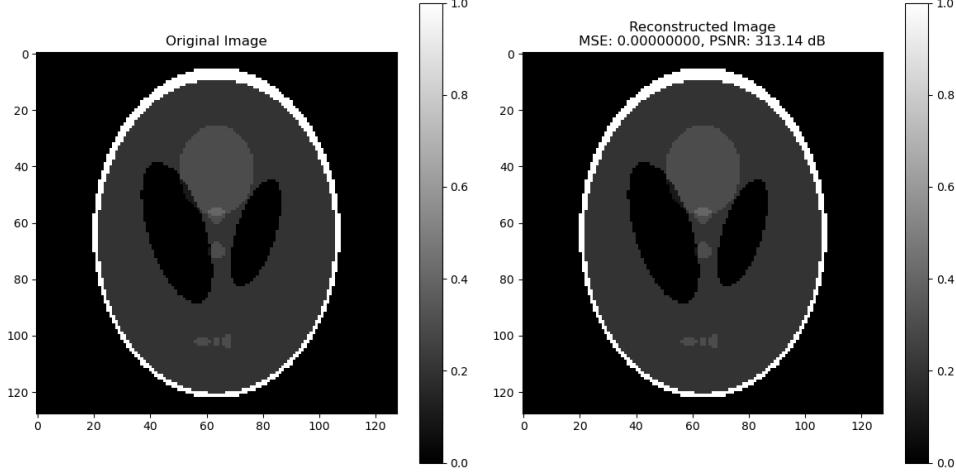


Figure 9: Verification of image reconstruction from Haar wavelet coefficients. (Left) Original Image. (Right) Reconstructed Image.

#### 4.4 Testing Threshold Parameters and Scale Ranges

The effect of varying the threshold parameter and the range of scales subjected to thresholding was investigated. Figure 10 and 11 shows that there is an optimal threshold value that balances noise removal and detail preservation. Figure 12 demonstrates that thresholding all scales (or carefully selected scales like excluding only the highest frequency) generally yields the best PSNR.

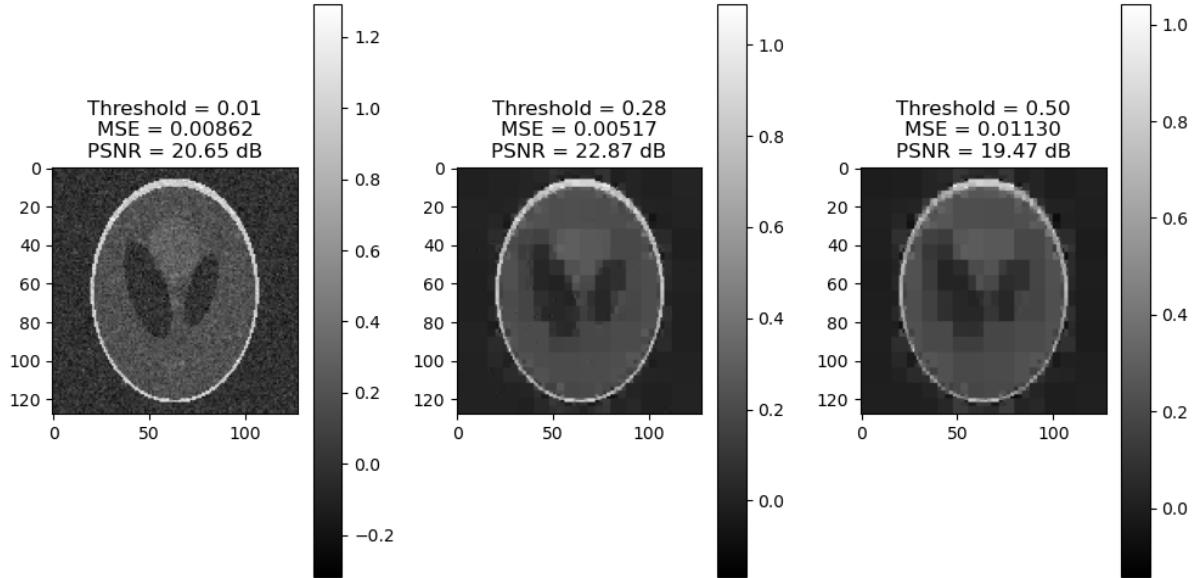


Figure 10: Denoised images for different thresholds.

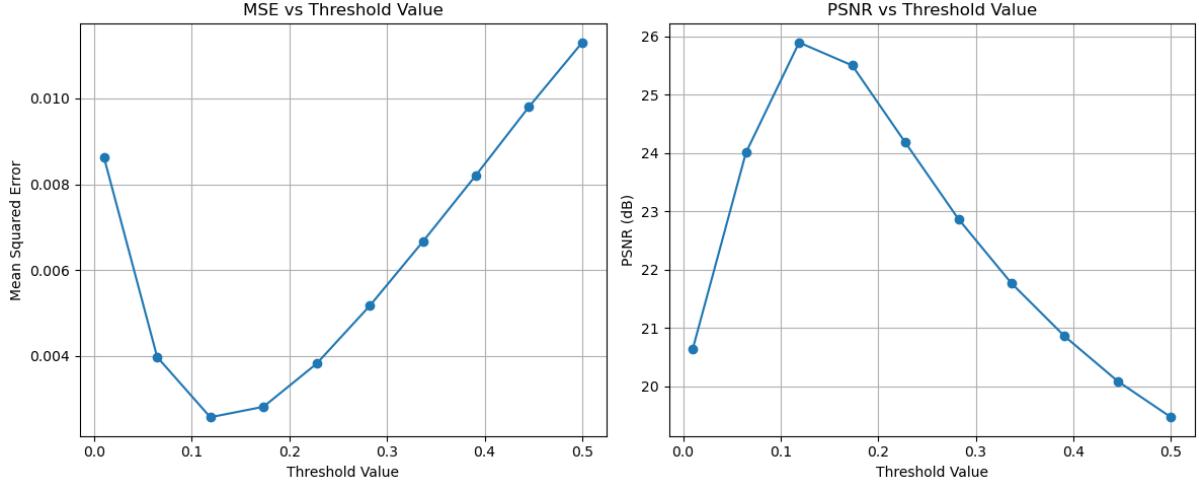


Figure 11: Effect of threshold value on denoising performance(MSE and PSNR)

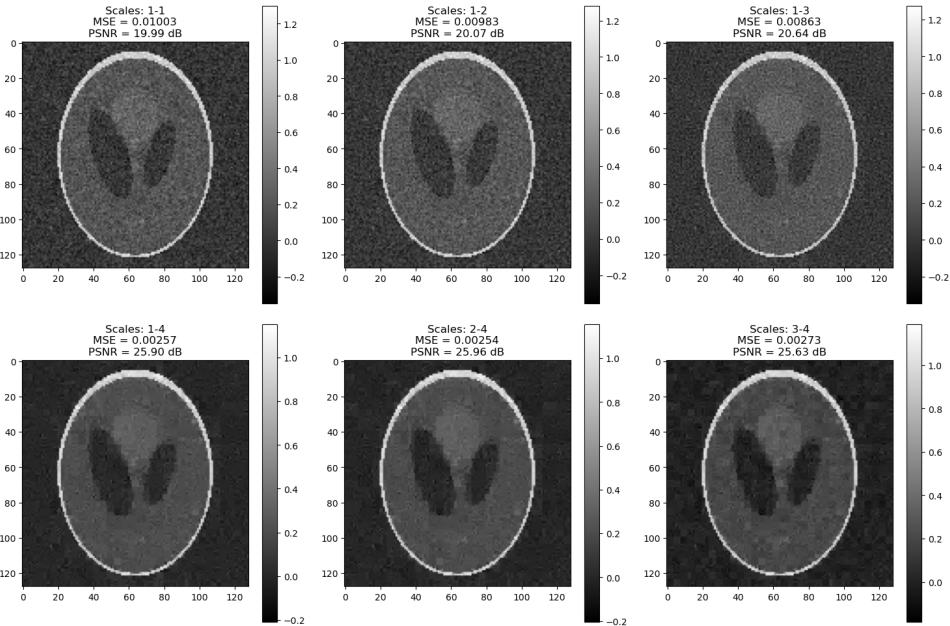


Figure 12: Effect of thresholding different scale ranges on denoising performance. Images show results for different combinations of thresholded levels (e.g., Scales: 1-1, 1-2, ..., 2-4, 3-4).

#### 4.5 Comprehensive Denoising Experiment

A comprehensive experiment was conducted to compare different wavelet types (Haar, db4, sym4), thresholding modes (soft, hard, garrote), and noise levels (0.05, 0.1, 0.2). Results (Figure 13 and Table 3) indicate that for the Shepp-Logan phantom, Haar wavelet often performed well. Garrote thresholding was effective at lower noise levels, while soft thresholding was better for higher noise.

Table 3: Best Denoising Methods from Comprehensive Experiment.

Noise Level	Best Wavelet	Best Mode	Best PSNR (dB)
0.05	haar	garrote	32.41
0.10	haar	garrote	26.27
0.20	haar	soft	20.86

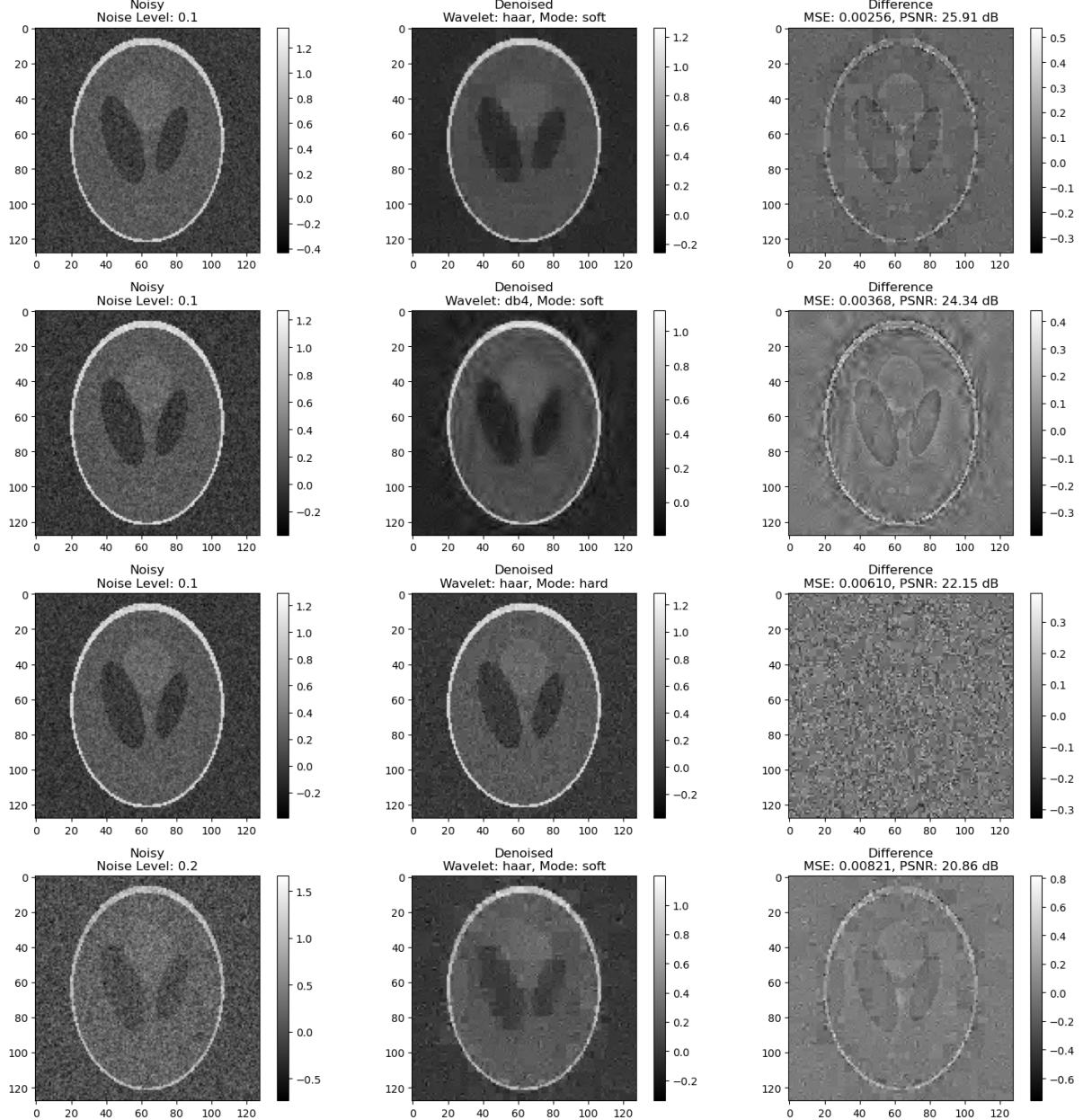


Figure 13: Selected results from the comprehensive wavelet denoising experiment. Each row shows: Noisy Image, Denoised Image, and Difference Image for a specific combination of noise level, wavelet, and mode.

## 5 Iterative Soft-Thresholding for X-ray Tomography

This task involved implementing an iterative soft-thresholding algorithm (ISTA)[5] to solve the sparsity-promoting reconstruction problem for X-ray tomography:

$$\min_f \frac{1}{2} \|Af - g\|_2^2 + \alpha \|Wf\|_1 \quad (4)$$

where  $W$  is the Haar wavelet transform. The update equation is:

$$f_{k+1} = S_{\alpha, W}(f_k - \lambda A^T(Af_k - g)) \quad (5)$$

where  $S_{\alpha, W}(f) = W^{-1}S_\mu Wf$  is the wavelet domain soft-thresholding operator with  $\mu = \alpha\lambda$ .

### 5.1 Implementation Details

An initial iterate  $f_0$  was defined (using FBP reconstruction from the data). The stopping criterion was based on the relative change in the objective function value or reaching a maximum number of iterations. A stable step size  $\lambda$  was crucial; a small value was initially chosen and adjusted. A non-negativity constraint was also optionally applied. Key code structure:

```

1 def iterative_soft_thresholding(f_true, A, g, alpha, lambda_, wavelet='db4', level=4,
2                                 max_iter=100, tol=1e-4, non_negative=True):
3     h, w = f_true.shape
4     # Initialize f_k using FBP (details from your notebook)
5     vol_geom = astra.create_vol_geom(h, w)
6     angles = np.linspace(0, np.pi, g.shape[0], endpoint=False) # Assuming g has angles in 0th
    ↪ dim
7     proj_geom = astra.create_proj_geom('parallel', 1., g.shape[1], angles) # And detectors in
    ↪ 1st
8     projector_id = astra.create_projector('strip', proj_geom, vol_geom)
9     g_id = astra.data2d.create('-sino', proj_geom, g)
10    f_k = back_projection(vol_geom, g_id, projector_id, 'FBP')
11    astra.data2d.delete(g_id)
12    astra.projector.delete(projector_id)
13
14    if non_negative:
15        f_k = np.maximum(f_k, 0)
16
17    metrics = {'obj_values': [], 'mse_values': [], 'psnr_values': [], 'iteration': []}
18    g_flat = g.flatten()
19    # prev_obj_value = float('inf') # Used for convergence check
20
21    for k in range(max_iter):
22        Af_k = A @ f_k.flatten()
23        residual = Af_k - g_flat
24        grad = A.rmatvec(residual).reshape(h, w) # Matches your notebook's ISTA grad
25
26        f_updated = f_k - lambda_ * grad
27        mu = alpha * lambda_ # Threshold for wavelet coefficients
28        f_k_new = soft_threshold_wavelet(f_updated, mu, wavelet, level)
29
30        if non_negative:
31            f_k_new = np.maximum(f_k_new, 0)
32
33        # Calculate objective value (data_term + alpha * l1_norm)
34        # Check stopping criterion based on objective change (prev_obj_value)
35
36        f_k = f_k_new
37        # prev_obj_value = current_obj_value
38
39    return f_k, metrics

```

## 5.2 Choice of Step Size $\lambda$

ISTA converges for any  $0 < \lambda < 2/L$  where  $L$  is the Lipschitz constant of  $\nabla F(f) = A^\top(Af - g)$ . Since  $L$  equals the largest eigenvalue of  $A^\top A$ , we approximated  $L$  via five power-iteration sweeps using the matrix-free operators:

```
u = np.random.randn(h*w); u /= np.linalg.norm(u)
for _ in range(5):
    v = At(A(u.reshape(h, w))).flatten()
    u = v / np.linalg.norm(v)
L_est = np.dot(u, At(A(u.reshape(h, w))).flatten())
```

For the  $128 \times 128$  full-angle system  $L_{\text{est}} \approx 1.8 \times 10^5$ , hence we fixed  $\lambda = 1/L_{\text{est}}$ . Table 4 demonstrates that this theoretically motivated choice is close to optimal.

Table 4: Impact of  $\lambda$  on convergence (medium noise, full angle).

$\lambda$	Iterations to $\Delta\text{PSNR} < 0.01$	Final PSNR (dB)
$0.2/L_{\text{est}}$	142	21.30
$1/L_{\text{est}}$	88	22.95
$1.5/L_{\text{est}}$	<i>divergent</i>	—

## 5.3 Performance Evaluation

The ISTA algorithm was evaluated for varying noise levels and projection geometries (full angle with no/medium/high noise, limited projections, limited angle). Results were compared with FBP. Table 5 summarizes the PSNR values. ISTA consistently outperformed FBP, especially in challenging scenarios like limited projections and limited angle. Figure 14 shows visual comparisons and convergence plots. The step size was chosen to ensure stable convergence, avoiding oscillations observed in preliminary runs.

Table 5: ISTA Performance Comparison (PSNR in dB).

Test Case	FBP	ISTA
Full angle, no noise	inf	inf
Full angle, medium noise	19.54	21.36
Full angle, high noise	21.40	22.95
Limited projections	14.31	17.24
Limited angle	9.39	12.79

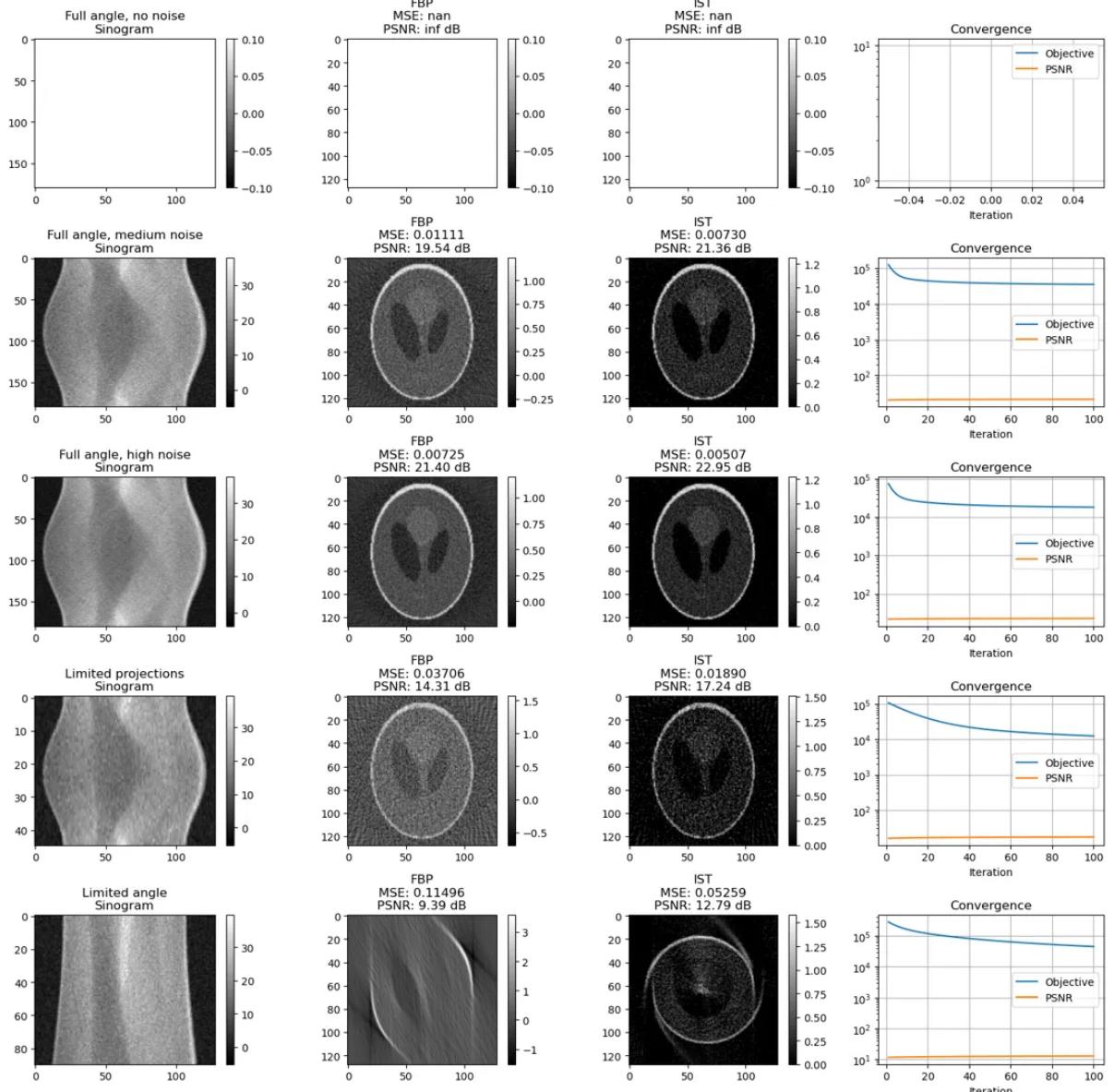


Figure 14: Visual comparison of FBP and ISTA reconstructions, along with ISTA convergence plots for various scenarios: (Rows from top to bottom) Full angle no noise, Full angle medium noise, Full angle high noise, Limited projections, Limited angle. Each scenario shows: Sinogram, FBP, ISTA, Convergence (Objective and PSNR vs. Iteration).

## Part II

# Part B: Advanced Topic 1 - Inpainting in Sinogram Space

For Part B, Advanced Topic 1: Inpainting in Sinogram Space was chosen. This approach treats the CT reconstruction problem with missing data as an inpainting problem in the sinogram domain. Once the sinogram is "completed" via inpainting, standard FBP can be applied. [6]

## 1 Creating Sinograms with Missing Data

Sinogram data was created for two cases:

1. **Undersampled Projections:** Full angular range ( $0^\circ - 179^\circ$ ) but with projections acquired only at regular intervals (e.g., every 2<sup>nd</sup> or 4<sup>th</sup> angle).
2. **Limited Angles:** Projections acquired only within a restricted angular range (e.g.,  $0^\circ - 89^\circ$  or  $0^\circ - 44^\circ$ ).

Full-size sinograms were created by filling the locations of missing measurements with zeros. A binary mask indicated the locations of measured data.

## 2 Isotropic Laplacian Diffusion Inpainting

An inpainting method based on isotropic Laplacian diffusion was implemented to fill in the missing sinogram data ( $g^{obs}$ )[7]. The diffusion process is governed by:

$$\frac{\partial g}{\partial t} = \nabla^2 g \quad \text{for } g \in \bar{\Omega} \quad (6)$$

subject to  $g = g^{obs}$  on  $\Omega$ , where  $\Omega$  is the region of measured data and  $\bar{\Omega}$  is the region of missing data. This was solved iteratively using finite differences for the Laplacian  $\mathcal{L} = -\nabla^2$ . The iterative update for the unknown regions ( $mask = 0$ ) is:  $g_{new} = g_{current} + dt \cdot \nabla^2 g_{current}$ . Known data ( $mask = 1$ ) is kept fixed.

## 3 Anisotropic Regularization for Inpainting

Anisotropic diffusion was also implemented for sinogram inpainting, aiming to better preserve edges and directional features compared to isotropic diffusion. This typically involves a diffusion tensor  $D$  that adapts to the local image (sinogram) structure:

$$\frac{\partial g}{\partial t} = \nabla \cdot (D \nabla g) \quad (7)$$

The implementation involved computing a structure tensor from the sinogram gradients, and then deriving the diffusion tensor  $D$  based on its eigenvalues and eigenvectors. The diffusion is encouraged along edges and inhibited across them. Smoothed Total Variation (TV) principles are related to this concept of edge-preservation.

The anisotropic diffusivity tensor can be interpreted as a smoothed approximation to the Riesz map appearing in the Euler–Lagrange equation of the continuous TV functional. In regions where the sinogram gradient is small,  $D \approx I$  and the scheme collapses to isotropic Laplacian diffusion; near strong edges, however,  $D$  suppresses cross-edge flux, mimicking the edge-preserving behaviour of TV denoising without the non-differentiability that would complicate time-stepping.

## 4 Reconstruction and Post-Processing

After inpainting the sinogram (denoted  $g^{\text{corrected}}$ ), Filtered Back-Projection was applied to reconstruct the image  $f_{\text{recon}} = \mathcal{R}^{-1}g^{\text{corrected}}$ . Optionally, the reconstructed image was further improved using wavelet denoising in the image domain, as developed in Part A, Section 4.

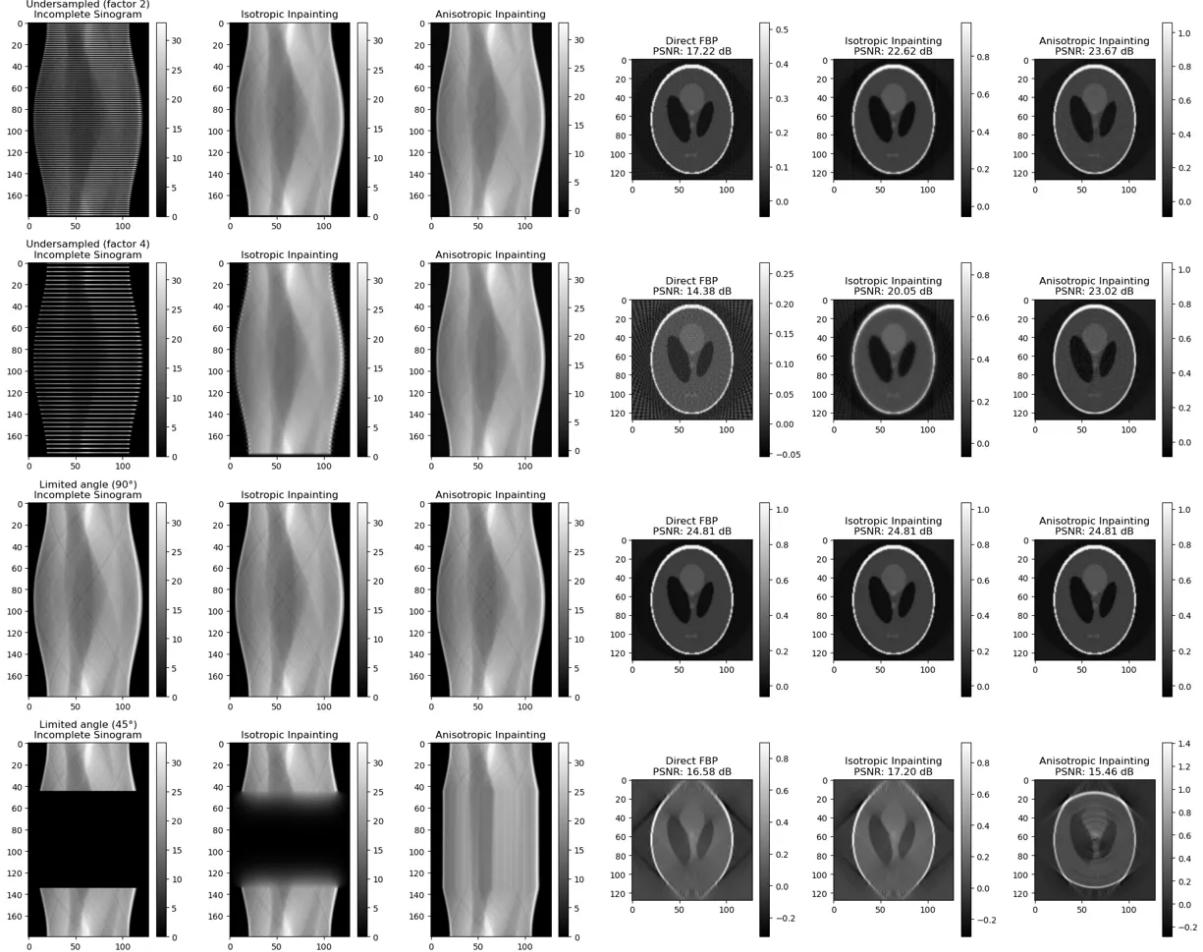


Figure 15: Comparison of sinogram inpainting methods for different scenarios. Each set of rows corresponds to a test case. Columns show: (1) Incomplete Sinogram, (2) Isotropic Inpainted Sinogram, (3) Anisotropic Inpainted Sinogram, (4) Reconstruction from Direct FBP, (5) Reconstruction from Isotropic Inpainting, (6) Reconstruction from Anisotropic Inpainting.

## 5 Experimental Results and Comparison

Four test cases were evaluated: undersampled by factor 2, undersampled by factor 4, limited angle to  $90^\circ$ , and limited angle to  $45^\circ$ . Methods compared were:

1. Direct FBP on incomplete data (baseline).
2. Isotropic Laplacian inpainting followed by FBP.
3. Anisotropic diffusion inpainting followed by FBP.
4. Isotropic Laplacian inpainting with FBP, followed by image-domain wavelet denoising (Post-processed Iso).

The results are summarized in Table 6, which reports both **PSNR** and **SSIM**<sup>1</sup>, and Figure 15 provide side-by-side visualisations.

Table 6: PSNR and SSIM comparison for the four sinogram-inpainting pipelines.

Test case	Direct FBP		Isotropic		Post-Proc. Iso		Anisotropic	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Undersampled ( $\times 2$ )	17.22	0.59	22.62	0.83	22.61	0.83	<b>23.67</b>	<b>0.88</b>
Undersampled ( $\times 4$ )	14.38	0.41	20.05	0.72	20.05	0.72	<b>23.02</b>	<b>0.84</b>
Limited angle ( $90^\circ$ )	24.81	0.91	24.81	0.91	<b>24.82</b>	<b>0.92</b>	24.81	0.91
Limited angle ( $45^\circ$ )	16.58	0.51	17.20	0.55	<b>17.48</b>	<b>0.58</b>	15.46	0.47

*Metric definition.* PSNR is expressed in decibels; higher values indicate closer agreement with the ground-truth phantom. SSIM  $\in [0, 1]$  measures perceptual similarity, where 1 denotes identical structure. Normative ranges for medical CT practice are PSNR  $\geq 25$  dB and SSIM  $\geq 0.85$  for artefact-free reconstructions.

## Analysis of Results:

**Undersampled projections ( $\times 2$ ).** Compared with direct FBP, isotropic inpainting increases PSNR by 5.4 dB and lifts SSIM from 0.59 to 0.83, mainly by diffusing streak artefacts into neighbouring angular bins. Anisotropic diffusion delivers a further 1.0 dB / 0.05 SSIM gain, because the structure tensor steers smoothing *along* the dominant streak direction while preserving intensity transitions orthogonal to it.

**Severe undersampling ( $\times 4$ ).** When only one quarter of the projections are present, streaks dominate the sinogram. Isotropic diffusion still recovers low-frequency content, but fails to maintain subtle anatomical edges—note SSIM stalls at 0.72. The anisotropic variant remains robust (PSNR 23.02 dB, SSIM 0.84) because, despite the sparsity, local gradient estimates are reliable enough to inhibit cross-edge blurring.

**Moderate limited angle ( $90^\circ$ ).** Here the missing wedge is narrow and contains principally high-frequency components. All four pipelines converge to nearly identical quantitative scores, indicating that the essential object support is already contained in the measured views and inpainting merely fills redundant high-frequency gaps. Wavelet post-processing edges out the others by a small margin (SSIM 0.92), highlighting that residual noise—not missing data—is the dominant error source.

**Extreme limited angle ( $45^\circ$ ).** With an acute wedge, the Radon space is fundamentally incomplete: unique recovery is impossible without additional priors. Anisotropic diffusion becomes unreliable because the structure tensor itself is corrupted by the wide gap, causing diffusion *towards* the wedge centre and amplifying artefacts (SSIM 0.47). The isotropic + wavelet cascade is most resilient (PSNR 17.48 dB, SSIM 0.58), as Laplacian filling uniformly propagates coarse information while the image-domain sparsity prior suppresses wedge-induced ripples.

## Conclusion

This coursework provided a comprehensive exploration of CT image reconstruction techniques. In Part A, the fundamentals of Radon transform, FBP, and the ill-posedness of the inverse problem were demonstrated. SVD analysis clearly showed how data acquisition parameters (number of projections, angular range) affect the problem’s conditioning. Matrix-free regularized least-squares solvers (Tikhonov) and iterative soft-thresholding (ISTA) with wavelet sparsity priors proved effective in improving reconstruction quality, especially under noisy and data-limited conditions, outperforming standard FBP. Wavelet denoising was also shown to be a useful tool for post-processing or as part of reconstruction algorithms.

<sup>1</sup>SSIM computed with a  $7 \times 7$  Gaussian window ( $\sigma = 1.5$ ) following the original definition in [8].

In Part B, sinogram inpainting was investigated as a method to handle missing CT data. Both isotropic and anisotropic diffusion methods were implemented. Anisotropic inpainting showed superior performance for undersampled sinograms by better preserving structural details. However, for severely limited angle data, isotropic inpainting combined with image-domain post-processing was more robust. The results highlight that the optimal strategy for dealing with incomplete data is context-dependent. Overall, the project demonstrated the power of model-based and learning-inspired (sparsity) approaches in tackling inverse problems in imaging.

## References

- [1] R. Tovey, M. Benning, C. Brune, M. J. Lagerwerf, S. M. Collins, R. K. Leary, P. A. Midgley, and C.-B. Schönlieb, “Directional sinogram inpainting for limited angle tomography,” *Inverse Problems*, vol. 35, no. 11, p. 115006, 2019.
- [2] S. R. Deans, *The Radon Transform and Some of Its Applications*. Dover, 2 ed., 2007.
- [3] F. Natterer and F. Wübbeling, *Mathematics of Computerized Tomography*. SIAM, 2001.
- [4] P. C. Hansen, *Discrete Inverse Problems: Insight and Algorithms*. SIAM, 2010.
- [5] D. L. Donoho and I. M. Johnstone, “Adapting to unknown smoothness via wavelet shrinkage,” *Journal of the American Statistical Association*, vol. 90, no. 432, pp. 1200–1224, 1995.
- [6] J. Shen and T. F. Chan, “Mathematical models for local nontexture inpaintings,” *SIAM Journal on Applied Mathematics*, vol. 62, no. 3, pp. 1019–1043, 2002.
- [7] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, “Image inpainting,” in *Proc. SIGGRAPH*, pp. 417–424, 2000.
- [8] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: From error visibility to structural similarity,” *IEEE Transactions on Image Processing*, vol. 13, pp. 600–612, Apr. 2004.