Self-Supervised CT Image Reconstruction Using Noise2Inverse: Architecture Comparison and Multi-View Analysis

Po-Kai Chen and Maryia Zhyrko
Leiden University, Leiden, Netherlands
{s4283341, s4093771}@vuw.leidenuniv.nl

***Abstract*—Computed tomography reconstruction from noisy projection data remains challenging in medical imaging, particularly when balancing radiation exposure with image quality. This paper implements and evaluates the Noise2Inverse algorithm for self-supervised CT image denoising, comparing DnCNN and U-Net architectures across varying projection view counts. We conduct two main experiments: (1) architecture comparison using both synthetic phantom data and clinical CT images with 256, 512, and 1024 projection views, and (2) cross-view reconstruction analysis where models trained on one view count are applied to different view configurations. Our experiments use 1000 synthetic phantoms per view setting and clinical data from the Large COVID-19 CT Slice Dataset[5]. Our results highlight a trade-off between the architectures: DnCNN provides superior training stability and quantitative performance, while U-Net, despite clear overfitting on phantom data, occasionally shows better preservation of high-contrast features in clinical images. Furthermore, the cross-view analysis reveals a paradoxical result: models trained on noisier, lower-view-count data are significantly more robust and generalize better across different protocols than models trained on cleaner, high-quality data.**

***Index Terms*—computed tomography, self-supervised learning, image denoising, neural networks, Noise2Inverse**

I. Introduction

Computed tomography has become essential in medical diagnostics by providing detailed cross-sectional images of internal body structures. However, CT imaging faces a fundamental challenge: the trade-off between radiation dose and image quality. Low-dose CT protocols reduce patient exposure to ionizing radiation but introduce noise that can compromise diagnostic accuracy. This challenge is particularly important in pediatric imaging, screening protocols, and interventional procedures requiring repeated scans.

Traditional reconstruction methods like Filtered Back Projection (FBP)[2] are computationally efficient but sensitive to measurement noise. The linear nature of FBP directly propagates projection domain noise into reconstructed images, often resulting in streaking artifacts and reduced contrast-to-noise ratio. Iterative reconstruction methods can incorporate regularization to suppress noise but require significant computational resources and parameter tuning.

Deep learning approaches have shown promise for medical image denoising, but supervised methods require paired clean and noisy training data. In clinical CT imaging, obtaining ground truth clean images is often impractical as it would require exposing patients to additional radiation for high-dose reference scans.

Self-supervised learning methods address this limitation by learning to denoise from noisy data alone. The Noise2Self[1] algorithm exploits statistical independence of noise between pixels to enable training without clean references. However, when applied to inverse problems like CT reconstruction, noise correlation introduced by the reconstruction process violates the independence assumptions required by these methods.

The Noise2Inverse[4] algorithm, introduced by Hendriksen et al., addresses this limitation by recognizing that while reconstructed noise exhibits spatial correlations, measurement noise in the projection domain remains statistically independent. By partitioning projection data and creating multiple sub-reconstructions, Noise2Inverse enables training of denoising networks while preserving the statistical properties necessary for self-supervised learning.

Despite the theoretical foundation of Noise2Inverse, practical implementation questions remain. The choice of neural network architecture may significantly impact denoising performance and training stability. Additionally, the relationship between projection view count and reconstruction quality has not been thoroughly investigated. Understanding how models trained on one view configuration perform on different acquisition protocols is relevant for clinical deployment where scan parameters may vary.

This work addresses these gaps through comprehensive implementation and evaluation of Noise2Inverse. We compare DnCNN[6] and U-Net[3] architectures, motivated by their different design philosophies and established performance in medical imaging. We conduct systematic analysis across 256, 512, and 1024 projection views using both synthetic phantom data and clinical CT images. Additionally, we perform cross-view analysis to evaluate how models transfer between different acquisition protocols.

Our contributions include: (1) systematic comparison of DnCNN and U-Net architectures in the Noise2Inverse framework, (2) evaluation across different projection view counts using standardized datasets, (3) cross-view reconstruction analysis examining model transferability, and (4) assessment on both synthetic phantom data and clinical CT images from the Large COVID-19 CT Slice Dataset [5].

The remainder of this paper is organized as follows. Section II (Methodology) presents our experimental approach, includ-

ing the phantom generation process, noise2inverse training framework, and the two neural network architectures (DnCNN, U-Net) evaluated in this study. Section III (Experimental Setup) details our dataset creation, covering synthetic phantom generation, real CT image preparation, training procedures, and evaluation metrics. Section IV (Results) presents comprehensive performance comparisons across different projection view counts (256, 512, 1024) for both synthetic phantoms and real CT images, including quantitative PSNR and SSIM metrics alongside visual quality assessments. Finally, Sections V and VI (Discussion and Conclusions) analyze the implications of our findings for clinical applications, discuss limitations of the current approach, and outline directions for future research in self-supervised CT reconstruction.

II. Methodology

A. Computed Tomography Reconstruction

X-ray computed tomography physics can be described through the Radon transform, modeling X-ray attenuation as they traverse tissue. For parallel-beam geometry, the relationship between the unknown attenuation map $x \in \mathbb{R}^n$ and measured projection data $y \in \mathbb{R}^m$ is:

$$y = Ax + \epsilon$$

where A represents the discretized Radon transform matrix, and ϵ denotes measurement noise from photon counting statistics, electronic noise, and detection variability.

Reconstruction seeks to recover x from noisy measurements $\tilde{y} = y + \epsilon$. Filtered Back Projection applies a ramp filter in frequency domain followed by back-projection:

$$\tilde{x} = R\tilde{y} = Ry + R\epsilon$$

where R is the reconstruction operator. While ϵ may exhibit statistical independence across measurement indices, reconstructed noise $R\epsilon$ becomes spatially correlated due to back-projection. This correlation violates independence assumptions required by traditional self-supervised methods like Noise2Self.

B. Noise2Inverse Algorithm

1) Theoretical Foundation: Noise2Inverse addresses correlation problems by exploiting forward model structure to create statistically independent training pairs. The algorithm recognizes that while reconstructed noise exhibits correlations, measurement noise remains independent in the projection domain.

The theoretical foundation relies on decomposing expected prediction error into supervised and noise variance components. For any function h, expected prediction error can be written as:

$$E[\|h(\hat{x}_{J^c}) - \hat{x}_J\|^2] = E[\|h(\hat{x}_{J^c}) - x_J^*\|^2] + E[\|x_J^* - \hat{x}_J\|^2]$$

where x_J^* represents the clean sub-reconstruction from projection subset J, $\hat{x}_{J^c}$ is the noisy input sub-reconstruction from the complement subsets, and $\hat{x}_J$ is the noisy target sub-reconstruction. The first term represents supervised prediction error, while the second term represents reconstructed noise variance, independent of h choice.

This decomposition shows that minimizing expected prediction error is equivalent to minimizing the difference between network output and unknown clean reconstruction, even when training only on noisy data. Statistical independence between input and target sub-reconstructions, achieved through projection domain partitioning, ensures cross-correlation terms vanish in expectation.

2) Implementation Strategy: The Noise2Inverse implementation involves several steps. First, projection data is partitioned into K disjoint subsets, with each subset containing every K-th projection angle. This angular interleaving ensures each subset provides uniform angular sampling while maintaining statistical independence.

We employ $K = 4$ splits based on empirical evidence and preliminary experiments. This choice balances statistical independence (improving with larger K) and reconstruction quality of individual sub-reconstructions (degrading with sparse angular sampling).

The training process uses the X:1 strategy, where input consists of averaged reconstruction from $K - 1$ subsets, and target is reconstruction from the remaining subset. This strategy provides higher signal-to-noise ratio input compared to 1:X alternatives, facilitating more stable training dynamics[4].

For each training sample, we generate an input consisting of the average of reconstructions from 3 projection subsets, a target consisting of the reconstruction from the remaining projection subset, and ground truth consisting of the clean reconstruction from all projections used for evaluation only.

C. Neural Network Architectures

1) DnCNN Architecture: DnCNN was selected for its proven effectiveness in image denoising and architectural simplicity that reduces overfitting risk. The key innovation is residual learning, where the network learns to predict noise components rather than clean images directly. This design is well-suited for denoising applications as it allows focus on additive noise while preserving underlying image structure.

The architecture consists of convolutional layers with batch normalization and ReLU activation. First and last layers use bias terms, while intermediate layers employ bias-free convolutions followed by batch normalization. This design facilitates gradient flow and enables training deeper networks without vanishing gradients.

The DnCNN architecture uses 15 layers and 80 features by default as defined in the model implementation, with the first and last layers using bias terms while intermediate layers employ bias-free convolutions followed by batch normalization. The hyperparameter optimization explores layer counts from 5 to 17 and feature channel counts of 32, 48, 64, 80, and 96 as specified in the HPO code.

2) U-Net Architecture: U-Net was included due to its widespread adoption in medical image analysis and unique architectural properties. The encoder-decoder structure with

skip connections enables capturing both local and global image features, potentially providing advantages for preserving anatomical structures while removing noise.

Skip connections serve dual purposes: facilitating gradient flow during training and preserving high-resolution features that might be lost in encoder-decoder bottlenecks. This characteristic is relevant for medical imaging where preservation of fine anatomical details is crucial for diagnostic accuracy.

However, increased U-Net complexity introduces challenges. Larger parameter count increases overfitting risk, particularly with limited training data. Additionally, multi-scale feature processing may be less suited for specific noise characteristics of CT reconstruction, where dominant noise sources are often high-frequency streaking artifacts.

D. Cross-View Analysis

Beyond architecture comparison, we investigate how models trained on one projection view count perform when applied to different view configurations. This cross-view analysis addresses practical deployment scenarios where acquisition protocols may vary between scanners or clinical requirements.

We train models on 256, 512, and 1024 projection views separately, then evaluate each trained model on test data from all three view configurations. This provides insights into model transferability and robustness across different acquisition protocols.

The cross-view experiment motivation stems from clinical scenarios where different scanners may use varying projection counts, protocol optimization may require changing view counts, emergency scenarios may necessitate faster lower-view acquisitions, and model deployment across multiple institutions with different equipment configurations may be required.

III. Experimental Setup

A. Dataset Generation

1) Phantom Data Generation: Our phantom generation creates controlled test cases that capture essential characteristics of clinical CT data while providing ground truth for quantitative evaluation. Unlike anatomically realistic phantoms, our approach focuses on geometric complexity that challenges denoising algorithms.

Each phantom is a 256×256 pixel image composed of randomly positioned geometric elements with varying complexity levels. The phantoms include between 2-4 major regions, with each region containing 8-15 detail elements and 30-60 foam elements. Large geometric shapes include circles, squares, rectangles, ellipses, and polygons representing major anatomical regions, with attenuation values ranging from 0.1 to 1.5 corresponding to air-filled spaces through bone structures. Additionally, the phantoms contain fine-scale circular elements with radii ranging from 1 pixel to approximately $\text{size}/20$ pixels, creating foam-like regions that simulate microscopic tissue details and background variations to challenge denoising algorithms.

The fine-scale elements serve dual purposes: providing texture variation that challenges denoising algorithms, and simulating complexity of trabecular bone, lung parenchyma, and other fine anatomical structures. The random positioning and sizing of elements ensures each phantom instance is unique, preventing overfitting during training.

For each projection view configuration (256, 512, 1024), we generate 1000 unique phantom instances. This provides sufficient diversity for network training while remaining computationally tractable. Each phantom undergoes projection simulation and reconstruction to create the full dataset.

2) Clinical CT Data: Clinical evaluation uses images from the Large COVID-19 CT Slice Dataset[5], accessed through Kaggle. This dataset provides real-world CT images with authentic noise characteristics and anatomical complexity. We processed the non-COVID subset to ensure consistent data characteristics.

The preprocessing pipeline includes selection of 3 images per patient to ensure diversity while maintaining computational feasibility, resizing to 256×256 pixels for consistency with phantom data, and intensity normalization to $[0, 1]$ range for consistent dynamic range across scanner types.

Clinical data provides evaluation on real-world complexity including authentic artifacts, anatomical variations, and noise characteristics that may not be fully captured in synthetic phantoms. The dataset addresses potential domain shift between controlled phantom studies and clinical acquisition reality.

B. Projection Simulation and Noise Modeling

Projection simulation implements physically realistic forward modeling using the Radon transform with parallel-beam geometry. This provides good approximation for fan-beam CT when object size is small relative to source-detector distance.

Projection angles use uniform spacing from 0° to 180°, consistent with standard CT protocols. The choice of 256, 512, and 1024 views spans from sparse-view CT (suitable for dynamic imaging or dose reduction) to high-resolution acquisitions for detailed diagnostic evaluation.

Noise modeling employs Gaussian noise added to the sinogram domain before reconstruction. The noise is generated with standard deviation proportional to the maximum sinogram value to ensure consistent noise-to-signal ratios across different data characteristics. The noise level is set to 0.05, providing realistic noise conditions typical in clinical low-dose protocols while remaining challenging for denoising algorithms.

C. Training Methodology

Training employs PyTorch with automatic differentiation and GPU acceleration. For each experiment, we use a three-way data split with 80% for training, 10% for validation, and 10% for testing.

Due to the need to train a total of 12 models, it is not computationally feasible to perform hyperparameter optimization for each one individually. Instead, we use a fixed set of hyperparameters that have been empirically shown to perform well across similar settings. These values, shown in Table I, were selected based on prior experience and practical considerations.

TABLE I: Architectural hyperparameters for UNet and DnCNN models.

Model	Hyperparameter	Value
UNet	Initial Feature Size	64
	Encoder Depth	4
	Bottleneck Channels	1024
	Convolution Kernel Size	3 (with padding=1)
	Activation Function	ReLU
DnCNN	Number of Layers	15
	Feature Channels	80
	Convolution Kernel Size	3 (with padding=1)
	Activation Function	ReLU

Training uses the Adam optimizer with a default learning rate of 5×10^{-4} and mean-squared-error (MSE) loss. The batch size is 16 and the number of epochs is 100.

D. Evaluation Methodology

Evaluation employs quantitative metrics and qualitative assessment. Peak Signal-to-Noise Ratio (PSNR) provides standardized reconstruction fidelity measure widely used in image processing. Structural Similarity Index Measure (SSIM) incorporates perceptual factors including luminance, contrast, and structure, better capturing preservation of anatomical structures critical for diagnostic accuracy.

Qualitative evaluation includes visual assessment of reconstructed images and residual images showing differences between denoised and ground truth reconstructions. This assessment identifies potential artifacts or systematic biases not captured by quantitative metrics.

The cross-view analysis examines performance variation with projection view count, providing insights into practical trade-offs between acquisition time and image quality. This analysis is relevant for protocol optimization in clinical settings where scan time impacts patient throughput and motion artifacts.

IV. Results

A. Training Dynamics Analysis

Figure 1 shows the training dynamics for both DnCNN and U-Net architectures on clinical CT data across different projection view counts. The training loss curves demonstrate that DnCNN achieves faster convergence and more stable training across all view configurations. For DnCNN, the training loss consistently decreases and stabilizes, while U-Net shows less stable convergence with higher final loss values.

The validation PSNR curves reveal significant differences between architectures. DnCNN maintains consistently higher and more stable PSNR values throughout training across all view configurations. The PSNR curves show steady improvement and stabilization for DnCNN, with 1024 views achieving the highest performance, followed by 512 views, and then 256 views. In contrast, U-Net shows highly erratic validation PSNR behavior with substantial drops in performance during training, particularly evident in the 256 and 512 view configurations where PSNR values decrease significantly after initial improvements, indicating severe overfitting.

The validation SSIM metrics confirm the PSNR results. DnCNN achieves stable SSIM values that gradually improve and remain consistent across all view configurations. Its SSIM curves exhibit smooth progression with minimal fluctuations. In contrast, U-Net displays highly unstable training behavior, with sharp drops in SSIM values—particularly under lower view counts. This instability suggests that U-Net may be more susceptible to overfitting within the self-supervised Noise2Inverse framework, where the training signal inherently contains noise correlations. One possible explanation is that U-Net's higher capacity makes it more prone to memorizing noise rather than generalizing. This raises the question of whether a smaller, more regularized version of U-Net might perform better in this context.

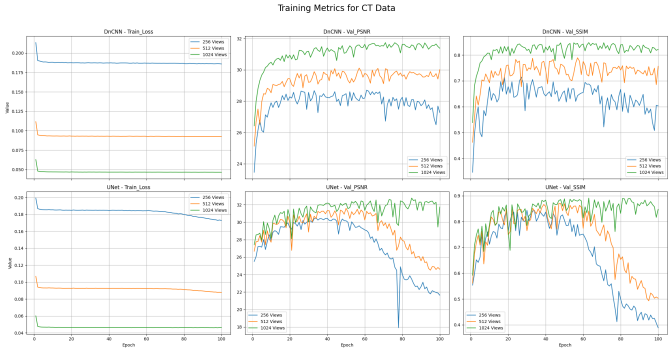


Fig. 1: Training dynamics for DnCNN and U-Net on clinical CT data. Top row shows DnCNN results, bottom row shows U-Net results. Left column: training loss, middle column: validation PSNR, right column: validation SSIM. DnCNN demonstrates superior convergence stability across all metrics and view configurations.

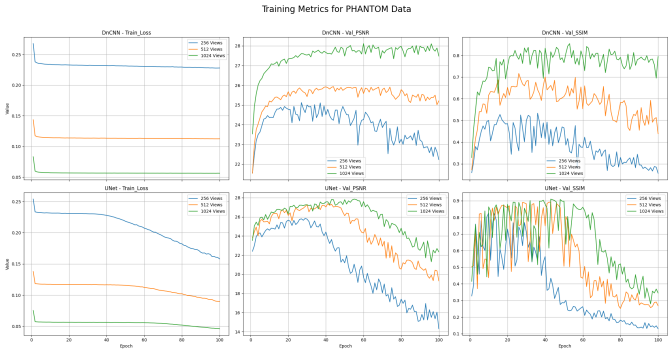


Fig. 2: Training dynamics for DnCNN and U-Net on synthetic phantom data. The patterns mirror clinical data results, with DnCNN showing superior stability. U-Net exhibits severe training instability particularly evident in PSNR and SSIM degradation during training.

Figure 2 presents the training dynamics on synthetic phantom data. The results show similar patterns to the clinical data, with DnCNN achieving more stable training and better final performance metrics. However, the phantom data reveals even more pronounced instability in U-Net training, with severe

degradation in both PSNR and SSIM values during training for all view configurations.

The phantom data results demonstrate even more extreme instability for U-Net compared to clinical data. The PSNR curves for U-Net show catastrophic drops during training, with values plummeting from reasonable initial levels to significantly degraded performance. This behavior is consistent across all three view configurations, but most severe in the 256-view case. The SSIM curves exhibit similar deterioration, with U-Net showing progressive degradation throughout training, in stark contrast to the steady improvement observed with DnCNN. Notably, however, even the DnCNN model begins to overfit in the 256-view setting toward the end of training, as seen in the declining PSNR and SSIM curves. This may be due to the increased noise in the input data resulting from fewer views in the Radon transformation, which can cause the network to learn and fit noise instead of meaningful structure. Nevertheless, the significantly more severe and consistent degradation in U-Net across all view configurations reinforces our previous observation: the U-Net architecture used here is overly complex for this self-supervised task and appears fundamentally incompatible with the learning paradigm.

B. Clinical CT Data Results

Figures 3a and 3b present the reconstruction results on clinical CT data for DnCNN and U-Net, respectively. A visual inspection confirms that both architectures successfully reduce noise compared to the input images, with DnCNN generally providing more effective removal of streaking artifacts. The reconstructions from 256 views are notably more blurry for both models, an expected consequence of the sparse projection data.

An interesting observation arises when comparing the models. Despite its unstable training dynamics and inferior quantitative metrics, the U-Net model in some cases appears to preserve strong black-and-white contrast and the general shape of anatomical structures more effectively than DnCNN, as seen in Figure 3. This suggests its architectural biases may favor the retention of high-contrast features even while failing to remove finer noise textures.

Furthermore, it is noteworthy that models trained on 1024 views do not always yield superior reconstruction quality. Even with a less noisy input, the denoised images can sometimes appear slightly over-smoothed. This implies that models trained on data with a moderate amount of noise (i.e., from fewer views) may develop more robust representations. The exposure to more significant artifacts could force the network to learn more generalized features, a finding that has implications for training strategies in real-world applications where data quality varies.

C. Phantom Data Results

The reconstruction results on synthetic phantom data, presented in Figures 4a and 4b, reinforce the conclusions drawn from the clinical data and offer clearer insights into the models' behaviors.

It is immediately apparent from Figure 4b that the U-Net architecture is performing poorly on the phantom data. For each view count, the reconstruction is quite noisy and fails to adequately restore the underlying geometric shapes, making the results not comparable to those generated by DnCNN. This outcome is consistent with the training instability discussed previously and confirms the model's limitations within this framework.

In contrast, the DnCNN results in Figure **??** show much more effective denoising. A clear performance trend is visible, where the 256-view case apparently performs the worst, while the 512-view and 1024-view reconstructions are qualitatively quite similar. However, it is also worth noticing a key failure mode: when distinct objects in the ground truth have very similar intensity values, the reconstruction tends to ignore the boundary and merge them. This seems to result from the noise in the input image completely eliminating the signal of the boundary, leaving the model with no information to recover it. This underscores a critical point: even as we develop better denoising models, it is essential to dedicate effort to reducing measurement noise at the source, as no post-processing algorithm can restore information that is fundamentally lost.

D. Cross-View Analysis on Clinical Data

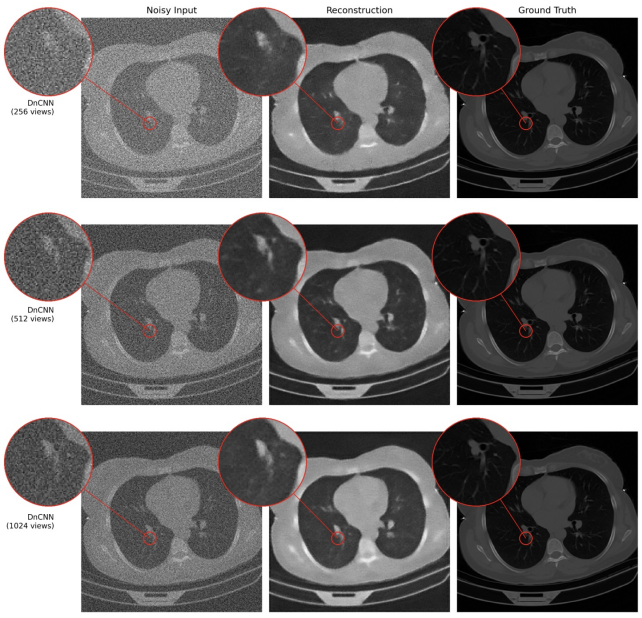
Given the demonstrated instability of the U-Net architecture, this cross-view analysis focuses exclusively on the DnCNN models to evaluate their transferability. Figure 5 presents this analysis, where rows represent a model trained on a specific view count and columns represent the input data it was tested on. As expected, the reconstruction quality along the diagonal improves with more views; the reconstruction from the 512-view model on 512-view data is better than the 256-view case, and the 1024-view case is better still.

The key insight, however, lies in the off-diagonal elements which reveal how robust each model is. The model trained on 256 views, while yielding a noisy result on its native 256-view input, demonstrates remarkable flexibility. When this same model is applied to the cleaner 512- and 1024-view inputs, its performance improves markedly. In contrast, models trained on cleaner data are less robust. The model trained on 1024 views, for example, excels on high-quality input but largely fails when confronted with the highly corrupted 256-view data, resulting in a poor reconstruction with heavy artifacts.

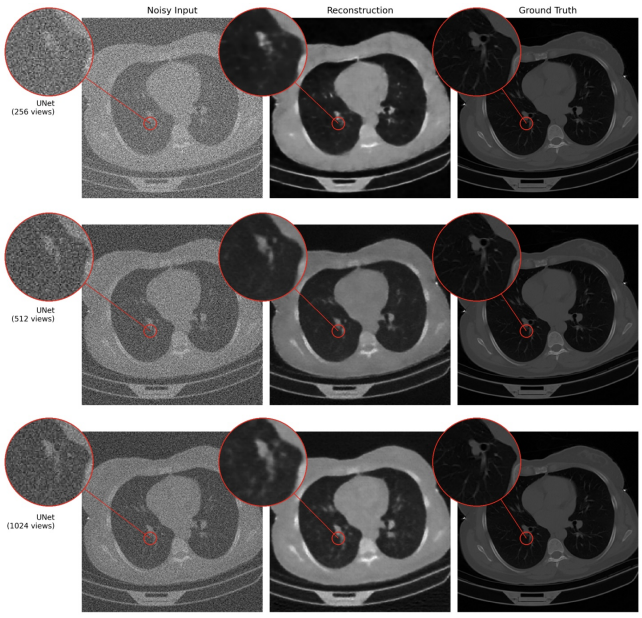
This asymmetric transferability suggests that training on more challenging, noisier data acts as a form of regularization, forcing the model to learn more generalizable features. The resulting model is less specialized to a single, high-quality condition and therefore more robust. This implies that for developing models for real-world clinical use, training on more difficult data may be a superior strategy for achieving broad applicability across different scanning protocols.

E. Cross-View Analysis on Phantom Data

Figure 6 shows the cross-view analysis on phantom data, which provides further evidence for the transferability patterns observed in the clinical setting and validates them in a more

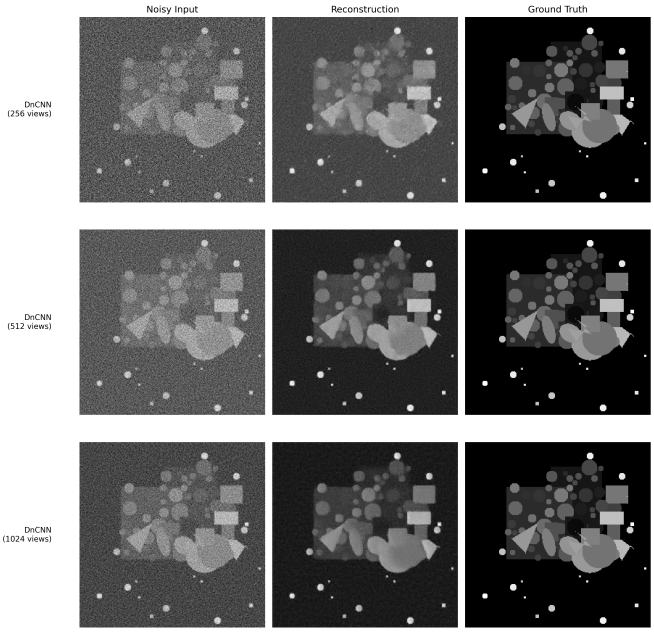


(a) DnCNN reconstruction results showing noisy input, reconstruction, and ground truth for different view counts.

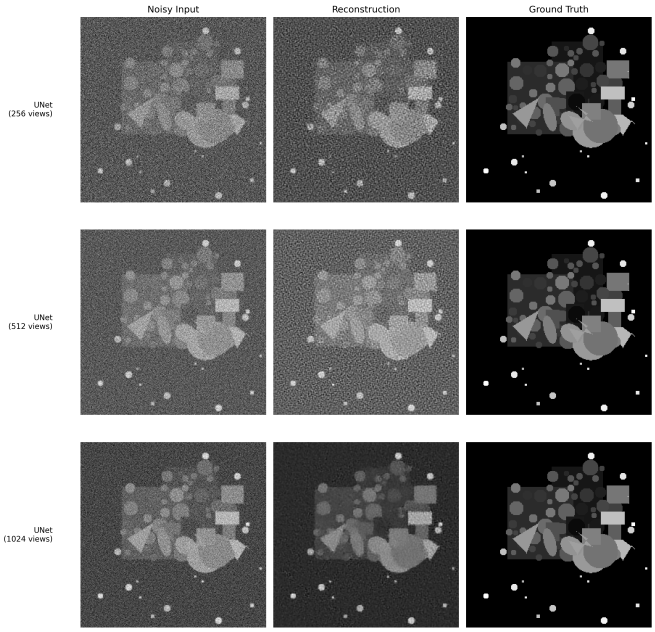


(b) UNet reconstruction results showing a similar layout. The performance comparison highlights differences in artifact removal.

Fig. 3: Visual comparison of DnCNN and UNet reconstruction performance on clinical CT data across 256, 512, and 1024 projection views.



(a) DnCNN reconstruction results on synthetic phantom data.



(b) UNet reconstruction results on the same phantom data.

Fig. 4: Visual comparison of DnCNN and UNet reconstruction performance on synthetic phantom data across 256, 512, and 1024 projection views.

controlled environment. The results unambiguously demonstrate the same asymmetric transferability seen previously.

Specifically, the model trained on the noisiest 256-view data shows remarkable robustness. While its reconstruction on the native 256-view input is grainy, it adapts exceptionally well when applied to cleaner inputs, producing progressively sharper reconstructions of the geometric shapes for the 512- and 1024-view cases. Conversely, the model trained on the clean 1024-view data proves to be brittle. It fails catastrophically when applied to the 256-view input, converting the grainy noise into large, blotchy artifacts and failing to resolve the phantom's structural details.

This controlled experiment provides strong validation for the hypothesis that training on noisier, more challenging data

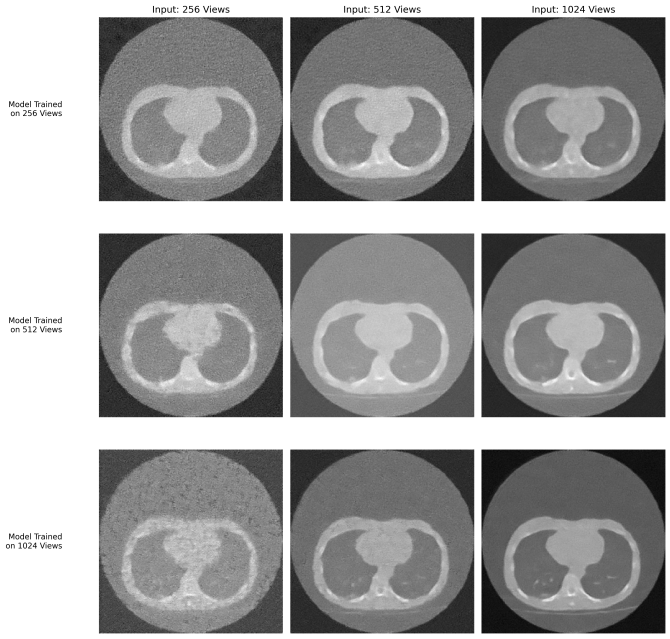


Fig. 5: Cross-view analysis on clinical CT data. Rows represent DnCNN models trained on different view counts, columns represent input data from different view counts. The diagonal shows models applied to their training view count, while off-diagonal elements show transferability.

acts as a powerful regularizer, forcing the model to learn more fundamental features. The consistent findings across both phantom and clinical data confirm that developing models on lower-view-count data can lead to more robust and general-purpose solutions for CT denoising.

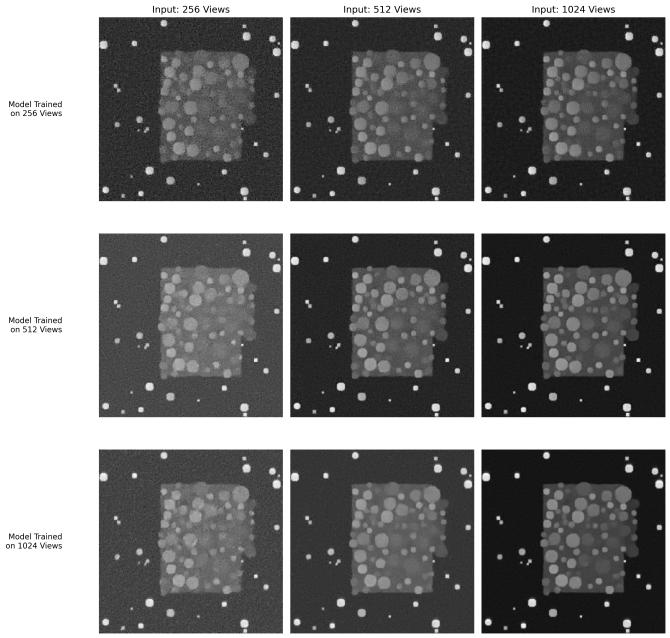


Fig. 6: Cross-view analysis on phantom data with the same layout as Figure 5. Results confirm transferability patterns observed in clinical data.

V. Discussion

Our findings offer a nuanced perspective on self-supervised denoising for CT reconstruction, highlighting the interplay between network architecture, data quality, and model robustness. In this section, we summarize the key insights from our results, reflect on the study's limitations, and outline potential directions for future research.

A. Architectural Performance: Beyond the Metrics

Quantitative metrics and training dynamics told a clear story: DnCNN delivered stable and consistently strong performance, while U-Net showed significant signs of overfitting. However, visual inspection of the clinical CT results presents a more complex picture. In some cases, U-Net's reconstructions, although noisier overall, appeared to preserve high-contrast anatomical boundaries more sharply than DnCNN. This suggests that U-Net's multi-scale design, despite its tendency to memorize noise in this setting, may have some inherent advantages in retaining structural detail.

These subtleties were not present in the phantom data. On these clean, synthetic images, U-Net's overfitting was evident both quantitatively and visually, with reconstructions suffering from prominent noise and distortion. These results reinforce the idea that, within the Noise2Inverse framework where the goal is to distinguish signal from statistically independent noise, the architectural simplicity and residual-focused design of DnCNN are better suited than the more complex U-Net.

B. The Role of View Count: A Trade-off Between Quality and Robustness

Our cross-view analysis revealed a key insight: models trained on noisier data with fewer views showed greater robustness, generalizing better to cleaner inputs. However, this does not imply that reducing view count is always beneficial. Reconstructions from the 256-view models consistently had the lowest quality, indicating a clear trade-off between robustness and fidelity. Notably, models trained on 512 views often performed similarly to those trained on 1024 views. This suggests a point of diminishing returns, where increasing the number of views does not necessarily lead to better outcomes and may even reduce the model's flexibility.

These observations suggest a promising strategy for training denoising models. In clinical settings where the acquisition protocol uses a high number of views, one could simulate lower-view data during training to encourage the network to develop robustness. The trained model could then be applied to the original high-quality reconstruction, potentially combining the benefits of high-fidelity input with improved generalization. Further work is needed to explore this balance between data quality during training and performance during deployment.

C. Strengths, Limitations, and Future Work

The primary strength of this study lies in its systematic, head-to-head comparison of DnCNN and U-Net within the Noise2Inverse framework, supported by a novel cross-view

analysis on both clinical and phantom datasets. This setup enabled us to uncover the important relationship between training data quality and model robustness.

That said, there are several limitations to consider. First, we did not conduct hyperparameter optimization due to computational constraints. Both architectures were used in standard configurations, and it is plausible that U-Net's overfitting could be mitigated by reducing its capacity, for example by using fewer channels or layers. A dedicated hyperparameter tuning study focused on adapting U-Net to this task would be a valuable next step. Second, our experiments were limited to two network architectures and assumed a parallel-beam geometry with additive Gaussian noise. Expanding this analysis to include a broader range of architectures and more realistic acquisition models, such as fan-beam geometry and compound Poisson-Gaussian noise, would help to better approximate real-world clinical conditions.

VI. Conclusion

In this study, we systematically compared the DnCNN and U-Net architectures for self-supervised CT image denoising using the Noise2Inverse framework, evaluating their performance across different numbers of projection views. Our findings clearly show that DnCNN is better suited for this task, offering more stable training and consistently stronger quantitative results. Although U-Net occasionally preserved high-contrast clinical features, it struggled with overfitting—particularly evident in its poor performance on phantom data—making it less reliable in its default form.

One of the most striking insights from our work is the asymmetric transferability of the trained models. Surprisingly, models trained on noisier, low-view-count data generalized better across varying noise levels than those trained on cleaner, high-view-count data. This result challenges the common assumption that better training data always leads to better models. In self-supervised inverse problems like CT reconstruction, our findings highlight a key trade-off between the quality of the training data and the model's ability to generalize.

Overall, our work emphasizes the importance of jointly designing network architectures and training strategies that account for the unique statistical challenges of self-supervision. By showing that training on more challenging data can act as a form of regularization, we offer a new perspective for building more robust and clinically practical solutions for low-dose CT denoising.

References

[1] Joshua Batson and Loic Royer. *Noise2Self: Blind Denoising by Self-Supervision*. Jan. 2019. DOI: 10.48550/arXiv.1901.11365.

[2] Thorsten Buzug and Dimitris Mihailidis. "Computed Tomography From Photon Statistics to Modern Cone-Beam CT". In: *Medical Physics* 36 (Aug. 2009), pp. 3858–. DOI: 10.1118/1.3176026.

[3] Yi Cai and Jiangying Yuan. "A Review of U-Net Network Medical Image Segmentation Applications". In: *Proceedings of the 2022 5th International Conference on Artificial Intelligence and Pattern Recognition*. AIPR '22. Xiamen, China: Association for Computing Machinery, 2023, pp. 457–461. ISBN: 9781450396899. DOI: 10.1145/3573942.3574048. URL: https://doi.org/10.1145/3573942.3574048.

[4] Allard Adriaan Hendriksen, Daniel Maria Pelt, and K. Joost Batenburg. "Noise2Inverse: Self-Supervised Deep Convolutional Denoising for Tomography". In: *IEEE Transactions on Computational Imaging* 6 (2020), pp. 1320–1335. ISSN: 2573-0436. DOI: 10.1109/tci.2020.3019647. URL: http://dx.doi.org/10.1109/TCI.2020.3019647.

[5] M. Maftouni et al. "A Robust Ensemble-Deep Learning Model for COVID-19 Diagnosis based on an Integrated CT Scan Images Database". In: *Proceedings of the 2021 Industrial and Systems Engineering Conference*. Virtual Conference. Virtual Conference, May 2021, pp. 22–25.

[6] Kai Zhang et al. "Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising". In: *CoRR* abs/1608.03981 (2016). arXiv: 1608.03981. URL: http://arxiv.org/abs/1608.03981.