I-INR: Iterative Implicit Neural Representations

Ali Haider[1], Muhammad Salman Ali[1], Maryam Qamar[1], Tahir Khalil[1], Soo Ye Kim[2], Jihyong Oh[3*], Enzo Tartaglione[4], Sung-Ho Bae[1†]

[1]Kyung Hee University, Republic of Korea
[2]Adobe Research
[3]Chung-Ang University, Republic of Korea
[4]LTCI, Télécom Paris, Institut Polytechnique de Paris, France
{alihaider, salmanali, maryamqamar, tahirkhalil26}@khu.ac.kr, sooyek@adobe.com,
jihyongoh@cau.ac.kr, enzo.tartaglione@telecom-paris.fr, shbae@khu.ac.kr

Abstract

Implicit Neural Representations (INRs) have revolutionized signal processing and computer vision by modeling signals as continuous, differentiable functions parameterized by neural networks. However, INRs are prone to the spectral bias problem, limiting their ability to retain high-frequency information, and often struggle with noise robustness. Motivated by recent trends in iterative refinement processes, we propose Iterative Implicit Neural Representations (I-INRs). This novel plug-and-play framework iteratively refines signal reconstructions to restore high-frequency details, improve noise robustness, and enhance generalization, ultimately delivering superior reconstruction quality. I-INRs integrate seamlessly into existing INR architectures with only a 0.5–2% increase in parameters. During reconstruction, the iterative refinement adds just 0.8–1.6% additional FLOPs over the baseline while delivering a substantial performance boost of up to +2.0 PSNR. Extensive experiments demonstrate that I-INRs consistently outperform WIRE, SIREN, and Gauss across various computer vision tasks, including image fitting, image denoising, and object occupancy prediction. The code is available at https://github.com/optimizer077/I-INR.

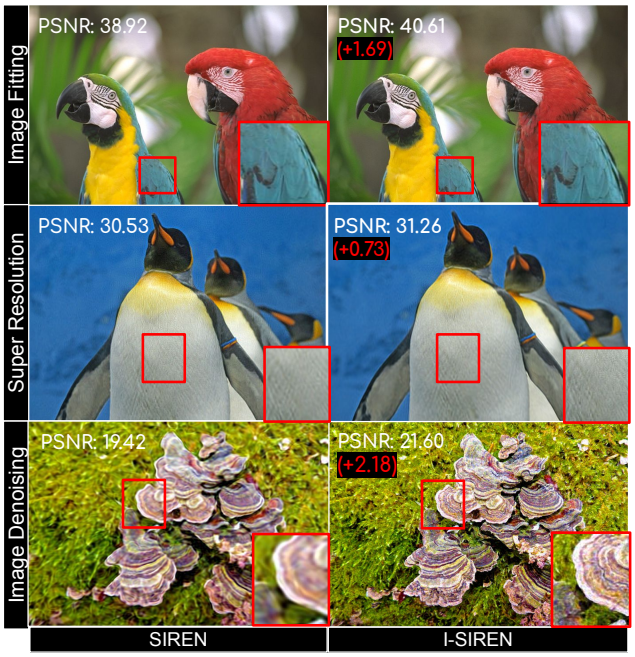


Figure 1: Effectiveness of the proposed methods across multiple tasks, including image fitting, super resolution (2× scale), and denoising, compared to the baseline representative INR method (SIREN). Our novel Iterative-INR method (I-SIREN) consistently improves detail preservation, fidelity, and high-frequency reconstruction across all tasks, outperforming the baseline.

1 Introduction

Implicit Neural Representations (INRs) have emerged as a transformative approach in signal representation, shifting from traditional discrete grid-based methods to continuous coordinate-based models (Sitzmann et al. 2020). By leveraging neural networks, typically multi-layer perceptrons (MLPs), INRs map spatial or temporal coordinates directly to signal attributes such as pixel intensity, colour, or 3D occupancy. This continuous representation inherently offers resolution independence, compact encoding, and seamless interpolation, making INRs highly versatile across a wide range of domains, including computer vision (Chen, Liu, and Wang 2021; Saragadam et al. 2022; Sitzmann et al. 2020; Mildenhall et al. 2021; Jiang, Hua, and Han 2023), and beyond (Sun et al. 2021; Shen et al. 2021; Zhong et al. 2019; Reed et al. 2021). Initially introduced for tasks like 3D shape reconstruction (Mildenhall et al. 2021; Peng et al. 2020; Srinivasan et al. 2021) and novel view synthesis (Irshad et al. 2023), INRs have since been applied to diverse applications including image fitting (Liu et al. 2023b), super-resolution (Delbracio and Milanfar 2023; Saharia et al. 2022), and solving inverse problems (Cha et al. 2024; Wang et al. 2019; Chen, Liu, and Wang 2021).

Despite advancements, INRs continue to face substantial challenges in capturing high-frequency details, maintaining robustness against noise, and handling incomplete data (Saragadam et al. 2023). Typically optimized with L1 or L2 loss (Sitzmann et al. 2020), INRs inherently exhibit a

*Corresponding Authors

spectral bias that favors low-frequency components, often at the expense of fine-grained details (Rahaman et al. 2019).

To mitigate this, recent approaches have introduced positional encoding schemes (Tancik et al. 2020; Fathony et al. 2020) and alternative activation functions (Sitzmann et al. 2020; Saragadam et al. 2023; Liu et al. 2023b; Gao and Jaiman 2024; Thennakoon et al. 2025) to better capture high-frequency content. Positional encodings inject high-frequency signals through orthogonal Fourier bases (Tancik et al. 2020), while periodic activations such as sine functions enable MLPs to model high-frequency structures more effectively compared to traditional ReLU-based networks (Sitzmann et al. 2020). Nevertheless, these methods often assume clean and fully observed inputs, limiting their effectiveness in practical scenarios characterized by noise and occlusion (Saragadam et al. 2023). More recent efforts, such as the integration of complex Gabor wavelet activations, have aimed to improve robustness under such conditions (Saragadam et al. 2023), but challenges persist when scaling to real-world, noisy data. Consequently, there remains significant potential to enhance the fidelity, robustness, and generalization of INRs, particularly in preserving high-frequency signals under adverse conditions.

To address these limitations, and drawing inspiration from iterative models (Delbracio and Milanfar 2023; Rissanen, Heinonen, and Solin 2022; Ho, Jain, and Abbeel 2020), we introduce *Iterative Implicit Neural Representations (I-INRs)* a plug-and-play framework that integrates seamlessly with existing implicit architectures. Unlike traditional single-shot INRs, I-INRs reconstruct signals progressively over multiple iterations, refining details, enhancing reconstruction quality, and improving noise robustness, as illustrated in Figure 1.

The proposed framework features a novel architecture that incorporates existing implicit networks as a BackboneNet without major modifications, while introducing two lightweight modules, i.e FeedbackNet and FuseNet as shown in Figure 2a. These blocks add only 0.5–2% to the total parameter count of the baseline model and support the iterative reconstruction process. Owing to this design, the iterative I-INR process incurs only 0.8–2% additional FLOPs over the baseline INR for two-step reconstruction. Our experiments demonstrate that I-INRs deliver superior reconstructions with enhanced detail, better generalization, and minimal computational overhead, consistently outperforming single-shot methods such as WIRE (Saragadam et al. 2023), SIREN (Sitzmann et al. 2020), and Gauss (Ramasinghe and Lucey 2022) across tasks including image fitting, super-resolution, denoising, and 3D occupancy reconstruction.

Our main contributions are as follows:

- We present **I**terative **I**mplicit **N**eural **R**epresentations, called **I-INRs**, a novel framework for INRs that reconstructs signals *iteratively*, effectively capturing high-frequency details, enhancing robustness to noise, and improving generalization.
- I-INRs are designed as a plug-and-play framework, compatible with existing implicit architectures, enabling seamless integration and adoption in various applications.
- We validate the effectiveness of I-INRs through extensive experiments across multiple tasks, demonstrating superior performance over traditional single-shot INRs in terms of detail reconstruction and robustness.

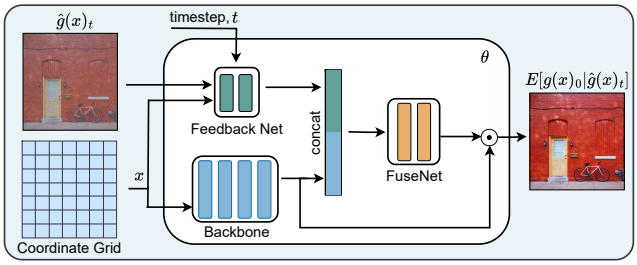
2 Related Work

Spatial Encoding Techniques. INRs suffer from spectral bias (Rahaman et al. 2019; Tancik et al. 2020), struggling to capture high-fidelity details of complex signals. This limitation affects applications requiring fine-grained detail, such as a single image super resolution (SR), 3D reconstruction, and scene modeling. To address this challenge, various approaches have been proposed, including spatial encoding with frequency-based representations, polynomial decomposition (Raghavan et al. 2023; Singh, Shukla, and Turaga 2023), and high-pass filtering (Fathony et al. 2020), all of which emphasize high-frequency components. DINER (Xie et al. 2023) applies a hash map to unevenly map input coordinates to feature vectors, optimizing spatial frequency distribution for faster, more accurate reconstructions.

Role of Activations. Beyond encoding, activation functions critically influence spectral bias. Standard choices like ReLU cannot capture high-frequency components due to limited representational capacity (Sitzmann et al. 2020). To address this, alternatives such as sinusoidal (Sitzmann et al. 2020) and Gaussian (Ramasinghe and Lucey 2022) activations have been proposed. Activations tuned for high accuracy can suffer from reduced noise robustness; WIRE (Saragadam et al. 2023) leverages Gabor wavelets to provide enhanced robustness to noise. However, there remains room for improvement in achieving both high-frequency fitting and noise robustness (Saragadam et al. 2023).

Iterative Methods. Iterative models have been very successful in image restoration and translation (Saharia et al. 2022; Rissanen, Heinonen, and Solin 2022; Delbracio and Milanfar 2023; Hui et al. 2024; Chu et al. 2025; Chen et al. 2024), video generation (Ji et al. 2025), language modeling (Nie et al. 2025), 3D generation (Qian et al. 2023; Liu et al. 2023a), and more. In contrast with single-shot models, iterative models such as diffusion models (Ho, Jain, and Abbeel 2020; Rissanen, Heinonen, and Solin 2022; Delbracio and Milanfar 2023) produce high-quality samples by reversing degradation processes over multiple steps. Despite their widespread adoption and effectiveness, iterative models remain underexplored in the context of implicit neural representations.

Inspired by the strong detail reconstruction capabilities of iterative models, we propose a novel INR framework that reconstructs signals implicitly over multiple steps. Our approach provides improved details and enhanced noise robustness, offering a flexible, plug-and-play solution that can be integrated into any existing INR framework. However, for the scope of this paper, we specifically focus on WIRE (Saragadam et al. 2023), Gauss (Ramasinghe and Lucey 2022), and SIREN (Sitzmann et al. 2020).



(a)

(b)

Figure 2: (a) The proposed architecture of the I-INR model. The framework consists of a Backbone, which can be any baseline INR architecture. Additionally, a Feedback Net incorporates feedback to refine representations, while FuseNet integrates features. The final output is obtained by combining the outputs of the Backbone and the FuseNet, enhancing expressivity and reconstruction quality. (b) Iterative reconstruction process of the proposed I-INR framework.

3 Preliminary

3.1 Standard INR

Let $g : X \subseteq \mathbb{R}^p \rightarrow Y \subseteq \mathbb{R}^q$ be a target function mapping p-dimensional inputs (such as pixel positions in a 2D image or volumetric domains in 3D scenes) to q-dimensional outputs (such as color or density values), such that $y = g(x)$ for $x \in X$. An INR (Sitzmann et al. 2020) is a learnable neural function $f_\theta : X \rightarrow Y$, parameterized by θ, aiming to approximate $\mathcal{I}(x)$ across X. The objective is to find θ such that:

$$f_\theta(x) \approx \mathcal{I}(x), \tag{1}$$

where $\mathcal{I}(x)$ represents a true image function evaluated at coordinates x. A typical INR is trained by reducing the signal error in an L_p metric:

$$\min_\theta \mathbb{E}_x \left[\|f_\theta(x) - \mathcal{I}(x)\|_p \right] \approx \min_\theta \sum_i \|f_\theta(x^i) - \mathcal{I}(x^i)\|_p, \tag{2}$$

where $p = 2$ for standard INR training, corresponding to the mean squared error.

3.2 Inversion by Direct Iteration (InDI)

Inversion by Direct Iteration (InDI) (Delbracio and Milanfar 2023) addresses ill-posed inverse problems, such as image restoration, by iteratively refining an estimate of the unknown signal b from an observed measurement a. Instead of solving the full inverse problem in one step, InDI interpolates between a and b via $b_t = ta + (1-t)b$, with $t \in [0, 1]$, and uses a neural network F_θ to approximate the conditional expectation $\mathbb{E}[b \mid b_t]$. The network is trained to minimize:

$$\min_\theta \mathbb{E}_{a,b,t,n} \left[\|F_\theta(b_t + \epsilon_t n,\, t) - b\|_1 \right],$$

where $\epsilon_t n$ is Gaussian noise. At inference, the estimate is updated recursively using the learned conditional mean, gradually solving a sequence of easier inverse problems from a to b.

4 Proposed Method

In this section, we present a detailed explanation of our proposed methods, including a comprehensive description of the iterative plug-and-play framework.

4.1 Iterative Implicit Neural Representations: I-INR

I-INR reconstructs the signal over multiple steps, progressively improving the quality at each iteration. As shown in Figure 2b, the process starts from an initial state $g(x)_1 = \mathcal{Z}$ at $t = 1$ and gradually approaches the final reconstruction $g(x)_0 = \mathcal{I}(x)$ at $t = 0$, using steps of size δ. This is achieved by inverting the forward process:

$$g(x)_t = \mathcal{I}(x)(1-t) + \mathcal{Z}t, \qquad t \in [0, 1], \tag{3}$$

where $\mathcal{Z}$ is sampled from a known distribution matching the shape of $\mathcal{I}(x)$.

At each step, I-INR estimates the next state by computing the conditional expectation $\mathbb{E}[g(x)_0 \mid g(x)_t]$, which represents the best possible prediction of the original signal given the current state. This estimate is then used to update the reconstruction as follows:

$$\begin{aligned} \hat{g}(x)_{t-\delta} &= \mathbb{E}\left[g(x)_{t-\delta} \mid \hat{g}(x)_t\right] \\ &= \frac{\delta}{t}\,\mathbb{E}\left[g(x)_0 \mid \hat{g}(x)_t\right] + \left(1 - \frac{\delta}{t}\right)\hat{g}(x)_t, \end{aligned} \tag{4}$$

by repeatedly applying this update, I-INR smoothly transitions from the initial state to the reconstructed signal, leveraging the conditional expectation at each step to improve accuracy.

I–INR Training. To realize the update rule in Eq. (4), we train an implicit neural network f_θ (see Figure. 2a) that maps the intermediate state $\tilde{g}(x)_t$, the spatial coordinates x, and the time index t directly to the clean target $\mathcal{I}(x)$:

$$\min_\theta \mathbb{E}_{x,t,n} \left\| f_\theta(\tilde{g}(x)_t,\, x,\, t) - \mathcal{I}(x) \right\|_2^2, \tag{5}$$

$$\tilde{g}(x)_t = (1-t)\,\mathcal{I}(x) + t\,\mathcal{Z} + \varepsilon t n, \tag{6}$$

where $\mathcal{Z} \sim p(\mathcal{Z})$ and $n \sim \mathcal{N}(0, 1)$. The small perturbation $\varepsilon t n$ (Delbracio and Milanfar 2023) satisfies the regularity requirement, guaranteeing stable reconstruction during inference. The full training procedure appears in Algorithm 1.

I–INR Reconstruction. Reconstruction begins from the latent state $\mathcal{Z}$ and proceeds for $1/\delta$ iterations of Eq. 4, iteratively refining the estimate towards the final signal (Algorithm 2).

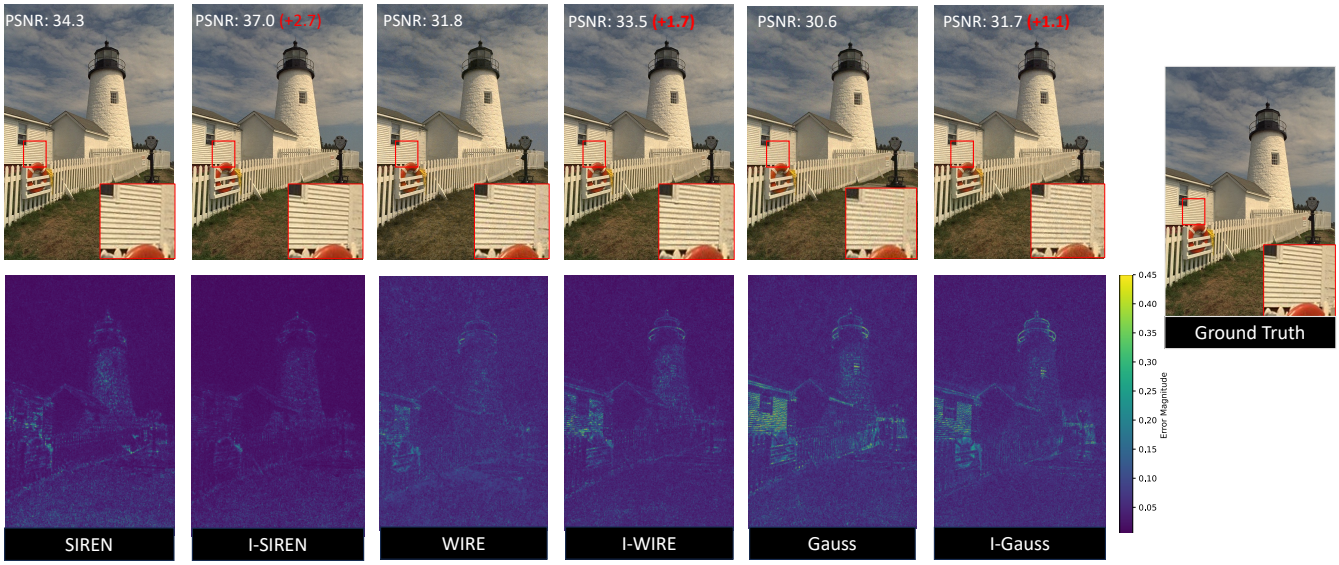


Figure 3: Image fitting results for different non-linearities and their iterative extensions (prefix "I"). Each column presents the reconstructed image (top) and residual map (bottom), highlighting reconstruction quality. PSNR values are reported, with iterative improvements in red. The rightmost column shows the ground truth.

Algorithm 1: I-INRs Model Training of f_θ

Require: $\mathcal{I}(x), \varepsilon, \eta$

$\mathcal{Z} \sim p(\mathcal{Z})$

repeat

  $x \sim p(x)$

  $t \sim \mathcal{U}(0,1),\ n \sim \mathcal{N}(0, I)$

  $\tilde{g}(x)_t \leftarrow (1-t)\,\mathcal{I}(x) + t\,\mathcal{Z} + \varepsilon\, t\, n$

  $\theta \leftarrow \theta - \eta\, \nabla_\theta \left\| f_\theta(\tilde{g}(x)_t,\, x,\, t) - \mathcal{I}(x) \right\|_2^2$

until converged

Algorithm 2: I-INRs Model Reconstruction

Require: $\mathcal{Z}$, f_θ, δ, x

$\hat{g}(x)_1 \leftarrow \mathcal{Z}$

for $t \leftarrow 1$ **to** 0 **step** $-\delta$ **do**

  $\hat{g}(x)_{t-\delta} \leftarrow \frac{\delta}{t}\, f_\theta\big(\hat{g}(x)_t, x, t\big) + \left(1 - \frac{\delta}{t}\right) \hat{g}(x)_t$

end for

return $\hat{g}(x)_0$

Network Architecture. Figure 2a provides an overview of our plug-and-play I-INR framework, which comprises three main components: a Backbone network, a FeedbackNet module, and a FuseNet module. The Backbone extracts initial features from the input; the FeedbackNet incorporates the intermediate state and time conditioning; and the FuseNet merges these feature streams. The final output is given by Eq. (7). This design enables flexible integration with existing INR architectures and supports efficient iterative refinement. The information flow between components is formalized as

$$f_\theta\big(\hat{g}(x)_t,\, x,\, t\big) = \text{FuseNet}\big(\text{concat}(\mathbf{f}, \mathbf{b})\big) \odot \mathbf{b}, \qquad (7)$$

where

$$\mathbf{b} := \text{Backbone}(x), \qquad \mathbf{f} := \text{FeedbackNet}\big(\hat{g}(x)_t,\, x,\, t\big).$$

At each iteration, the current state is processed by the FeedbackNet, and its output is fused with the Backbone features via FuseNet. Notably, the Backbone is forward-passed only once throughout the reconstruction, while the lightweight FeedbackNet and FuseNet operate at each iteration, resulting in a substantial reduction in computational cost.

5 Experiments

To evaluate the effectiveness and robustness of our proposed method, we conduct extensive experiments across diverse tasks and backbones. Our evaluation encompasses regression tasks, such as 2D and 3D signal fitting, as well as generalization and robustness tasks, including image SR and image denoising. To demonstrate its comparative advantage, we benchmark our method against established baseline approaches, including SIREN, Gauss, and WIRE.

Implementation Settings. The proposed I-INR framework is implemented in PyTorch (Paszke et al. 2019) and optimized using the Adam optimizer (Diederik 2014). For all experiments, the noise parameter ϵ is empirically set to 0.1, and the inference *steps* are fixed to 2 unless stated otherwise. The initial state $\mathcal{Z}$ is sampled from a standard normal Gaussian distribution across all experiments. All baseline methods, including SIREN, WIRE, and Gauss, as well as their iterative counterparts, are initialized following the configurations specified in their respective papers. In our implementation, both FeedbackNet and FuseNet are lightweight

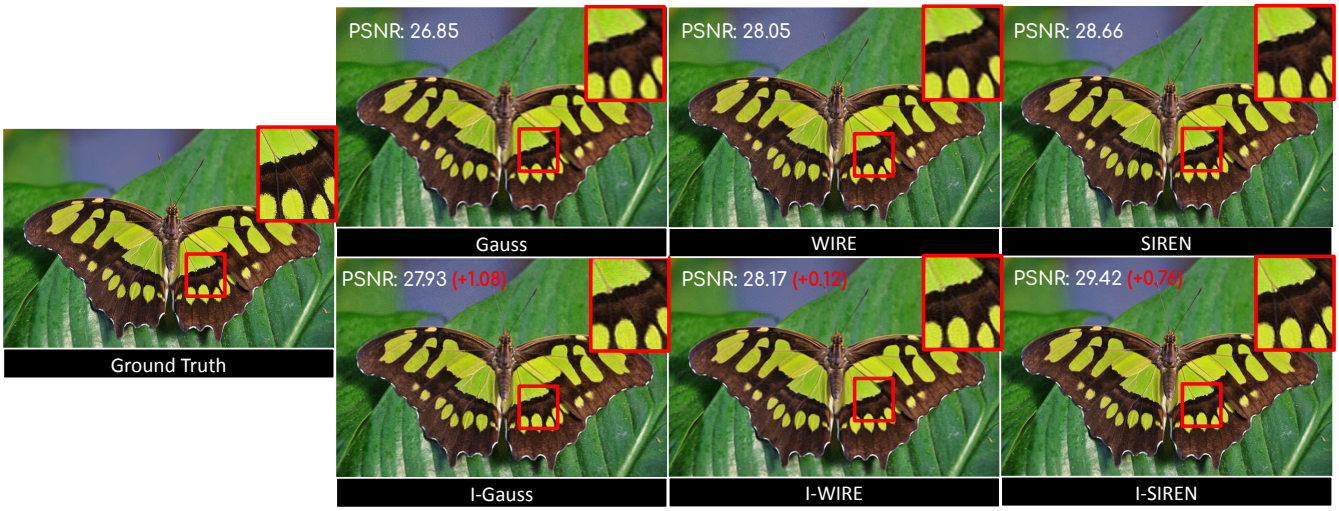


Figure 4: Visual quality comparison of super-resolution results at 2× scale using various methods. The ground truth is compared against baseline INR methods SIREN, WIRE, Gauss, and their iterative counterparts. The iterative approaches consistently achieve sharper reconstructions with fewer artifacts, effectively preserving finer details and high-frequency structures.

	SIREN		WIRE		Gauss	
	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑
Baseline	34.57	0.931	32.15	0.898	31.33	0.880
Ours	**37.53**	**0.961**	**33.73**	**0.924**	**31.93**	**0.884**

Table 1: Image fitting results on Kodak comparing SIREN, WIRE, and Gauss with their iterative counterparts.

MLPs with two layers, each layer having a width of 30 for FeedbackNet and 100 for FuseNet.

Evaluation Metrics. To comprehensively evaluate our proposed method, we conduct extensive experiments using both distortion-based metrics, such as Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM) (Wang et al. 2004), as well as perceptual metrics like Learned Perceptual Image Patch Similarity (LPIPS) (Zhang et al. 2018). For a detailed analysis, refer to the supplementary materials.

5.1 Results

Our results consistently demonstrate that I-INR outperforms existing methods across diverse tasks, including image fitting, SR, and 3D occupancy. It achieves superior reconstruction quality and enhanced robustness compared to baseline approaches (SIREN, WIRE, Gauss).

Image Fitting. For image fitting, each baseline employs a 3-layer MLP with 300 neurons per layer, serving as the Backbone INR for iterative models (Figure 2a). All networks are trained on the full-resolution Kodak dataset (Kodak 1993), with average results reported in Table 1. The proposed iterative models consistently outperform their one-shot counterparts. As shown in Figure 3, error maps from baseline models display uniform noise, especially in high-frequency regions, whereas iterative variants produce smoother reconstructions with reduced errors, including in challenging areas.

Scale	Method	SIREN		WIRE		Gauss	
		PSNR ↑	LPIPS ↓	PSNR ↑	LPIPS ↓	PSNR ↑	LPIPS ↓
2×	Baseline	26.77	0.414	26.14	0.457	25.19	0.538
	Ours	**27.64**	**0.367**	**27.21**	**0.388**	**26.82**	**0.363**
4×	Baseline	25.03	0.597	24.57	0.618	23.85	0.673
	Ours	**25.53**	**0.575**	**25.78**	**0.496**	**25.18**	**0.620**

Table 2: Comparison of (SR) performance for 2× and 4× upscaling using a model trained on 2× SR. Results are shown alongside SIREN, WIRE, and Gauss, including their iterative counterparts. The best results are shown in bold.

Image Super-resolution. We evaluate our I-INR framework against traditional one-shot INRs on SR tasks using 40 images from the DIV2K dataset (Agustsson and Timofte 2017). The model is trained exclusively on 2× SR with a two-layer MLP of 256 neurons per layer. To assess generalization, we evaluate I-INR on both 2× and 4× SR, despite training only on 2×. Table 2 compares its performance with single-shot baselines, demonstrating that our approach consistently enhances fidelity (PSNR) and perceptual quality (LPIPS) across all architectures. Figure 4 visualizes the butterfly image for 2× SR (refer to supplementary for visualization at 4×), comparing each baseline with its iterative counterpart. The proposed approach consistently outperforms baseline methods, generating sharper reconstructions with fewer artifacts across all non-linearities.

Image Denoising. To assess the robustness of I-INRs in modeling noisy signals, we utilize 40 high-resolution color images from the DIV2K dataset and introduce Poisson-distributed noise to each pixel, with a maximum mean photon count of 30 and a readout noise level of 2, following

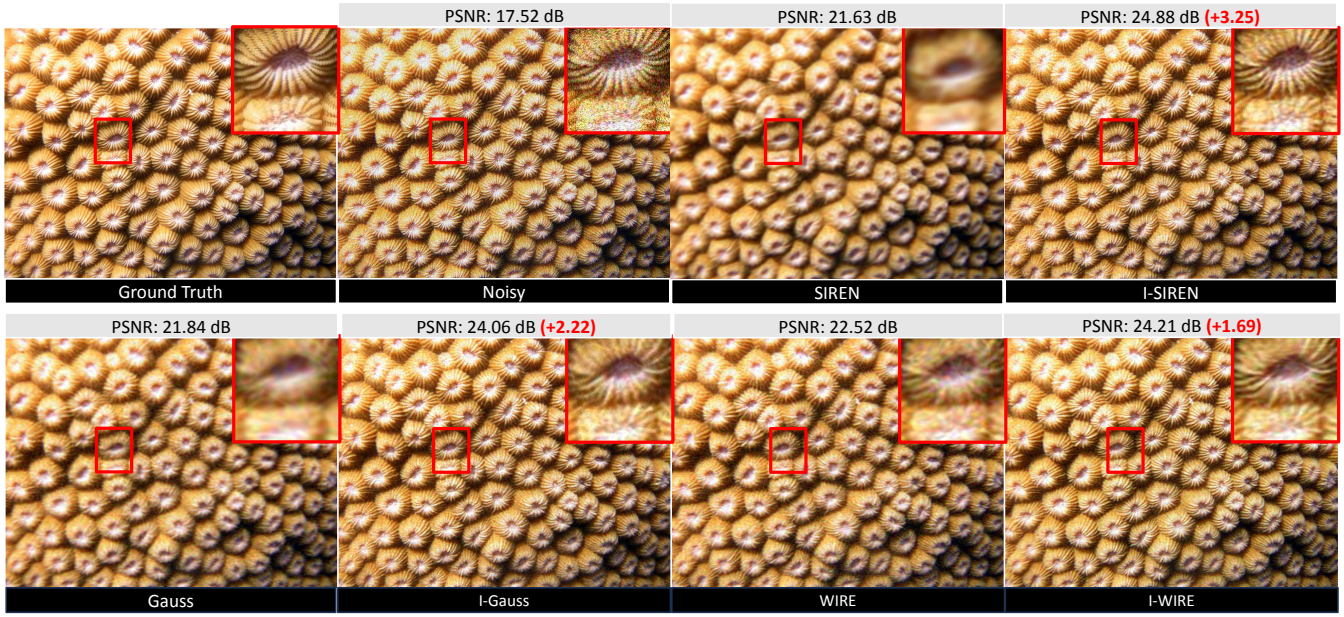


Figure 5: Visual comparison of denoising results using various methods. The ground truth is compared against noisy input, baseline methods SIREN, WIRE, and Gauss, as well as their iterative counterparts. I-INR demonstrates superior artifact reduction and detail preservation compared to their non-iterative counterparts.

	SIREN		WIRE		Gauss	
	PSNR ↑	LPIPS ↓	PSNR ↑	LPIPS ↓	PSNR ↑	LPIPS ↓
Baseline	23.86	0.604	23.32	0.746	23.10	0.783
Ours	**25.59**	**0.540**	**24.76**	**0.490**	**24.20**	**0.533**

Table 3: Image denoising results on 40 images sampled from the DIV2K dataset, comparing SIREN, WIRE, and Gauss with their iterative counterparts.

	SIREN ↑	WIRE ↑	Gauss ↑
Baseline	0.9840	0.9917	0.9855
Ours	**0.9934**	**0.9950**	**0.9967**

Table 4: Averaged Intersection over Union (IoU) results comparing SIREN, WIRE, and Gauss with their iterative counterparts on the Armadillo, Dragon, and Thai Statue 3D.

(Saragadam et al. 2023). The same architecture and training procedure as SR is employed. Table 3 presents quantitative results comparing the denoising performance of baseline architectures with their iterative counterparts. Iterative models consistently achieve significant improvements in both PSNR and LPIPS over their single-shot baselines. Figure 5 provides a qualitative comparison of denoising performance for SIREN, WIRE, and Gauss alongside their iterative versions. The iterative approach significantly enhances reconstruction quality across all architectures, achieving up to +3.25 dB PSNR improvement (for SIREN).

3D Occupancy. The 3D occupancy task aims to reconstruct 3D shapes by modeling the occupancy of points in space. We conduct experiments on the Armadillo, Dragon, and Thai Statue 3D scenes (Gal and Cohen-Or 2006), utilizing a two-layer MLP with 256 neurons per layer for both the baseline INRs and their iterative counterparts. Table 4 presents the averaged Intersection over Union (IoU) results, while Figure 6 visualizes the reconstructions achieved using different nonlinearities and their iterative variants. Notably, I-INRs outperform their single-shot baselines, generating sharper and more precise reconstructions. The iterative approach enhances detail preservation, particularly in complex regions as shown in Figure 6.

5.2 Ablation Studies

We ablate inference complexity, Feedback/FuseNet impact, and reconstruction steps. Further analysis on initial state, training cost, layer depth, and statistical significance is included in the supplementary material.

Inference Complexity. During reconstruction, I-SIREN executes the Backbone network only once, while subsequent refinement steps involve forward passes through the lightweight FeedbackNet and FuseNet modules. The iterative setup introduces minimal computational overhead: the Backbone accounts for 106.8 GFLOPs, and each refinement step adds just 0.43 GFLOPs. For two refinement steps (steps $= 2$), the total computational cost amounts to 107.6 GFLOPs, approximately equivalent to the original SIREN.

Impact of FeedbackNet and FuseNet: To evaluate the impact of architectural components of I-INR, we performed

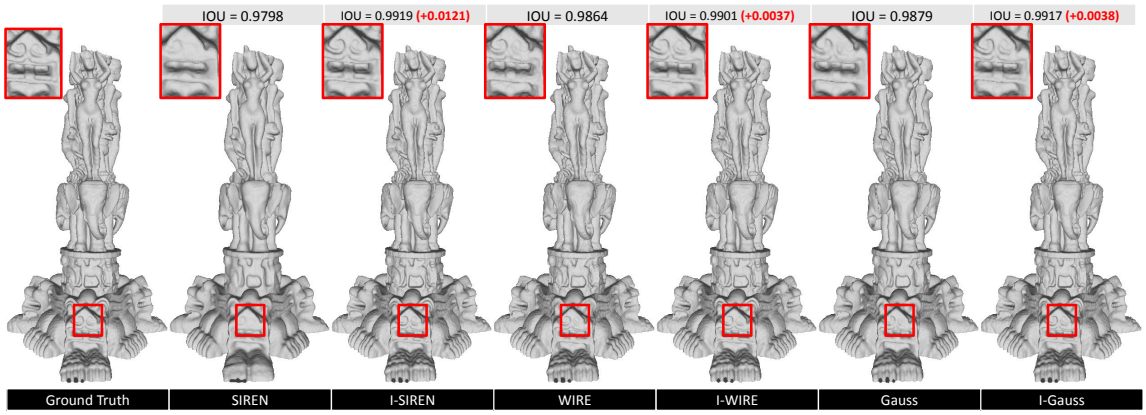
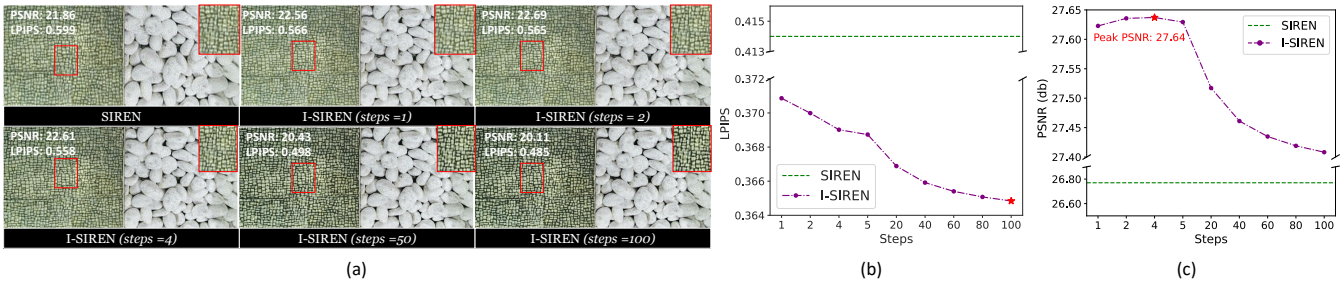


Figure 6: Visualization of 3D occupancy reconstruction results using various methods and their iterative counterparts. The ground truth is compared to baseline models SIREN, WIRE, Gauss, and their iterative counterparts.



(a) (b) (c)

Figure 7: Analysis of I-SIREN's performance for super-resolution at 2× resolution across different iteration $steps$. (a) I-SIREN achieves peak PSNR at $steps = 4$, where further increasing time steps enhances perceptual quality but reduces fidelity due to the fidelity-perception tradeoff. (b) Increasing the number of steps consistently improves perceptual quality. (c) PSNR reaches its highest performance at $steps = 4$, and while additional iterations slightly reduce PSNR, it remains superior to the baseline.

FeedbackNet (Params)	FuseNet (Params)	PSNR
✗	✗	20.27
✓ (1×)	✗	31.88
✓ (1×)	✓ (1×)	37.53
✓ (2×)	✓ (1×)	37.25
✓ (1×)	✓ (2×)	37.77
✓ (2×)	✓ (2×)	37.56

Table 5: Impact of FeedbackNet and FuseNet on image fitting task with I-SIREN on the Kodak dataset.

ablation studies by selectively disabling the FeedbackNet and FuseNet modules and varying their parameter scales. These experiments were conducted on the image fitting task using I-SIREN on the Kodak dataset. As summarized in Table 5, removing both modules leads to a significant performance drop, with PSNR falling to 20.27. Introducing FeedbackNet alone substantially improves reconstruction quality, and the addition of FuseNet provides further gains. The best performance is achieved when both modules are included, with moderate parameter scaling yielding an optimal balance between computational cost and accuracy.

Analysis of Reconstruction Steps. We evaluate the effect of reconstruction steps on 2× super-resolution using 40 images from the DIV2K dataset (Figures 7b and 7c). PSNR peaks at $steps = 4$, with the largest improvement from 1 to 2 steps; beyond 2, gains saturate. While PSNR slightly drops for $steps > 4$, perceptual quality (LPIPS) continues to improve, reflecting the Perception-Distortion Tradeoff (Blau and Michaeli 2018). Figure 7a visually compares SIREN and I-SIREN across steps, showing that I-SIREN produces increasingly vivid and detailed reconstructions, confirming that additional refinement steps are beneficial for capturing finer textures and high-frequency components. This trend generalizes to other tasks, as discussed in the supplementary material.

6 Conclusion

This work introduces a novel I-INR framework that progressively reconstructs a signal. The proposed method serves as a plug-and-play solution compatible with existing INR methods, significantly enhancing their performance. Through extensive experiments, we demonstrate that I-INR effectively captures high-frequency details, improves noise robustness, and achieves superior reconstruction quality across various tasks and nonlinearities.

References

Agustsson, E.; and Timofte, R. 2017. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 126–135.

Blau, Y.; and Michaeli, T. 2018. The perception-distortion tradeoff. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 6228–6237.

Cha, J.; Haider, A.; Yang, S.; Jin, H.; Yang, S.; Uddin, A. S.; Kim, J.; Kim, S. Y.; and Bae, S.-H. 2024. Descanning: From scanned to the original images with a color correction diffusion model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 954–963.

Chen, Y.; Liu, S.; and Wang, X. 2021. Learning continuous image representation with local implicit image function. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 8628–8638.

Chen, Y.; Wang, O.; Zhang, R.; Shechtman, E.; Wang, X.; and Gharbi, M. 2024. Image neural field diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8007–8017.

Chu, J.; Du, C.; Lin, X.; Zhang, X.; Wang, L.; Zhang, Y.; and Wei, H. 2025. Highly accelerated MRI via implicit neural representation guided posterior sampling of diffusion models. *Medical Image Analysis*, 100: 103398.

Delbracio, M.; and Milanfar, P. 2023. Inversion by direct iteration: An alternative to denoising diffusion for image restoration. *Transactions on Machine Learning Research*.

Diederik, P. K. 2014. Adam: A method for stochastic optimization. *(No Title)*.

Fathony, R.; Sahu, A. K.; Willmott, D.; and Kolter, J. Z. 2020. Multiplicative filter networks. In *International Conference on Learning Representations*.

Gal, R.; and Cohen-Or, D. 2006. Salient geometric features for partial shape matching and similarity. *ACM Transactions on Graphics (TOG)*, 25(1): 130–150.

Gao, R.; and Jaiman, R. K. 2024. H-SIREN: Improving implicit neural representations with hyperbolic periodic functions. *arXiv preprint arXiv:2410.04716*.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33: 6840–6851.

Hui, M.; Wei, Z.; Zhu, H.; Xia, F.; and Zhou, Y. 2024. Microdiffusion: Implicit representation-guided diffusion for 3D reconstruction from limited 2D microscopy projections. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11460–11469.

Irshad, M. Z.; Zakharov, S.; Liu, K.; Guizilini, V.; Kollar, T.; Gaidon, A.; Kira, Z.; and Ambrus, R. 2023. NeO 360: Neural Fields for Sparse View Synthesis of Outdoor Scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 9187–9198.

Ji, S.; Luo, H.; Chen, X.; Tu, Y.; Wang, Y.; and Zhao, H. 2025. LayerFlow: A Unified Model for Layer-aware Video Generation. *arXiv preprint arXiv:2506.04228*.

Jiang, S.; Hua, J.; and Han, Z. 2023. Coordinate quantized neural implicit representations for multi-view reconstruction. In *Proceedings of the IEEE/CVF international conference on computer vision*, 18358–18369.

Kodak, E. 1993. Kodak lossless true color image suite (PhotoCD PCD0992).

Liu, R.; Wu, R.; Van Hoorick, B.; Tokmakov, P.; Zakharov, S.; and Vondrick, C. 2023a. Zero-1-to-3: Zero-shot one image to 3d object. In *Proceedings of the IEEE/CVF international conference on computer vision*, 9298–9309.

Liu, Z.; Zhu, H.; Zhang, Q.; Fu, J.; Deng, W.; Ma, Z.; Guo, Y.; and Cao, X. 2023b. FINER: Flexible spectral-bias tuning in Implicit NEural Representation by Variable-periodic Activation Functions. *arXiv preprint arXiv:2312.02434*.

Mildenhall, B.; Srinivasan, P. P.; Tancik, M.; Barron, J. T.; Ramamoorthi, R.; and Ng, R. 2021. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1): 99–106.

Nie, S.; Zhu, F.; You, Z.; Zhang, X.; Ou, J.; Hu, J.; Zhou, J.; Lin, Y.; Wen, J.-R.; and Li, C. 2025. Large language diffusion models. *arXiv preprint arXiv:2502.09992*.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.

Peng, S.; Niemeyer, M.; Mescheder, L.; Pollefeys, M.; and Geiger, A. 2020. Convolutional occupancy networks. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, 523–540. Springer.

Qian, G.; Mai, J.; Hamdi, A.; Ren, J.; Siarohin, A.; Li, B.; Lee, H.-Y.; Skorokhodov, I.; Wonka, P.; Tulyakov, S.; et al. 2023. Magic123: One image to high-quality 3d object generation using both 2d and 3d diffusion priors. *arXiv preprint arXiv:2306.17843*.

Raghavan, N.; Xiao, Y.; Lin, K.-E.; Sun, T.; Bi, S.; Xu, Z.; Li, T.-M.; and Ramamoorthi, R. 2023. Neural Free-Viewpoint Relighting for Glossy Indirect Illumination. In *Computer Graphics Forum*, volume 42, e14885. Wiley Online Library.

Rahaman, N.; Baratin, A.; Arpit, D.; Draxler, F.; Lin, M.; Hamprecht, F.; Bengio, Y.; and Courville, A. 2019. On the spectral bias of neural networks. In *International conference on machine learning*, 5301–5310. PMLR.

Ramasinghe, S.; and Lucey, S. 2022. Beyond periodicity: Towards a unifying framework for activations in coordinate-mlps. In *European Conference on Computer Vision*, 142–158. Springer.

Reed, A.; Blanford, T.; Brown, D. C.; and Jayasuriya, S. 2021. Implicit neural representations for deconvolving SAS images. In *OCEANS 2021: San Diego–Porto*, 1–7. IEEE.

Rissanen, S.; Heinonen, M.; and Solin, A. 2022. Generative modelling with inverse heat dissipation. *arXiv preprint arXiv:2206.13397*.

Saharia, C.; Ho, J.; Chan, W.; Salimans, T.; Fleet, D. J.; and Norouzi, M. 2022. Image super-resolution via iterative refinement. *IEEE transactions on pattern analysis and machine intelligence*, 45(4): 4713–4726.

Saragadam, V.; LeJeune, D.; Tan, J.; Balakrishnan, G.; Veeraraghavan, A.; and Baraniuk, R. G. 2023. Wire: Wavelet implicit neural representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 18507–18516.

Saragadam, V.; Tan, J.; Balakrishnan, G.; Baraniuk, R. G.; and Veeraraghavan, A. 2022. Miner: Multiscale implicit neural representation. In *European Conference on Computer Vision*, 318–333. Springer.

Shen, S.; Wang, Z.; Liu, P.; Pan, Z.; Li, R.; Gao, T.; Li, S.; and Yu, J. 2021. Non-line-of-sight imaging via neural transient fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(7): 2257–2268.

Singh, R.; Shukla, A.; and Turaga, P. 2023. Polynomial implicit neural representations for large diverse datasets. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2041–2051.

Sitzmann, V.; Martel, J.; Bergman, A.; Lindell, D.; and Wetzstein, G. 2020. Implicit neural representations with periodic activation functions. *Advances in neural information processing systems*, 33: 7462–7473.

Srinivasan, P. P.; Deng, B.; Zhang, X.; Tancik, M.; Mildenhall, B.; and Barron, J. T. 2021. Nerv: Neural reflectance and visibility fields for relighting and view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7495–7504.

Sun, Y.; Liu, J.; Xie, M.; Wohlberg, B.; and Kamilov, U. S. 2021. Coil: Coordinate-based internal learning for imaging inverse problems. *arXiv preprint arXiv:2102.05181*.

Tancik, M.; Srinivasan, P.; Mildenhall, B.; Fridovich-Keil, S.; Raghavan, N.; Singhal, U.; Ramamoorthi, R.; Barron, J.; and Ng, R. 2020. Fourier features let networks learn high frequency functions in low dimensional domains. *Advances in neural information processing systems*, 33: 7537–7547.

Thennakoon, P.; Ranasinghe, A.; De Silva, M.; Epakanda, B.; Godaliyadda, R.; Ekanayake, P.; and Herath, V. 2025. BandRC: Band Shifted Raised Cosine Activated Implicit Neural Representations. *arXiv preprint arXiv:2505.11640*.

Wang, T.; Yang, X.; Xu, K.; Chen, S.; Zhang, Q.; and Lau, R. W. 2019. Spatial attentive single-image deraining with a high quality real rain dataset. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 12270–12279.

Wang, Z.; Bovik, A. C.; Sheikh, H. R.; and Simoncelli, E. P. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4): 600–612.

Xie, S.; Zhu, H.; Liu, Z.; Zhang, Q.; Zhou, Y.; Cao, X.; and Ma, Z. 2023. Diner: Disorder-invariant implicit neural representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6143–6152.

Zhang, R.; Isola, P.; Efros, A. A.; Shechtman, E.; and Wang, O. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 586–595.

Zhong, E. D.; Bepler, T.; Davis, J. H.; and Berger, B. 2019. Reconstructing continuous distributions of 3D protein structure from cryo-EM images. *arXiv preprint arXiv:1909.05215*.

7 Additional Results

7.1 Additional Quantative Results

Table 6 presents a detailed comparison of our proposed method against the baseline across image fitting, denoising, and SR at $2\times$ for all three evaluation metrics (PSNR, SSIM, and LPIPS). Our iterative approach consistently enhances performance across tasks and different baselines, achieving superior reconstruction quality.

7.2 Additional Qualitative Comparison

We also present an additional qualitative comparison of our proposed method against SIREN, Gauss, and WIRE. Figures 8, 9, 10, 11, illustrates a comparative analysis for image fitting, SR at $2\times$ and $4\times$ scales, and image denoising, demonstrating the improvements achieved by our approach over baseline INRs. Our proposed method consistently outperforms single-shot baselines, demonstrating superior reconstruction quality across various models and tasks.

8 Additional Ablation Studies

Initial State. I-INR reconstructs a signal iteratively over multiple steps, starting from an initial state $\mathcal{Z}$. The choice of this initial state can significantly impact the reconstruction process. To assess its effect, we conduct extensive experiments with different types of initial states. Empirically, $\mathcal{Z}$ sampled from a standard normal Gaussian distribution $\mathcal{N}(0, 1)$ consistently outperforms all-zeros and all-ones initial states across different non-linearities for the image fitting task. The results, summarized in Table 7, highlight the superiority of noise-based initial states in achieving higher PSNR and SSIM values.

Training Complexity. Our proposed I-INR method achieves faster convergence than the non-iterative baseline while maintaining similar training complexity, as shown in Figure 12a. The results are averaged over the Kodak dataset for the image fitting task. Figure 12b further illustrates I-SIREN's training performance under different learning rates.

Number of Layers in Backbone Network. For the image fitting task on the Kodak dataset, we investigate the impact of increasing the number of layers in SIREN and its iterative counterpart, I-SIREN. The results, presented in Table 8, show that while the performance of SIREN saturates beyond a certain depth, specifically at five layers, I-SIREN continues to improve as more layers are added. Notably, a three-layer I-SIREN outperforms a five-layer SIREN, demonstrating the efficiency of the iterative approach.

Additional Reconstruction Steps Analysis. For SR at $2\times$, we further analyze the reconstruction quality of I-SIREN over multiple steps, comparing it to SIREN and the ground truth. As shown in Figure 13, the results demonstrate that with an increasing number of steps, details become more pronounced, and more high-frequency information is incorporated into the reconstruction.

Additionally, We analyze the impact of reconstruction steps on image fitting using the Kodak dataset (see Figure 14). Our findings indicate that the model achieves its highest PSNR at $steps = 4$, with the most significant gain occurring between $steps = 1$ and $steps = 2$. Beyond $steps = 2$, PSNR begins to plateau. Notably, even though PSNR decreases for $steps > 4$, perceptual quality continues to improve.

We also analyze the effect of reconstruction steps on image denoising performance using 40 images from the DIV2K dataset (see Figure 15). Our findings indicate that PSNR peaks at $steps = 2$ before gradually declining as $steps$ increases. Meanwhile, LPIPS improves until $step = 4$ and stabilizes around step 5. Beyond $steps = 4$, both PSNR and LPIPS degrade as I-INR begins reconstructing noise present in the data. At higher iterations, the model inadvertently reconstructs these distortions, leading to a decline in performance.

Feature Fusion. We conduct ablation studies on the final feature-fusion mechanism for signal reconstruction, evaluating two strategies that combine the outputs of the *Backbone*

and *FuseNet* modules (Figure 2a of main manuscript):

$$\mathbf{b} := \text{Backbone}(x), \qquad \mathbf{f} := \text{FeedbackNet}(\hat{g}(x)_t,\, x,\, t)$$

$$\mathbf{z} := \text{FuseNet}(\text{concat}(\mathbf{f},\, \mathbf{b}))$$

$$\text{Multiplicative:} \quad out = \mathbf{b} \odot \mathbf{z},$$

$$\text{Adaptive:} \quad out = \mathbf{b} \cdot t + \mathbf{z} \cdot (1 - t)$$

In *Multiplicative* fusion, the final output is obtained via element-wise multiplication of the Backbone and FuseNet outputs. The *Adaptive* method, conversely, uses the interpolation factor t to smoothly shift emphasis from the Backbone output (dominant at $t = 1$) toward the FuseNet output at higher values of t. Quantitative results comparing both fusion strategies for the image-fitting task are summarized in Table 9. Multiplicative fusion consistently outperforms Adaptive fusion, as evidenced by the results. Therefore, we adopt Multiplicative fusion as the standard approach for our method.

Time Complexity. We compare SIREN and I-SIREN on the Kodak dataset, trained for 2000 iterations on an NVIDIA RTX 4090 GPU. As shown in Table 10, I-SIREN adds only a small overhead in training time ($+6\%$) and inference latency ($+3$ ms per image) while achieving higher reconstruction quality, confirming its efficiency.

Statistical Significance. All prior experiments were conducted using a single random seed. To assess the robustness and statistical significance of our approach, we additionally evaluate performance across five different seeds. Table 11 reports the results for image fitting, SR, and denoising tasks, comparing SIREN and I-SIREN, along with confidence intervals. The dataset and experimental settings remain consistent with those previously described. As shown in Table 11, I-SIREN consistently outperforms SIREN across all tasks. The improvements achieved by I-SIREN are statistically significant, with p-values < 0.00001 based on the Wilcoxon signed-rank test.

9 Training Setup

Table 12 presents the key hyperparameters for various experiments. These hyperparameters were chosen to maximize peak performance across different tasks.

Additionally, we used 2000 training iterations for image fitting, super-resolution (SR), and denoising tasks, while for object occupancy, we used 200 training iterations.

Task	Model	SIREN PSNR↑	SIREN SSIM↑	SIREN LPIPS↓	Gauss PSNR↑	Gauss SSIM↑	Gauss LPIPS↓	WIRE PSNR↑	WIRE SSIM↑	WIRE LPIPS↓
Image Fitting	Baseline	34.57	0.931	0.080	31.33	0.880	0.147	32.15	0.898	0.120
	Ours	**37.53**	**0.961**	**0.032**	**31.93**	**0.884**	**0.124**	**33.73**	**0.924**	**0.077**
Super Resolution	Baseline	26.77	0.763	0.414	25.19	0.709	0.538	26.14	0.749	0.457
	Ours	**27.64**	**0.783**	**0.367**	**26.82**	**0.761**	**0.363**	**27.21**	**0.778**	**0.388**
Denoising	Baseline	23.86	**0.658**	0.604	23.09	0.604	0.783	23.32	0.596	0.746
	Ours	**25.59**	0.619	**0.540**	**24.20**	**0.612**	**0.533**	**24.76**	**0.645**	**0.490**

Table 6: Experimental results for Denoising, Image Fitting, and Super Resolution for **Baseline** model with **Ours**, evaluating their performance across three tasks. The best values in each column are highlighted in bold.

	I-SIREN PSNR ↑	I-SIREN SSIM ↑	I-WIRE PSNR ↑	I-WIRE SSIM ↑	I-Gauss PSNR ↑	I-Gauss SSIM ↑
Noise	**37.53**	**0.961**	**33.73**	**0.924**	**31.93**	**0.884**
Ones	37.42	0.961	33.33	0.921	30.96	0.867
Zeros	37.47	0.961	33.39	0.921	31.22	0.882

Table 7: Impact of different initial states ($\mathcal{Z}$) on image fitting task using the I-INR framework on Kodak dataset.

Layers	I-SIREN PSNR	I-SIREN Params (k)	SIREN PSNR	SIREN Params (k)
3	37.53	274	34.62	273
4	38.71	364	36.60	363
5	39.01	455	36.49	453

Table 8: Impact of varying Backbone network layer depths on the image fitting task using SIREN and I-SIREN on the Kodak dataset.

Fusion Method	I-SIREN PSNR↑	I-SIREN SSIM↑	I-SIREN LPIPS↓	I-WIRE PSNR↑	I-WIRE SSIM↑	I-WIRE LPIPS↓	I-Gauss PSNR↑	I-Gauss SSIM↑	I-Gauss LPIPS↓
Multiplicative	**37.53**	**0.961**	**0.032**	**33.73**	**0.924**	**0.077**	**31.93**	**0.884**	**0.124**
Adaptive	37.01	0.960	0.037	33.63	0.923	0.078	31.80	0.883	0.129

Table 9: Comparison of fusion methods for the image-fitting task at $steps = 2$ on the Kodak dataset with Gaussian initial state ($\mathcal{Z} \sim \mathcal{N}(0, 1)$).

Method	Train Time (s)	Inference (ms / image)	PSNR (dB)
SIREN	100	18.9	34.57
I-SIREN (2 steps)	106	21.9	**37.53**

Table 10: Training and inference time comparison on the Kodak dataset (RTX 4090).

Method	Image Fitting	SR	Denoising
I-SIREN	37.50 ± 0.247	27.22 ± 0.045	25.36 ± 0.044
SIREN	34.59 ± 0.027	26.77 ± 0.035	23.89 ± 0.021

Table 11: Evaluation on 5 different seeds for image fitting, super-resolution tasks, and denoising on SIREN and I-SIREN.

Task	Parameter	I-Gauss	I-SIREN	I-WIRE
Image Fitting	ω	-	52	7
	σ	18	-	13
Image SR	ω	-	30	4
	σ	11	-	10
Image Denoising	ω	-	55	10
	σ	18	-	16
3D Occupancy	ω	-	55	10
	σ	17	-	20

Table 12: Hyperparameter settings for I-INRs across different tasks.

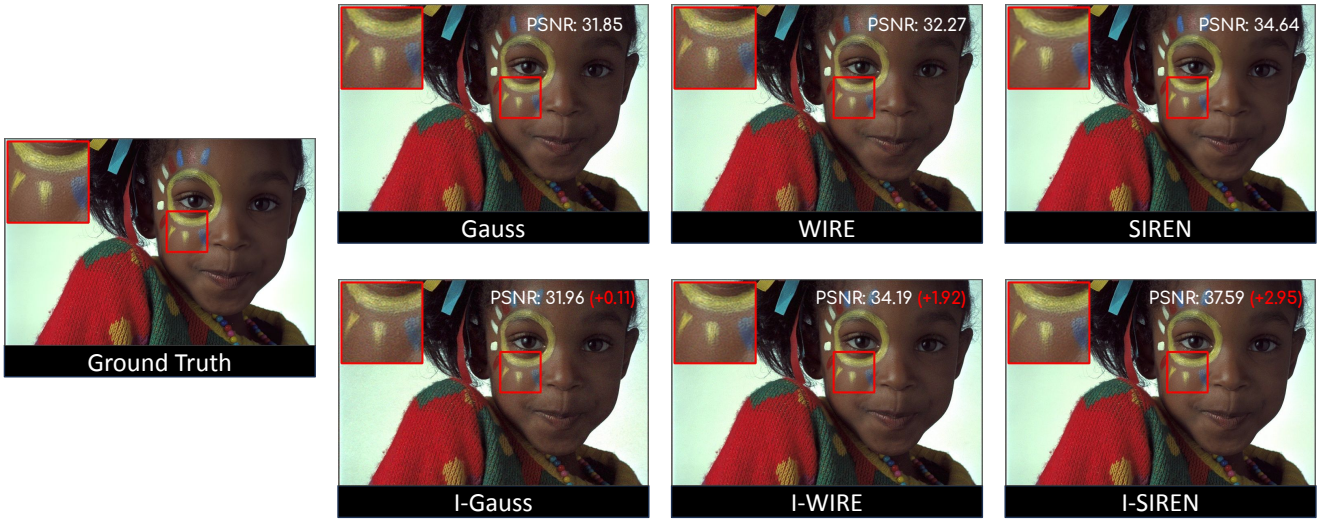


Figure 8: Image fitting results for different non-linearities and their iterative extensions (prefix "I").

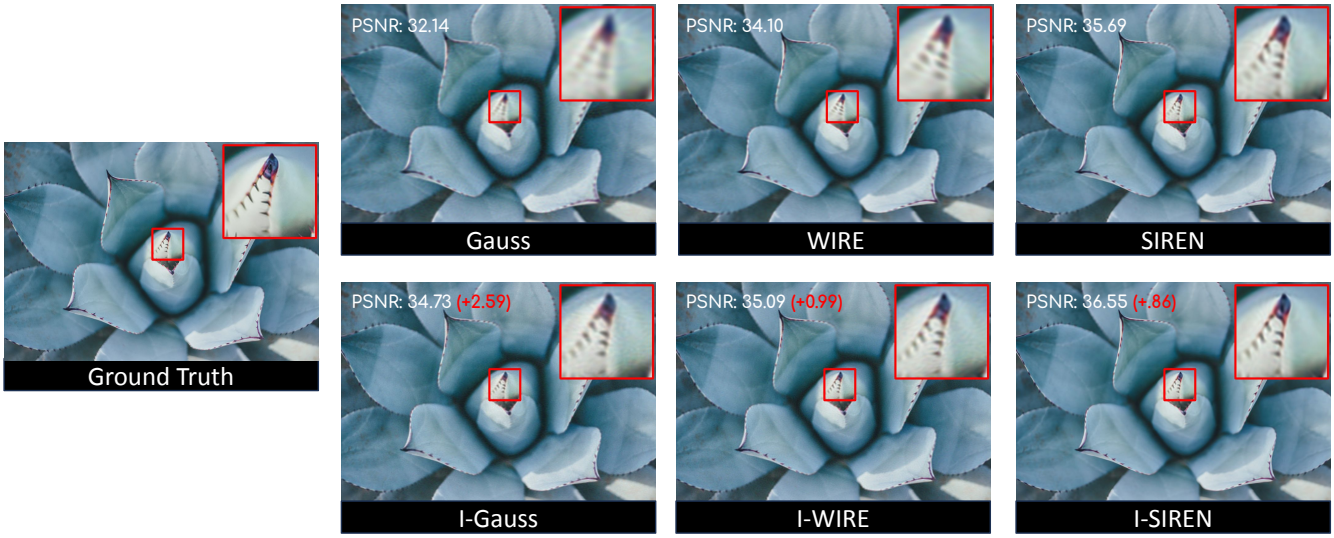


Figure 9: Visual quality comparison of super-resolution results at 2× scale using various methods. The ground truth is compared against baseline INR methods SIREN, WIRE, Gauss, and their iterative counterparts. The iterative approaches consistently achieve sharper reconstructions with fewer artifacts, effectively preserving finer details and high-frequency structures.

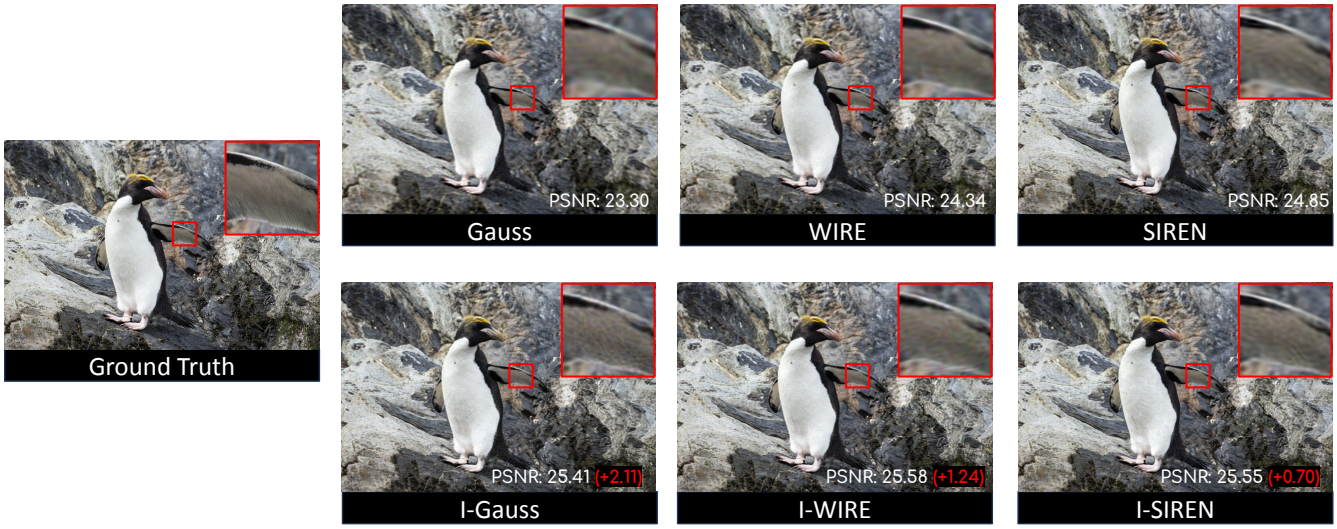


Figure 10: Visual quality comparison of super-resolution results at 4× scale using various methods. The ground truth is compared against baseline INR methods SIREN, WIRE, Gauss, and their iterative counterparts. The iterative approaches consistently achieve sharper reconstructions with fewer artifacts, effectively preserving finer details and high-frequency structures.

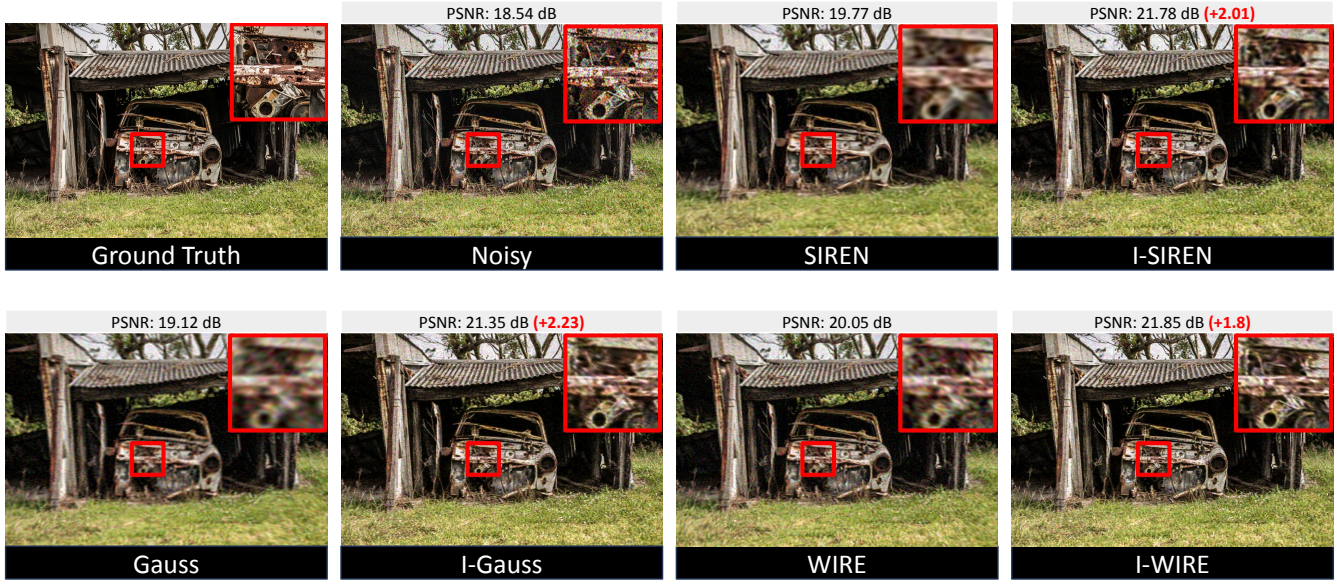
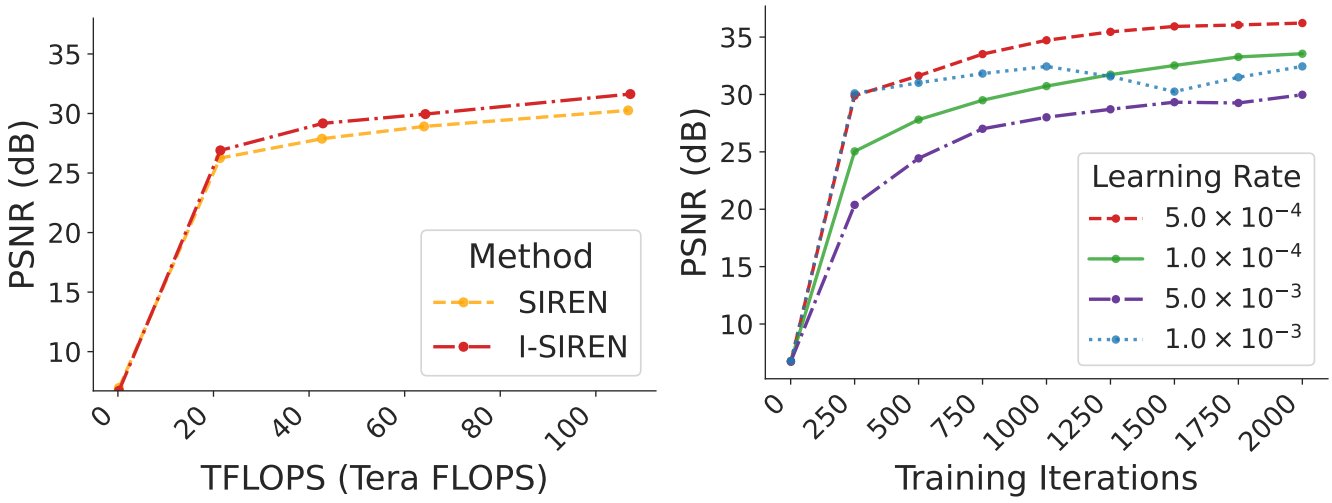


Figure 11: Visual comparison of denoising results using various methods. The ground truth is compared against noisy input, baseline methods SIREN, WIRE, and Gauss, as well as their iterative counterparts. I-INR demonstrates superior artifact reduction and detail preservation compared to their non-iterative counterparts.



(a) (b)

Figure 12: On the Kodak dataset for image fitting: (a) PSNR vs. training FLOPs comparison between SIREN and I-SIREN. (b) PSNR vs. training iterations for I-SIREN with different learning rates.

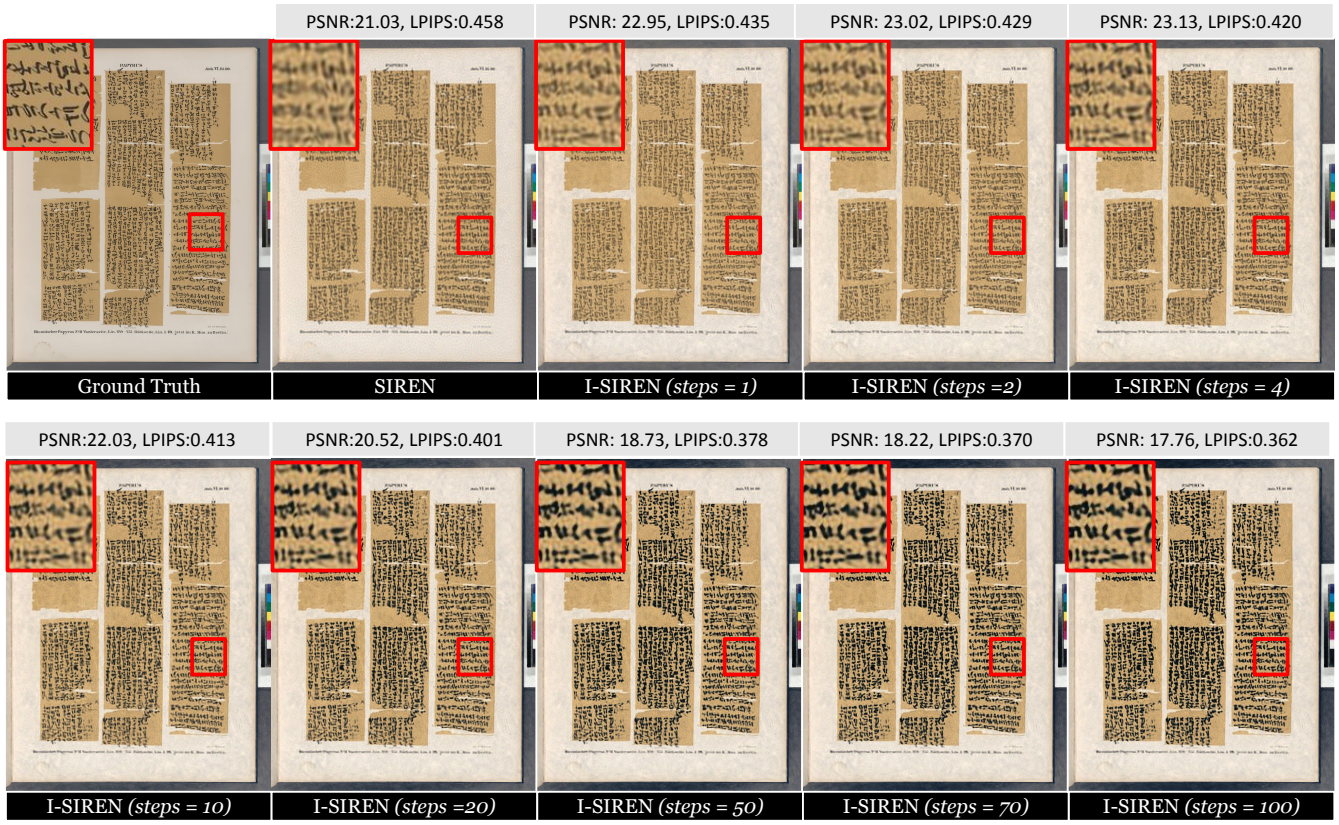
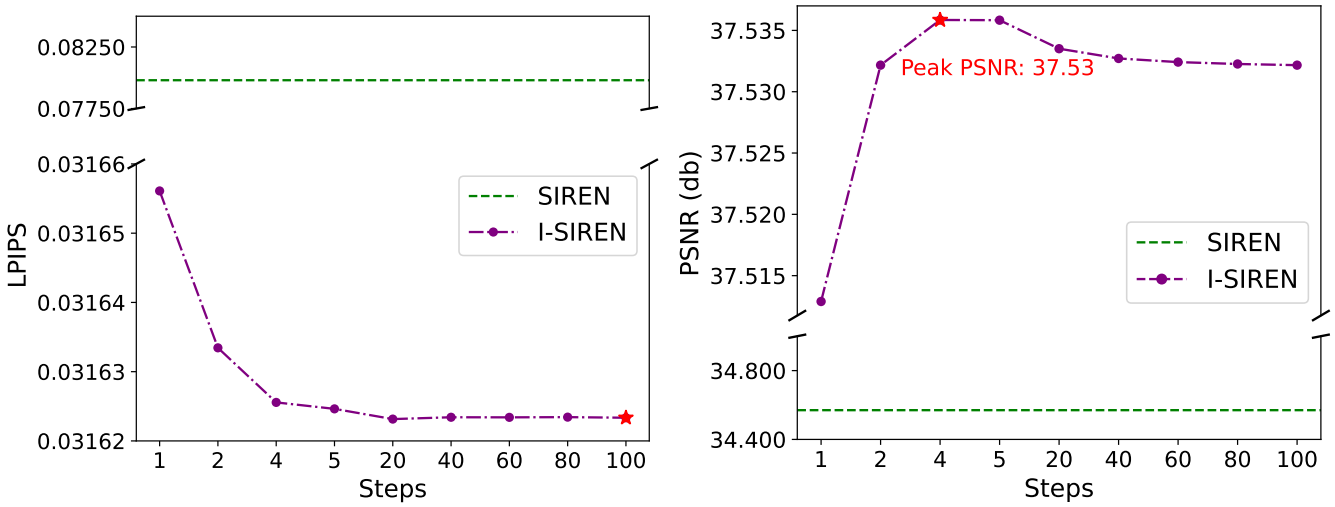
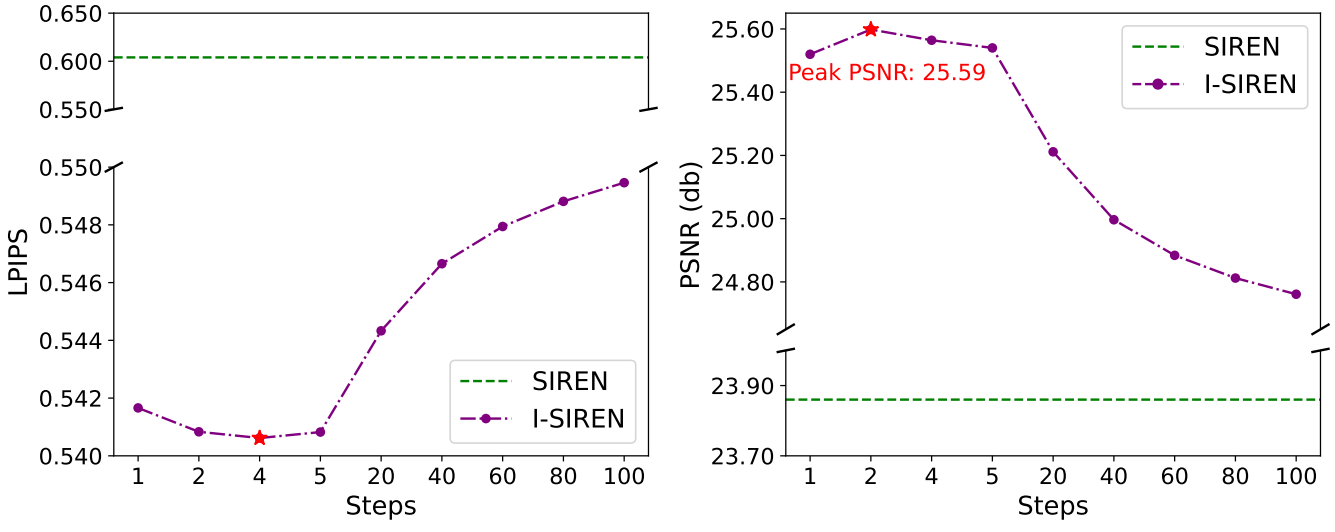


Figure 13: Analysis of I-SIREN's performance for super-resolution at 2× resolution across different steps. I-SIREN achieves peak PSNR at $steps = 4$, where further increasing steps enhances perceptual quality.



(a) (b)

Figure 14: Analysis of I-SIREN's performance for image fitting across different iteration steps. (a) Increasing the number of steps consistently improves perceptual quality. (b) PSNR reaches its highest performance at $steps = 4$, and while additional iterations slightly reduce PSNR, it remains superior to the baseline.



(a) (b)

Figure 15: Analysis of I-SIREN's performance for denoising across different reconstruction steps. (a) LPIPS improves until step 4 and stabilizes around step 5. (b) PSNR reaches its peak at $steps = 2$, but further iterations degrade performance.