

INCODE: Implicit Neural Conditioning with Prior Knowledge Embeddings

Amirhossein Kazerouni¹ Reza Azad² Alireza Hosseini³ Dorit Merhof^{4,5} Ulas Bagci⁶

¹ School of Electrical Engineering, Iran University of Science and Technology, Iran

² Institute of Imaging and Computer Vision, RWTH Aachen University, Germany

³ School of Electrical and Computer Engineering, College of Engineering, University of Tehran, Iran

⁴ Faculty of Informatics and Data Science, University of Regensburg, Germany

⁵ Fraunhofer Institute for Digital Medicine MEVIS, Bremen, Germany

⁶ Department of Radiology, Northwestern University, Chicago, USA

{amirhossein477, rezazad68}@gmail.com, {arhosseini77}@ut.ac.ir

{dorit.merhof}@ur.de, {ulas.bagci}@northwestern.edu

<https://xmindflow.github.io/incode>

Abstract

Implicit Neural Representations (INRs) have revolutionized signal representation by leveraging neural networks to provide continuous and smooth representations of complex data. However, existing INRs face limitations in capturing fine-grained details, handling noise, and adapting to diverse signal types. To address these challenges, we introduce INCODE, a novel approach that enhances the control of the sinusoidal-based activation function in INRs using deep prior knowledge. INCODE comprises a harmonizer network and a composer network, where the harmonizer network dynamically adjusts key parameters of the activation function. Through a task-specific pre-trained model, INCODE adapts the task-specific parameters to optimize the representation process. Our approach not only excels in representation, but also extends its prowess to tackle complex tasks such as audio, image, and 3D shape reconstructions, as well as intricate challenges such as neural radiance fields (NeRFs), and inverse problems, including denoising, super-resolution, inpainting, and CT reconstruction. Through comprehensive experiments, INCODE demonstrates its superiority in terms of robustness, accuracy, quality, and convergence rate, broadening the scope of signal representation. Please visit the project's website for details on the proposed method and access to the code.

1. Introduction

The realm of signal representation has undergone a significant transformation with the emergence of Implicit Neural Representations (INRs), also known as coordinate-based neural representations. Unlike traditional methods where

signal values are discretely stored on coordinate grids, this new approach revolves around training neural networks, specifically Multilayer Perceptrons (MLPs), equipped with continuous nonlinear activation functions. The goal is to approximate the complex relationship between coordinates and their corresponding signal values, ultimately providing a continuous signal representation [41].

INRs have received considerable attention for their ability to learn tasks involving complex and high-dimensional data more compactly and flexibly. They have shown promise in applications spanning computer graphics [10, 26, 28], computer vision [21, 22, 27, 50], virtual reality [6, 17], and so on. The inherent attributes of seamlessness and continuity within INRs offer a wide range of advantages, most notably in applications involving super-resolution and inpainting tasks. Unlike Convolutional Neural Networks (CNNs), INRs bypass the limitations attributed to locality biases and leverage the power of neural networks to directly learn the relationship between inputs and desired outputs, thereby enhancing their effectiveness in modeling complex tasks. However, their potential is hampered by limitations. Previous approaches have not fully exploited the high representation capacity of INRs, failing to extract fine-grained details. Additionally, these methods often disregard data noise, rendering them ineffective for tasks such as super-resolution, denoising, and inpainting. Their applicability across signal types is limited, and scalability to handle large signal sets poses difficulties. Overcoming these challenges is crucial for unlocking INRs' efficacy in diverse signal-processing contexts.

Conditional neural networks constitute a significant advancement in deep learning, endowing networks with adaptability based on auxiliary information, a departure from

conventional context-agnostic counterparts. This adaptability introduces context awareness and targeted responsiveness. The incorporation of supplementary conditions enables the accommodation of data distribution variations. In the domain of INRs, latent code concatenation with MLP spatial coordinates is prevalent [5, 24, 35]. An alternative, the dual-MLP approach by Mehta et al. [24], deploys a ReLU-based modulator network for amplitude modulation of sinusoidal activations across the hidden layers of the synthesis network. This modulation involves element-wise multiplication of modulator and synthesis activations. Shen et al. [40] augment CT and MRI reconstruction by embedding prior image data into MLP weights, initializing a reconstruction network, and facilitating its training. However, the concatenation strategy imposes limitations on reconstruction quality, the modulated synthesizer approach fails to fully exploit the potential of sinusoidal activation, the utilization of initialization techniques necessitates a two-step process, and using hyper-networks [15, 49] is computationally expensive and requires significant memory costs.

To mitigate these problems, we present a novel INR method to enhance the hierarchical representation capabilities of the INRS. The proposed method excels in achieving high-quality reconstructions across various tasks, encompassing audio, image, and 3D shape, as well as intricate challenges such as NeRF and inverse problems including denoising, super-resolution, inpainting, and CT (computed tomography) reconstruction. The architectural foundation of our proposed model is characterized by a dual-component MLP structure, comprising a *harmonizer network* and a *composer network*. The composer network is distinguished by a general form of sinusoidal activation function ($\underline{a} \sin(\underline{b}\omega_0x + \underline{c}) + \underline{d}$), which effectively establishes a mapping between spatial coordinates and their respective values. Concurrently, the harmonizer network conditions the composer with a deep prior knowledge by dynamically adjusting the parameters \underline{a} , \underline{b} , \underline{c} , and \underline{d} during the learning process. To this end, task-specific pre-trained models are used to generate object embeddings. At each learning step, the obtained embedding is fed into the harmonizer network, yielding the extraction of sinusoidal parameters. This symbiotic arrangement empowers the composer network to adeptly capture detailed information and refine intricacies crucial for accurate and comprehensive representation. Furthermore, we employ a regularization technique for the estimated parameters to expedite the convergence of the model. Our extensive experiments on various applications clearly demonstrate the superiority of our approach in terms of robustness, accuracy, quality, and convergence rate.

2. Related Works

Implicit Neural Representation. Recent works have shown remarkable success in representing various signals

using neural networks. INR applications span across several domains: 3D shape, image, video, and audio signals [7, 13, 19, 29, 37, 38, 41, 42]. Most INR works have been pursued to address the challenge of spectral bias encountered in ReLU-based MLPs, which inherently inclines towards learning low-frequency components [33]. Sitzmann et al. in [41] utilize sinusoidal-based activation functions for INR. Tanckik et al. [43] introduce an FFN that applies a Fourier feature mapping before the actual network to promote the learning of high-frequency data. Fathony et al. [9] introduce two variations of the Multiplicative Filter Networks (MFN): one employing sinusoids and another one utilizing a Gabor wavelet as the filter applied after each layer. Some works take advantage of using an aggregate of smaller networks to represent the signal rather than using one large MLP. In [12], the input signal is broken into regular grids of smaller sizes, and a separate network is responsible for representing each cell inside the grid. [23, 39] introduce an adaptive method for resource allocation based on the local complexity of the signal, enabling INR to work on larger signals, e.g., gigapixel images. Moreover, Mildenhall et al. [25] employ volume rendering to represent 3D scenes that take advantage of coordinate-based neural networks. Since vanilla NeRF is difficult to train and entails lengthy training processes, other methods [3, 4, 10, 18, 32, 48] utilized similar approaches to improve the fidelity and efficiency of NeRFs. KiloNeRF [36] shortens the rendering process by three orders of magnitude, where they utilize thousands of tiny MLPs to represent different segments of a scene and merge the outputs to obtain the entire scene. Müller et al. [28] have utilized hash encoding to expedite the training and inference process in NeRFs.

Periodic Activation Functions. In recent studies, periodic activation functions have exhibited favorable results in INR tasks by instructing the network to learn high-frequency details. Such activation functions have been widely investigated since 1987, when Lapedes and Farber [16] showed that networks with such activations are generally difficult to train. Further, Parascandolo et al. [30] shed light on why training networks with periodic activation functions is challenging. They show that training is only successful if the networks do not rely on the periodicity of the given functions and propose using a truncated sinusoidal function. Klocek et al. [15], motivated by discrete cosine transform, propose to exploit cosine activation functions for a target network whose weights are determined by a hyper-network. Recently, Sitzmann et al. [41] leverage sinusoidal activation functions initialized carefully to represent complex unstructured data. Motivated by this, we propose INCODE, a general form of sinusoidal activation function, aiming to improve the representation accuracy and robustness of SIREN.

Conditional Neural Network. In this domain, the focus on improving model adaptability through contextual informa-

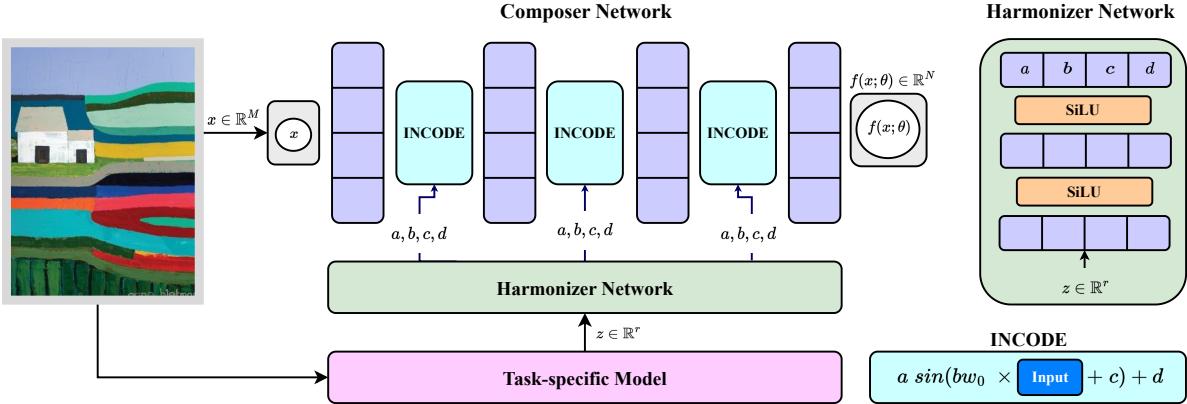


Figure 1. Illustration of the **INCODE** pipeline: adaptive implicit neural representation with prior knowledge embedding.

tion integration reflects a broader trend toward harnessing auxiliary data for enhanced model performance. In INRs, a common strategy involves the concatenation of latent codes obtained from an encoder with input coordinates [5, 31, 35]. Diverse approaches have emerged for contextual integration: Klocek et al. [15] leverage hyper-networks to compute weights for the primary network operating on coordinates, while Rebain et al. [35] propose an attention MLP conditioning mechanism using the latent code as keys and values and the coordinate as queries. Mehta et al. [24] modulate the implicit function through a modulator MLP. Consequently, we present a novel conditioning process, wherein we estimate the parameters of the proposed activation function using deep prior information and an auxiliary MLP network, thereby contributing to the growing landscape of adaptive conditional neural networks.

3. Method

The INR function operates by encoding a continuous target signal $S(x) : \mathbb{R}^M \rightarrow \mathbb{R}^N$ through a neural network $f(x; \theta) : \mathbb{R}^M \rightarrow \mathbb{R}^N$, i.e., an MLP, where the network's architecture is parameterized by a set of weights θ . This network establishes a functional mapping between input coordinates $x \in \mathbb{R}^M$ and signal values $S(x) \in \mathbb{R}^N$ (e.g., occupancy, color, etc.). This is achieved by minimizing a loss function as:

$$\arg \min_{\theta} \mathbb{E}_{x \in X} [||f(x; \theta) - S(x)||_2^2]. \quad (1)$$

By implementing $f(x; \theta)$ with ReLU-based MLP architectures, a notable trend emerges: the network displays a bias for capturing low-frequency signals. This trait, as shown by Rahaman et al. [33], frequently results in inferior-quality signal reconstructions. Sitzmann et al. [41] propose to use MLP with a sinusoidal activation function (SIREN method), where the post-activation layer is recursively defined as follows:

$$y_l = \sin(w_o (W_l y_{l-1} + b_l)), \quad l = 1, 2, \dots, L-1, \quad (2)$$

where $W_l \in \mathbb{R}^{P_{l-1} \times P_l}$ denotes the weights and $b_l \in \mathbb{R}^{P_l}$ indicates the bias at the l_{th} layer of the network. While SIREN offers superior representation capacity compared to ReLU, there is still an opportunity to enhance control over the *sine* activation function. This control could adaptively amplify representation capacity and counter noise's impact, diverging from the original SIREN's noise-equivalent treatment in image representation. Therefore, our focus in **INCODE** is on introducing a *sine*-based activation function that provides enhanced control throughout the learning process by using deep prior knowledge.

3.1. INCODE

We now present **INCODE**: a conditional INR model with prior knowledge embeddings, illustrated in Figure 1. **INCODE** is composed of two fundamental components: a *harmonizer* network and a *synthesizer* network. The harmonizer network endeavors to adjust the activation function of the composer network, while the composer network's duty is to craft a final piece. To initiate the process, we obtain a latent code $z \in \mathbb{R}^r$ from a pre-trained model tailored to the task. This latent code then serves as input for the harmonizer network, which conditions the composer network's mapping of spatial coordinates to signal values.

3.1.1 Composer Network

We define the composer network as an MLP with L hidden layers, each containing P hidden features that map the input coordinates to its output domain, e.g., RGB values for an image. Each layer within the network utilizes a periodic-nonlinear activation function, which post-activation layer can be defined as follows:

$$y_l = \underline{\mathbf{a}} \sin(\underline{\mathbf{b}} w_o (W_l y_{l-1} + b_l) + \underline{\mathbf{c}}) + \underline{\mathbf{d}}, \quad (3)$$

where values of $\underline{\mathbf{a}}$, $\underline{\mathbf{b}}$, $\underline{\mathbf{c}}$, and $\underline{\mathbf{d}}$ are adaptively determined at each iteration throughout the learning process, facilitated by the harmonizer network. In this function, each variable

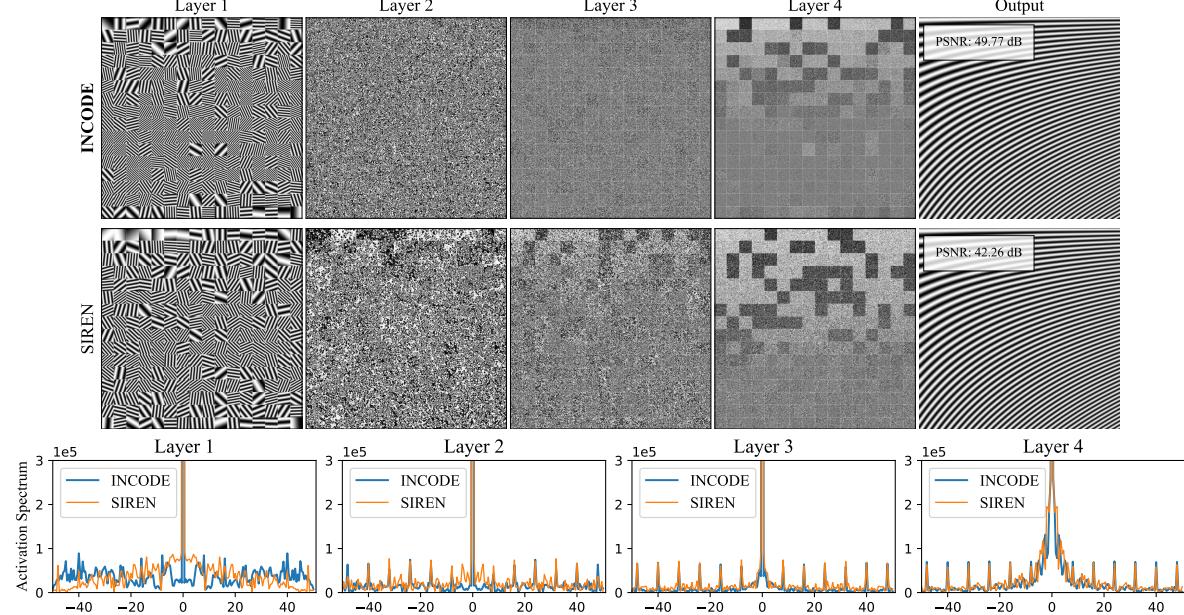


Figure 2. Comparison of frequency response representations of the proposed method vs. SIREN across network layers.

plays a distinct role in shaping the behavior of the activation function. We now proceed to individually analyze the effects of each variable and see how they increase the representation accuracy and robustness:

a (Amplitude): The amplitude \mathbf{a} plays a pivotal role in vertically scaling or stretching the sinusoidal wave. Specifically, our approach involves adjusting \mathbf{a} during the learning process to influence the strength of the activation function’s response. In denoising, a higher \mathbf{a} could potentially amplify noise, whereas, in representation, it could enhance the emphasis on certain features. As a result, guiding the model to achieve an optimal balance for \mathbf{a} leads to a feature enhancement in the representation tasks and noise suppression in the denoising-related tasks. Hence, the first objective of the harmonizer module is to optimally set the \mathbf{a} value based on the given task. This adaptive learning is a departure from fixed activation functions and allows the model to self-regulate its response based on the specific characteristics of the data.

b (Frequency Scaling): The variable \mathbf{b} governs the frequency scaling of the sinusoidal wave. The adjustment of \mathbf{b} plays a significant role in accentuating either finer or coarser details within the representation. Thus, careful selection of \mathbf{b} enables attenuating high-frequency noise in denoising tasks and regulates the granularity of captured features in representation tasks. Hence, adaptive calibration of \mathbf{b} during the learning process effectively enhances the representation capacity of representation models while reducing the high-frequency noise in the denoising tasks.

c (Phase Shift): The phase shift parameter \mathbf{c} horizontally displaces the sinusoidal wave along the x-axis. This adjustment impacts the alignment of features represented by the activation function, influencing their spatial arrangement

within the model’s generated representation. Consequently, modifying \mathbf{c} holds the potential to affect the quality and fidelity of the resulting representation. In denoising, altering \mathbf{c} can shift noise patterns, altering their perceptibility in the output; therefore, the model can learn to balance the effect of noise by shifting the sinusoidal wave.

d (Vertical Shift): The variable \mathbf{d} in the activation function acts as a vertical shift. Increasing \mathbf{d} adds a constant positive offset to the entire function, resulting in a raised baseline. This adjustment effectively enhances the overall brightness of the generated image, akin to intensifying light or color. By elevating \mathbf{d} , the output values of the activation function shift upwards, creating a visually brighter appearance in the representation. Thus, manipulating \mathbf{d} provides a mechanism for controlling baseline brightness within the INR framework. Therefore, devising a mechanism that dynamically adjusts these variables during the learning process at each iteration can guide us towards achieving our objectives of constructing a resilient model with substantial representation capacity.

3.1.2 Harmonizer Network

The harmonizer network employs an MLP architecture consisting of K hidden layers and p_1, p_2, \dots, p_K hidden features. Its primary function revolves around the direct regulation of the amplitude, frequency, and displacement of the sinusoidal activation such that it is defined as $g(z; \theta) : \mathbb{R}^r \rightarrow \mathbb{R}^4$. This network is structured to predict the $\mathbf{a}, \mathbf{b}, \mathbf{c}$, and \mathbf{d} values dynamically. It initiates its function by receiving a latent code z from a task-specific pre-trained model. Subsequently, it endeavors to predict these variable values in an adaptive manner. In our architectural framework, we

strategically integrate a task-specific pretrained model, harnessing the invaluable insights gained from its extensive training on a large-scale dataset. This integration is pivotal in dynamically transforming the data into a meaningful latent space, subsequently facilitating its utilization by the harmonizer network.

3.2. Loss function

In our approach, we employ the mean squared error (MSE) as a metric to minimize the differences between the predicted signal values and their corresponding true values. This optimization objective aims to make the predicted values closely align with the actual data. Additionally, we introduce a regularization term to the loss function. This term is designed to enforce positive values for the parameters **a**, **b**, **c**, and **d**. We enforce the variables to be positive in order to guide the model towards more relevant solutions, encourage the model to converge more rapidly, and reduce the likelihood of becoming trapped in a local optimum during the training process. This regularization mechanism contributes to a more efficient and effective optimization process, enhancing the overall performance of the model. Our loss function is defined as follows:

$$\begin{aligned} \arg \min_{\theta, \mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}} & \mathbb{E} [\|f(\mathbf{x}; \theta) - S(\mathbf{x})\|_2^2] \\ \text{s.t. } & \mathbf{a} \geq 1, \quad \mathbf{b} \geq 1, \quad \mathbf{c} \geq 0, \quad \mathbf{d} \geq 0. \end{aligned} \quad (4)$$

We control the strength of the regularization applied to the parameters **a**, **b**, **c**, and **d** through corresponding coefficients λ_1 , λ_2 , λ_3 , and λ_4 in the optimization process, enabling us to manage the trade-off between fitting the data and imposing constraints on the parameter values.

3.3. Expressiveness of INCODE

This section explores INCODE’s expressive capabilities and compares them with the SIREN architecture. Yüce et al. [51] analyze the two-layer SIREN. In a SIREN with two layers and input x , the first layer produces $Z^{(0)} = \sin(\Omega x)$, and the second layer yields $Z^{(1)} = \sin(\omega^{(1)} \sin(\Omega x))$. The second-layer output in SIREN can be expressed as:

$$\sum_{m=0}^{P-1} \sum_{s_1, \dots, s_N=-\infty}^{+\infty} \left(\prod_{t=0}^{N-1} J_{s_t} W_{m,t}^{(1)} \right) \sin \left(\sum_{t=0}^{N-1} s_t w_t x \right), \quad (5)$$

where J_s defines the Bessel function of the first kind of order s . The decreasing nature of $J_{s_t} W_{m,t}^{(1)}$ results in higher-order harmonics carrying smaller weights, concentrating energy around a narrow band centered at input frequencies Ω . Scaling coefficients like $\omega^{(1)}$ amplify higher-order harmonics, enabling a broader range of learnable frequencies.

INCODE introduces a harmonizer network learning activation function parameters a , b , c , and d , leading to $a \sin(b\Omega + c) + d$. The simplified second-layer output in

INCODE, with only a and b , becomes:

$$a \sum_{m=0}^{P-1} \sum_{s_1, \dots, s_N=-\infty}^{+\infty} \left(\prod_{t=0}^{N-1} J_{s_t} W_{m,t}^{(1)} ab \right) \sin \left(\sum_{t=0}^{N-1} s_t w_t bx \right). \quad (6)$$

The term a enhances noise robustness and emphasizes signal details, while ab amplifies coefficients for higher-order terms, broadening the frequency spectrum beyond that of SIREN, given that $ab \geq 1$. To ensure this condition, we consider **a** and **b** as e^a and e^b , respectively, in our proposed activation function to fulfill this condition. Parameter c crucially produces e^{jc} terms, effectively controlling b to prevent unbounded growth. This control enhances network stability and maintains meaningful frequency components.

For experimental purposes, an image is generated within specific frequency ranges, transmitting information from low to high frequency. Empirical evidence in Figure 2 demonstrates higher amplitudes at higher frequencies in INCODE’s first layer, confirming enhanced mapping capabilities and expanded frequency bandwidth compared to SIREN. The parameterization of the harmonizer network achieves broader frequency coverage while retaining sensitivity to essential signal details.

4. Experiments

Implementation Details. We utilize a 5-layer composer network with 256 units for all experiments. Specifics regarding the harmonizer network can be found in the relevant task description. Our experiments are performed using PyTorch on an Nvidia RTX 3070 Ti GPU with 8GB memory. We use the Adam optimizer [14] with a learning rate scheduler, aiding convergence by decreasing the learning rate by α at each epoch’s completion. Experiments are conducted for 500 epochs, except audio (1000 epochs), occupancy (200 epochs), and CT reconstruction (2000 epochs). λ_1 , λ_2 , λ_3 , and λ_4 are set to 0.1993, 0.0196, 0.0588, and 0.0269, respectively, obtained by training the model for the image representation task on 10 samples and using Optuna [1] for hyperparameter optimization. We extensively compare our methods with WIRE [38], SIREN [38], MFN [9], Gaussian [34], ReLU with Positional-Encoding (ReLU+P.E.) [43], and FFN [43]. Further architectural details of these methods are available in the supplementary materials.

4.1. Signal Representations

4.1.1 Image

Data. We conducted our image representation experiments using the DIV2K dataset [46], which was downsampled by a factor of 1/4. For example, Figure 3 is downsampled from $1644 \times 2040 \times 3$ to $411 \times 510 \times 3$.

Architecture. The Composer network maps 2D coordinates to RGB values, and w_o is set to 30 for this network. The Harmonizer network is a 3-layer MLP with 64, 32, and 4 features, equipped with the SiLU [8] activation function.

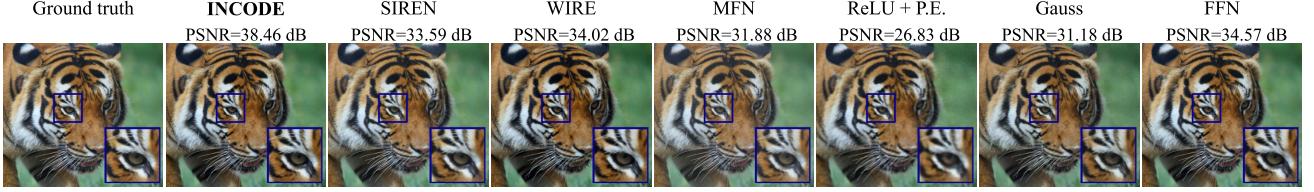


Figure 3. **Image representation:** Comparison of INCODE with SOTA methods.

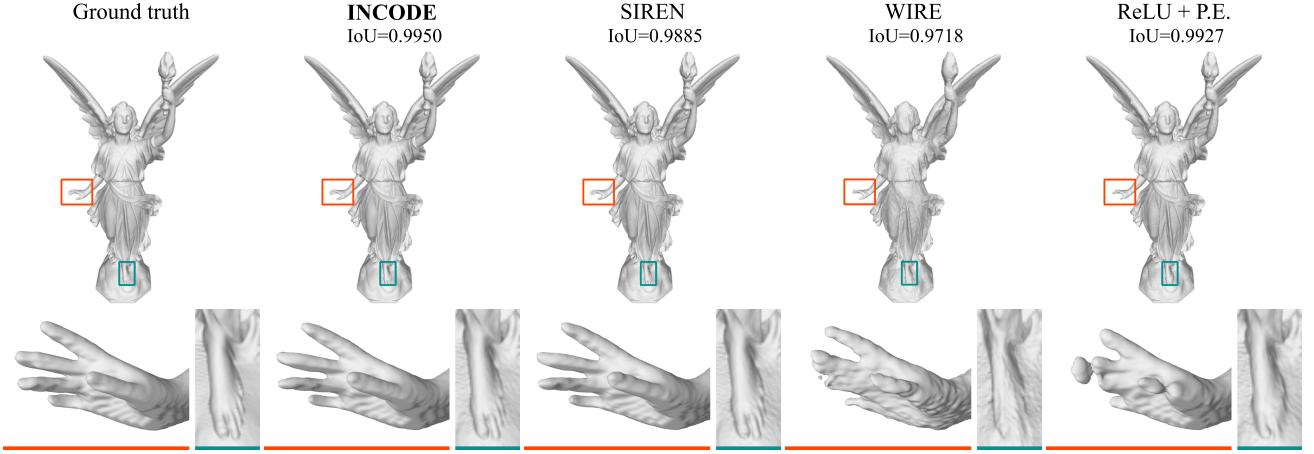


Figure 4. **Occupancy volume representation:** Comparison of INCODE with SOTA methods.

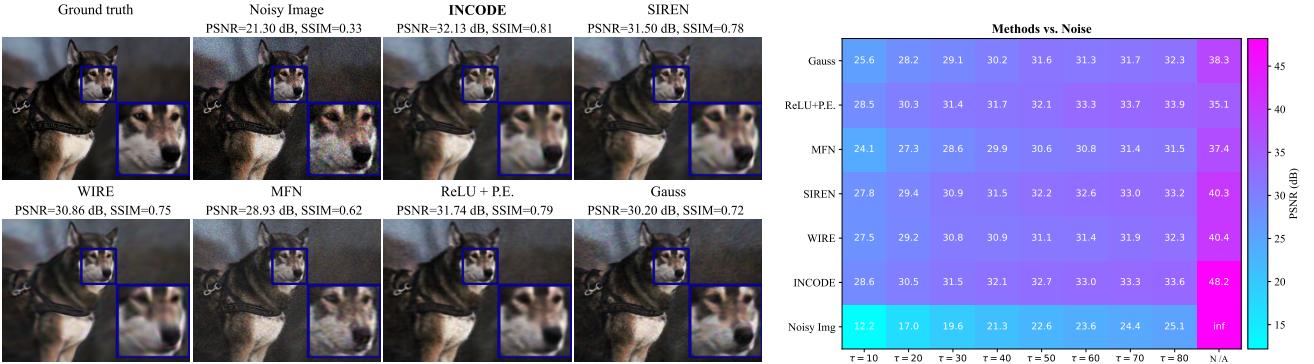


Figure 5. **Image denoising:** Qualitative and quantitative comparison of INCODE with SOTA methods.

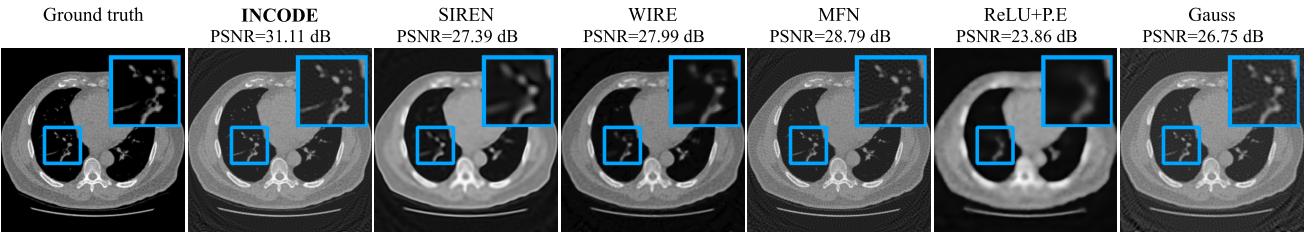


Figure 6. **CT Reconstruction:** Comparison of CT-based reconstruction with 150 angles with SOTA methods.

It maps the generated latent code with $r = 64$ to the four parameters of activation. Weights of this network are normally initialized $\mathcal{N}(0, 0.001)$ with constant biases of 0.31. We use ResNet34 [11] truncated to its fifth layer, followed by an adaptive average pooling, to generate the latent code.

The learning rate is set to 9×10^{-4} and α to 0.1.

Analysis. The experimental results of image representation are presented in Figure 3. The results clearly indicate that INCODE outperforms its counterparts in terms of representation quality. Notably, it achieves a substantial en-

hancement of +3.89 dB in PSNR values compared to the nearest counterpart, FFN, and +4.44 dB and +4.48 dB improvements compared to WIRE and SIREN, respectively. Additionally, INCODE has shown a sharper reconstruction of the tiger eyebrow than the other methods, particularly ReLU+P.E. and MFN. This observation underscores the promising potential of INCODE for image representation, capable of producing sharper images with finer details. More results are in the supplementary file.

4.1.2 Occupancy Volume

Data. We use the Lucy dataset from the Stanford 3D Scanning Repository and follow the WIRE strategy [38]. We create an occupancy volume through point sampling on a $512 \times 512 \times 512$ grid, assigning values of 1 to voxels within the object and 0 to voxels outside.

Architecture. Our network and training configurations resemble the image representation task, with the distinction that the composer network now maps 3D ($M = 3$) coordinates to signed distance function (SDF) values ($N = 1$). Utilizing ResNet3D-18 [47] truncated to the third layer for feature extraction to generate a latent code of size 128, our approach effectively incorporates volumetric data into the composer network.

Analysis. The results showcased in Figure 4 underscore INCODE’s effectiveness as a robust replacement for its counterparts in occupancy representation tasks. Remarkably, INCODE adeptly harnesses the informative latent code to condition the composer network, yielding an amplified representation capacity. This augmentation is particularly evident in the intensification of high-frequency information while also adeptly capturing low-frequency details. Our method yields higher Intersection over Union (IOU) values, particularly excelling in replicating intricate details such as Lucy’s hand and foot. INCODE remarkably enhances object details and scene complexity, enabling more accurate representation compared to existing methods.

4.1.3 Audio Representations

Data. We use the first 7 seconds of Bach’s Cello Suite No. 1: Prelude [41], with a sampling rate of 44100 Hz as our example for the audio representation task.

Architecture. The composer network transforms 1D ($M = 1$) input to its corresponding 1D output ($N = 1$). It employs strategic frequency initialization for effective learning due to the nature of audio: w_0 is set to 3000 for the first layer to capture high spatial frequency information, and hidden w_0 is set to 30 for subsequent layers. To capture audio features, Mel Frequency Cepstral Coefficients (MFCCs) [20] serve as the feature extractor. MFCCs encode both frequency and temporal information, suited for audio representation. The harmonizer network utilizes extracted features and generates the activation parameters. Also, the learning rate is 9×10^{-5} , and α is 0.2.

Analysis. We evaluate INCODE’s performance against established methods to gauge its effectiveness in audio signal representation. Results highlight INCODE’s substantial reduction in error rates and increase of +10.60 dB PSNR value compared to the second best, Gauss (See Supplementary, Figure 8). The periodicity of audio signals at various time scales leads to an accurate and efficient representation in INCODE, akin to SIREN. INCODE converges swiftly to a distortion-minimized representation, while Gauss and ReLU+P.E. methods manifest distortion during playback. Although SIREN strives to mitigate this, some dominant noise is witnessed in the background. INCODE notably excels in this aspect, as evidenced in its error figure.

4.2 Inverse Problems

4.2.1 Image denoising

Data. We employ an image from DIV2K dataset [46], downsampled by a factor of 1/4 from $1152 \times 2040 \times 3$ to $288 \times 510 \times 3$. We create the noisy image using realistic sensor measurement with readout and photon noise, where independent Poisson random variables are applied to each pixel. The mean photon count (τ) varied between 10 and 80, while the readout count (ro) set fixed at 2.

Architecture. The composer network is similar to previous tasks, however, we set w_0 to 10 for the first layer, while the other layers remain at 30. The choice of w_0 in the initial layer plays a crucial role in achieving a denoised image with higher fidelity and fewer artifacts. By setting w_0 to a lower value, the network becomes more adept at capturing low-frequency information and smoothing out noise-related variations. The first layer w_0 can also be calibrated in alignment with the noise characteristics to attain optimal signal quality. The harmonizer network is a 4-layer MLP, containing 32, 16, 8, and 4 nodes. Each layer is followed by a LayerNorm and SiLU activation function. This network is responsible for mapping the latent code ($r = 64$) generated by ResNet34 to the activation parameters. Weights are initialized using the normal distribution of $\mathcal{N}(0, 0.001)$ and constant biases of 0.0005. This initialization emphasizes the relevant signal components and suppresses noise-related artifacts. We train the model with a learning rate of 1.5×10^{-4} and $\alpha = 0.1$.

Analysis. We demonstrate the effectiveness of INCODE in solving inverse problems using the example of image denoising, capitalizing on its inductive bias and robustness. The visual comparison of our approach is presented in Figure 5 for $\tau = 40$ and $ro = 2$, where INCODE significantly enhances the fidelity of the noisy image with a +10.83 dB PSNR improvement and a 0.48 increment in the Structural Similarity Index (SSIM) metric. INCODE adeptly preserves image details while mitigating noise artifacts, particularly when compared to the MFN and Gauss methods, where noise effects still persist in the output. Furthermore,

our approach outperforms the ReLU+P.E. method by 0.39 dB and 0.02 in terms of SSIM. Furthermore, we present a histogram visualization in [Figure 5](#), wherein the image is subjected to varying degrees of noise to illustrate the comparative performance of each approach and the incremental trend as noise influence diminishes. Additionally, we conduct the methods without noise to exhibit the capacity of each approach in both denoising and representation tasks. Evidently, INCODE has shown a comparable performance with the ReLU+P.E. method, while the ReLU-based networks are particularly good for prioritizing learning low-frequency information, demonstrating the robustness and power of the INCODE in denoising tasks.

4.2.2 Image super resolution

Data. We adopt an image from the DIV2K dataset [46] and downsampled the image with the size of $1356 \times 2040 \times 3$ by factors of $1/2$, $1/4$, and $1/6$.

Architecture. We maintain the same architectural and training settings as the image representation task. By employing a downsampled image during training, we exploit the interpolation capabilities of INRs to reconstruct an image of its original size in the test.

Analysis. In super-resolution, the application of INRs as interpolants presents a promising avenue. This notion indicates that INRs possess inherent advantageous biases that can be harnessed to enhance super-resolution tasks. To validate this proposition, we conducted $1\times$, $2\times$, $4\times$, and $6\times$ super-resolution experiments on an image. As presented in [Table 1](#), the results demonstrate that INCODE consistently achieves superior PSNR and SSIM values across all super-resolution levels, outperforming alternative methods. Furthermore, the visual demonstration of INCODE's superiority is presented in (Supplementary, [Figure 12](#)), revealing its ability to retain sharper details compared to others that often result in blurrier outcomes.

Table 1. INCODE vs. SOTAs in super-resolution.

Methods	$1\times$		$2\times$		$4\times$		$6\times$	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Gauss	30.25	0.79	29.05	0.86	27.07	0.83	25.04	0.79
FFN	32.96	0.90	29.33	0.85	29.41	0.86	27.01	0.84
MFN	31.20	0.84	30.78	0.87	29.35	0.86	27.23	0.83
ReLU P.E.	32.41	0.87	30.29	0.88	25.52	0.82	24.26	0.81
WIRE	31.57	0.84	31.37	0.86	28.55	0.83	24.77	0.72
SIREN	32.10	0.87	31.51	0.89	28.81	0.85	26.46	0.84
INCODE	33.44	0.91	33.02	0.92	29.88	0.87	27.57	0.85

4.2.3 CT reconstruction

Data. We use a publicly available CT lung image (256×256) from the Kaggle Lung Nodule Analysis dataset [2] to assess our model's performance in CT reconstruction.

Architecture. The architecture remains consistent with that of the image representation task. INCODE involves using the ResNet34 architecture to process the undersampled sinogram and generate a latent code. We conduct training

over 2000 epochs, using the learning rate of 2×10^{-4} , coupled with $\alpha = 0.4$. We generate a sinogram according to the projection level using the radon transform. The model predicts a reconstructed CT image. Subsequently, we calculate the radon transform for the generated output and compute the loss function between these sinograms, to guide the model toward generating CT images with reduced artifacts.

Analysis. CT reconstruction is the process of generating a computed image from sensor measurements. Sparse CT reconstruction deals with the added complexity of generating accurate images when only a subset of measurements is available, posing challenges due to limited data constraints. INCODE addresses this challenge by employing a conditional harmonizer network to seamlessly integrate deep prior information into the model. As shown in [Figure 6](#), INCODE stands out by producing sharp reconstructions with clear details using 150 measurements (+2.32 dB improvement compared to the second best, MFN). Conversely, MFN shows artifacts similar to WIRE, and Gauss, yet achieving higher PSNR values. On the other hand, SIREN and ReLU+P.E. yield overly blurred results with reduced details. This underscores INCODE's robustness in addressing challenges posed by noisy and undersampled inverse problems. Its ability to balance image fidelity and noise reduction establishes INCODE as a promising solution in the underconstrained image reconstruction landscape. We also explore the relationship between the number of projections and the reconstructed CT quality (see Supplementary, [Figure 9](#)). Our method maintained its superiority compared to SOTA methods, underscoring its resilience in the face of measurement noise.

4.2.4 Inpainting

Data. We utilize Celtic spiral knots image with a resolution of $572 \times 582 \times 3$. The sampling mask is generated randomly, with an average of 20% of pixels being sampled.

Architecture. We adopt a configuration similar to that of image representation architecture, albeit with adjustments tailored to the task-specific model. Due to the random pixel sampling, a pre-trained model like ResNet cannot be employed. Hence, a custom model is crafted for latent code generation, consisting of two layers [Conv1D, ReLU, Max-Pooling], followed by another Conv1D layer. The resulting latent code is of size 64. For training, the model employs a learning rate of 1.5×10^{-4} and $\alpha = 0.25$.

Analysis. Despite only sampling 20% of pixels, INCODE effectively addresses inverse problems as demonstrated by single-image inpainting. The output in (Supplementary, [Figure 10](#)) illustrates that INCODE can achieve performance on par but better with other baseline methods. Specifically, INCODE exhibits the ability to generate sharper results with more details.

4.3. Neural radiance fields

Neural Radiance Fields (NeRFs) [25] combine INRs and volume rendering by using MLPs equipped with ReLU+P.E., aiming to implicitly represent scenes for synthesizing novel views. By training a 3D implicit function using spatial coordinates (x, y, z) and viewing directions (θ, ϕ), NeRFs can predict the color and density of that specific location. This allows for generating new views of objects from different angles by tracing camera rays through pixels using neural rendering. We, therefore, investigate the effectiveness of using INCODE without positional encoding in the NeRF. We found that our approach yields superior results in fewer epochs. We substantiate the excellence of our approach through comparative analyses and results showcased in the supplementary material.

5. Conclusion

In this paper, we have presented INCODE, a transformative approach to Implicit Neural Representations (INRs) that significantly enhances their representation capacity. By introducing a dynamic sinusoidal-based activation function with adaptive control, INCODE overcomes the limitations of existing INRs. The harmonizer network, guided by deep prior knowledge, dynamically adjusts activation function parameters, enabling the model to adapt to specific data characteristics. Our experiments demonstrate the superior performance of INCODE across a wide range of tasks.

6. Supplementary Material

A. Experimental Results

In this section, we broaden our experimental scope to encompass a more comprehensive comparison between our approach and state-of-the-art (SOTA) methods. We have demonstrated that the inherent simplicity of INCODE contributes to enhanced performance compared to its counterpart SOTA methods, specifically in terms of expressiveness and representation capacity. These findings underscore the efficacy of our approach in pushing the boundaries of INR networks and facilitating their applicability across diverse domains. We now present additional visualizations that distinctly show the advantage of our approach.

A.1. Image representation

As depicted in [Figure 7](#) and [Figure 11](#), it is evident that INCODE achieves superior qualitative and quantitative performance. Particularly in [Figure 7](#), INCODE exhibits an approximate accuracy improvement of +2.98 dB and +4.59 dB compared to FFN [43] and WIRE [38], respectively. The zoomed-in image distinctly illustrates INCODE’s ability to grasp intricate details of the Eiffel Tower. In contrast, ReLU+P.E. and MFN [9] yield blurry outcomes, while

Gauss [34] displays slight color alteration, although it captures certain intricate features. Gauss also struggles to recognize the orange object positioned at the tower’s center. Likewise, SIREN [41] fails to capture the full complexity of the tower’s structure, leading to a smoothed and blurred representation.

Additionally, [Figure 11](#) presents a challenging image with intricate patterns, posing a challenge for representation. Notably, INCODE and FFN emerge as the sole methods achieving a PSNR value over 30 dB, with INCODE exhibiting a +1.56 improvement over the second-ranking FFN. As evidenced in the zoom-in image, ReLU+P.E. expectedly yields a blurred output, given the inherent properties of its ReLU activation function. Interestingly, WIRE and Gauss encounter difficulty in precisely grasping the image’s color characteristics, leading to slight color differences. While MFN effectively addresses this color challenge, it falls short in capturing the image’s intricate details, particularly its edges.

Overall, our study shows that INCODE excels in image representations. It consistently outperforms other methods across various images, even with intricate patterns. This success is due to INCODE’s ability to capture intricate details. While alternative methods faced challenges in representing complex patterns, colors, or high-frequency information, INCODE exhibited competence in addressing these challenges. Thus, our findings highlight INCODE as one of the optimal choices for robust and superior image representation.

A.2. Audio representation

We present audio representation visualization results along with its error maps in [Figure 8](#). These visualizations help to understand the strength of our approach. We have provided a detailed analysis of these results in the main section of the paper to ensure a comprehensive understanding of our findings. In terms of sound playback quality, Gauss introduces a noticeable squeak-like sound that accompanies the main audio. With ReLU+P.E., noise dominance becomes more pronounced, making it difficult to discern the original sound. While employing SIREN, some moments are marred by bothersome noise, as indicated by the error map. However, INCODE significantly outperforms these methods by having notably less noise interference. This aspect positions INCODE as a favorable choice for encoding audio data with improved quality.

A.3. Super resolution

To illustrate the efficacy of our approach in the super-resolution task, we have included a visual comparison of 4 \times super-resolution in [Figure 12](#). From a quality perspective, INCODE produces sharper results with finer details in the butterfly’s wing, while the blurred outcomes of SIREN,

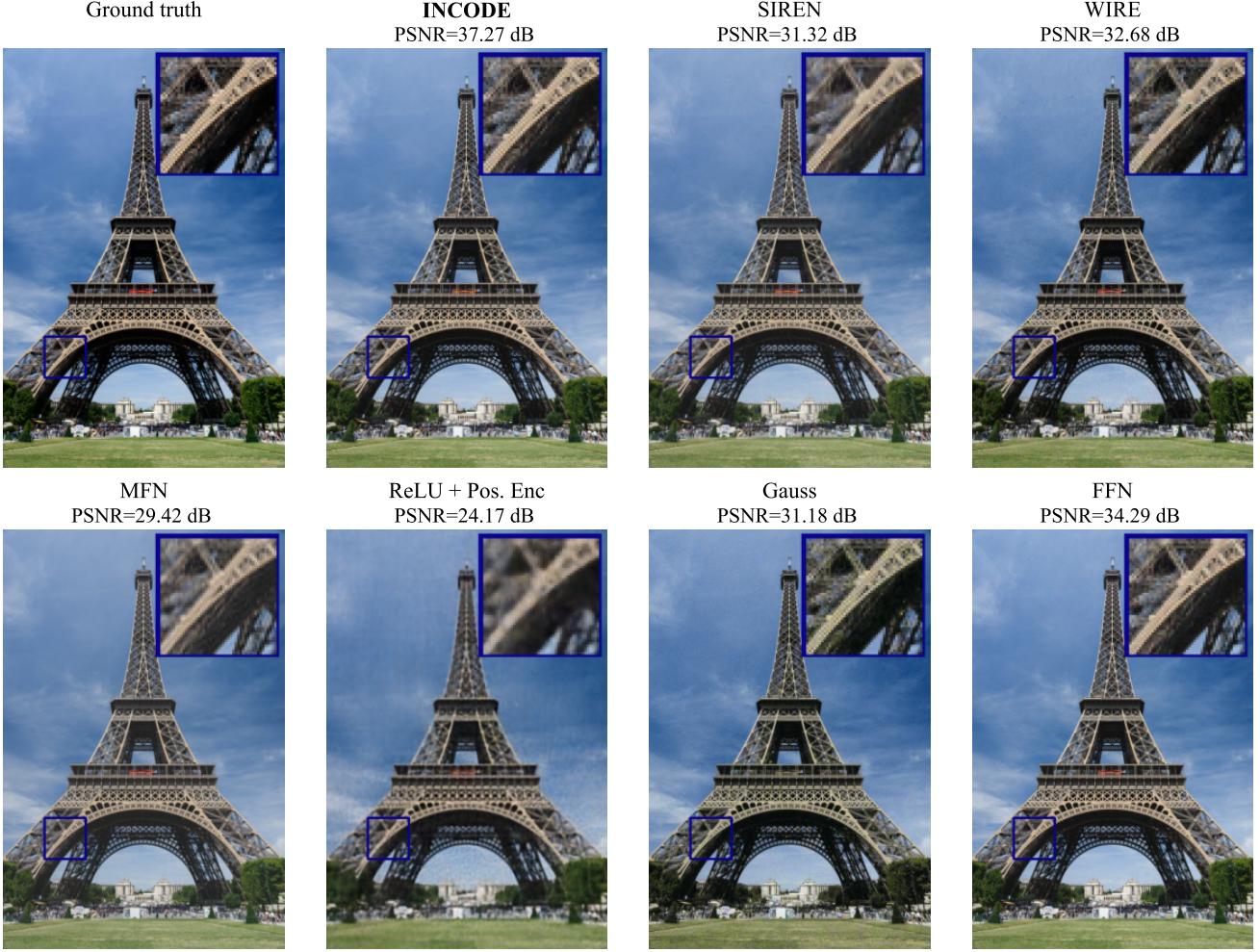


Figure 7. **Image representation:** Comparison of INCODE with SOTA methods.

FFN, Gauss, and ReLU+P.E. are evident, even though the quantitative values are relatively close. This visual comparison supports our quantitative findings in [Table 1](#) (see the main paper) and affirms INCODE’s proficiency in super-resolution tasks, where it offers better quality when performing upsampling.

A.4. Computed Tomography (CT) reconstruction

Under-measurement in CT samples results from a range of factors that reduce the accuracy of the imaging process. Artifacts, stemming from issues like patient movement during scanning, metallic objects causing beam distortion, and equipment calibration problems, contribute to discrepancies. INRs address these concerns and solve this inverse problem by leveraging their inductive bias. We investigate the impact of varying the number of measurements (ranging from 50 to 400, with increments of 50) as shown in [Figure 9](#). Notably, SIREN, WIRE, and ReLU+P.E. yield con-

sistent results across all measurements. Particularly, WIRE excels in CT reconstruction with 50 measurements; however, increasing the data information in such models doesn’t enhance their performance, indicating saturation. In contrast, INCODE exhibits considerable improvement as measurements increase from 100 to 400, showcasing the effectiveness of incorporating deep prior information. Notably, INCODE with 150 measurements outperforms all nonlinearities in the full range of projection numbers, except for MFN, which closely competes after reaching 200 projections and performs the second best. These findings acknowledge the robustness and power of INCODE in addressing under-measurement challenges within CT reconstruction.

A.5. Inpainting

Image inpainting poses a formidable challenge as models are tasked with predicting entire pixel values based on only

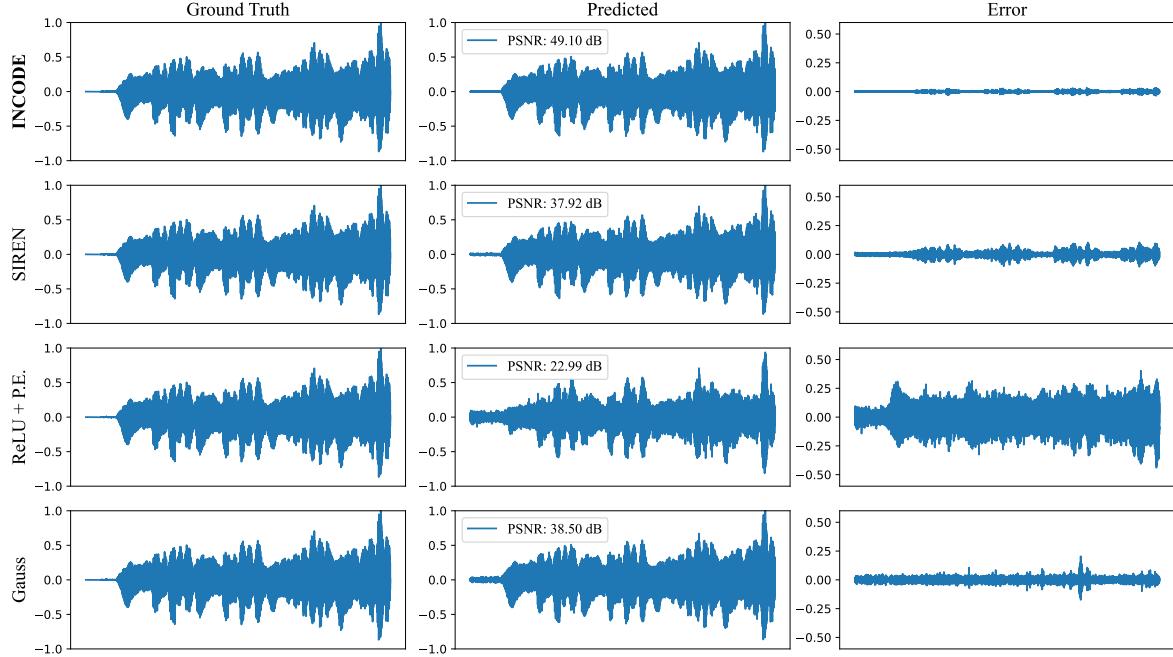


Figure 8. **Audio representation:** We compare INCODE with SOTA methods for audio representation. In the third column, we display the reconstruction error.

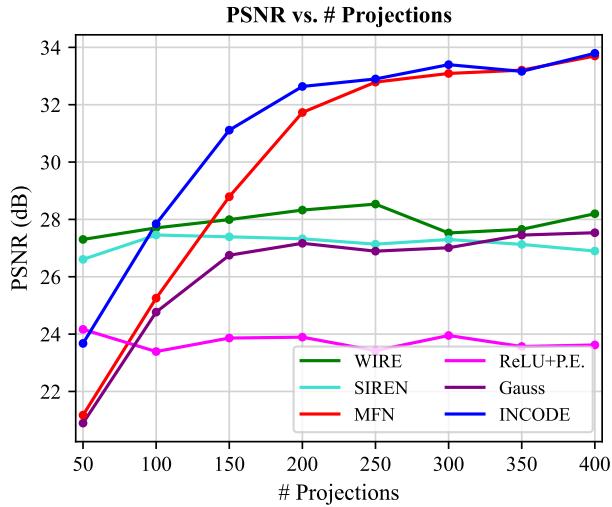


Figure 9. INCODE vs. SOTAs in CT reconstruction across different numbers of projections.

a fraction of trained pixel data. The high capacity of INR provides the opportunity to accomplish this inverse problem challenge. The strong prior ingrained within the space of INR functions paves the way for applications like inpainting from limited observations, where it uses the learned representation of the trained model to predict inpainting missing values. Our approach involves randomly sampling 20% of the pixels and then employing the model’s learned representation to predict the missing pixels. The comparison re-

sult is shown in Figure 10. As observed in other tasks, INCODE’s power in capturing intricate features, particularly edges, stands out compared to other methods that tend to yield blurred outcomes. While a modest +0.38 dB improvement is noted compared to SIREN, the visual presentation demonstrates that SIREN, much like ReLU+P.E., struggles to comprehensively capture high-frequency details.

A.6. Neural radiance fields

In our approach, we followed a strategy akin to [38], making use of the publicly available torch-ngp package [44, 45] to train the NeRF model. Our NeRF architecture encompasses two main networks: one for predicting sigma (σ) and the other for determining color (RGB). These networks are constructed as 4-layer MLPs, each with 182 hidden features.

Additionally, we introduced two harmonizer networks, one for the sigma network and another for the color network. These harmonizers employ 4-layer MLPs, featuring 32, 16, 8, and 4 nodes, with each layer followed by LayerNorm and the SiLU activation function. They receive a latent code and condition their corresponding composer networks, which are initialized similarly to the denoising task.

To generate the latent code, we utilized a truncated ResNet34 model at its fifth layer, followed by adaptive average pooling. During training, a single random image from the training dataset was used, and for testing and validation, we again employed one random training image. The color MLP took positional coordinates (x, y, z) and direction pa-

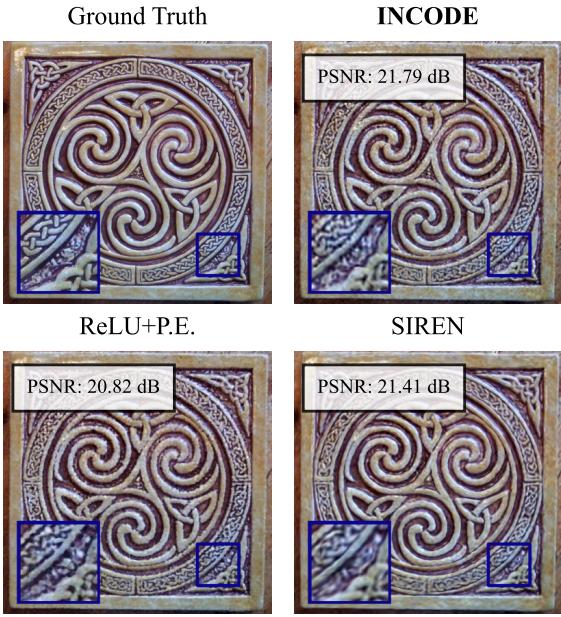


Figure 10. **Image inpainting:** Comparison of INCODE with SOTA methods.

rameters (θ, ϕ) as inputs, while the sigma MLP solely required positional information.

For our experimental results, depicted in Figure 13, we utilized a Lego dataset comprising 100 training images, each downsampled by 1/2 to 400×400 dimensions, for training the NeRF. Subsequently, we evaluated the model’s performance on an additional 200 images. Training of the NeRF models was conducted on an A-100 GPU with 20 GB of memory. Throughout training, we used learning rates of 3×10^{-4} for INCODE, 3×10^{-4} for SIREN, 6×10^{-4} for WIRE, 3×10^{-3} for Gauss, and 1×10^{-2} for ReLU+P.E. The learning rate is decreased to $0.1 \times$ initial value over a total of 3000 training epochs to achieve their optimal outputs. Additionally, we set omega (ω_0) to 40 for INCODE, SIREN, and WIRE, and sigma (s_0) to 40 for WIRE and Gauss. Apart from ReLU, we did not use positional encoding for other nonlinearities to highlight their individual capabilities.

As shown in Figure 13, our approach achieves a +0.16 dB improvement over SIREN and a +0.79 improvement compared to WIRE. Qualitative results also demonstrate a superior performance of INCODE compared to SOTA models. Notably, INCODE excels in capturing fine-grained details and information. For instance, it effectively captures intricate features such as the middle black connector in the loader, while SIREN failed to learn. Also, INCODE outperforms other methods like WIRE, ReLU+P.E., and Gauss, which exhibit blurred and smooth results in comparison.

B. Experimental Analysis

B.1. Convergence rate comparison

We analyze the convergence rate of INCODE in comparison to other methods across three distinct representation tasks: image, occupancy volume, and audio, as depicted in Figure 14. The data used for each task corresponds to the respective domain in the main paper. Remarkably, INCODE consistently showcases accelerated convergence compared to SOTA architectures. This expedited convergence is most pronounced in the audio domain, where a substantial gap between SIREN and INCODE is evident. Leveraging its robust approximation capacity, INCODE achieves fast convergence with high fidelity, rendering it an apt choice for representing different signals.

B.2. Impact of depth and width of the network

The analysis of the network’s depth and width are presented in Figure 15, which sheds light on the impact of architectural parameters in shaping the performance of INCODE. By systematically varying the number of hidden layers and their width, we gain insights into the trade-off between model complexity and approximation accuracy.

In the left figure, we vary the network’s depth from 2-layer MLP to 6-layer, while keeping the width constant at 256. Notably, INCODE exhibits competitive performance compared to other methods in lower layers. However, as the network deepens, INCODE distinctly outperforms FFN, demonstrating its capacity to effectively capture more intricate information with increasing model depth. Shifting to the right figure, we explore the effect of hidden features by adjusting the network’s width from 64 to 320, in increments of 64, while maintaining a 5-layer MLP. The trend depicted in the plot accentuates INCODE’s remarkable performance, showcasing a steep ascent. Throughout the spectrum of hidden feature counts, INCODE consistently outperforms other SOTA methods. This observation highlights INCODE’s proficiency in capturing broader patterns as the width of the network expands, underlining its versatility and ability to adapt to varying levels of complexity.

C. Experimental details

In all experiments, we employed a 5-layer MLP with 256 hidden features for all architectures. However, for WIRE, we followed their recommended structures as outlined in their paper to achieve optimal performance. Specifically, for image-based tasks, we used a 4-layer MLP with $s_0 = 30$ and $\omega_0 = 20$, featuring 300 hidden features. For the occupancy task, we utilized a 4-layer MLP with 256 hidden features, alongside $s_0 = 40$ and $\omega_0 = 10$. In the case of CT reconstruction, we employed a 5-layer MLP with 256 hidden features and set $s_0 = 10$ and $\omega_0 = 10$. Lastly, for the denoising task, we opted for the same architecture as

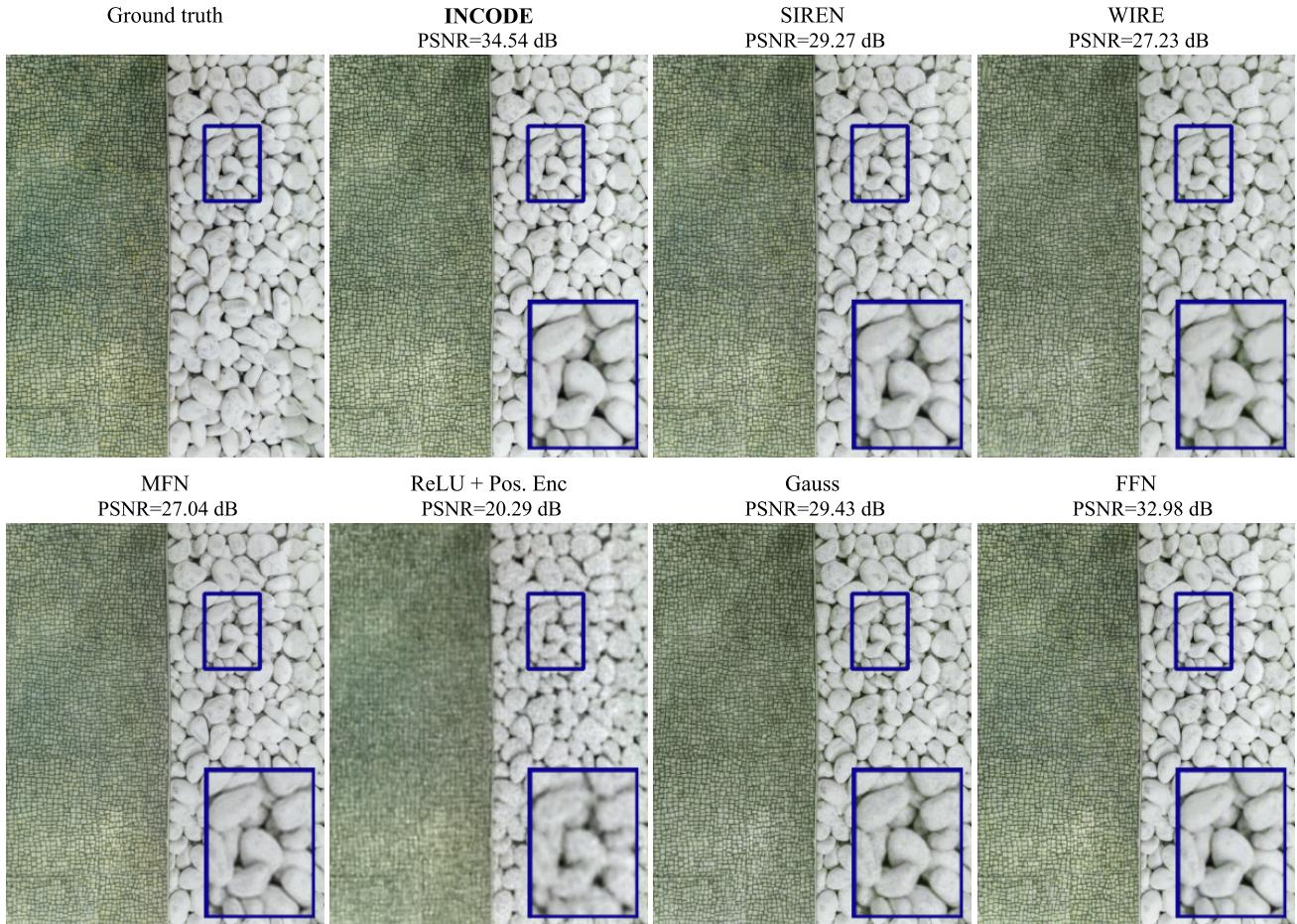


Figure 11. **Image representation:** Comparison of INCODE with SOTA methods.

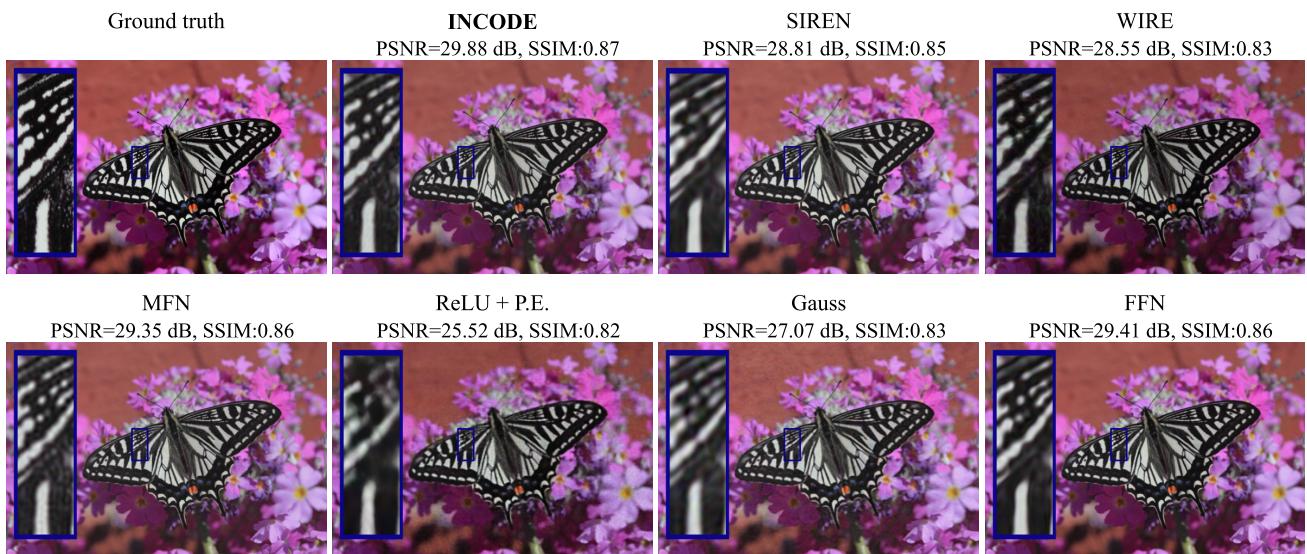


Figure 12. **Super Resolution.** Results of a 4 \times single image super-resolution using various approaches



Figure 13. Neural Radiance Fields: The figure presented above illustrates rendered images generated by a neural radiance field using different methods. Notably, INCODE consistently outperforms all other methods in terms of visual reconstruction quality, highlighting its robust feature representation.

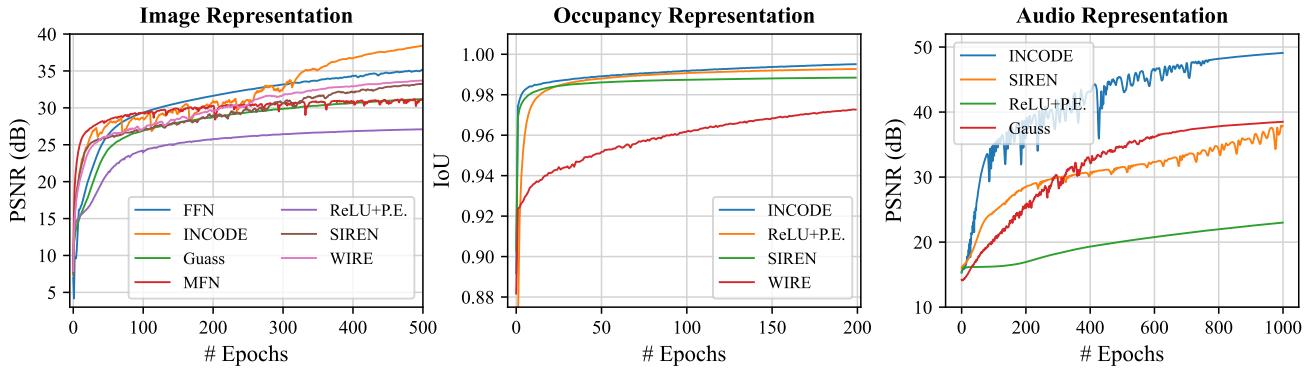


Figure 14. Convergence rates in different representations: Explore the convergence rates of Image, Occupancy volume, and Audio representations.

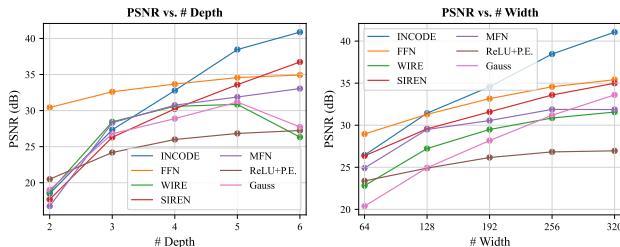


Figure 15. Impact of network depth and width: Explore the influence of network depth and width on performance.

the image representation and for $s_0 = 4$ and $\omega_0 = 4$. In FFN, a mapping input size of 256 is utilized, for instance,

to map image coordinates from 2 to 512, and the parameter \mathcal{B} , a random Gaussian matrix, is scaled by a factor of 10. We configured the value of s_0 for the Gauss model as follows: $s_0 = 30$ for image representation, $s_0 = 100$ for audio representation, and $s_0 = 10$ for the inverse problem tasks. In addition, we utilized the same initial parameters as described for INCODE in the case of SIREN.

References

- [1] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2019. 5

- [2] Reza Azad, Maryam Asadi-Aghbolaghi, Mahmood Fathy, and Sergio Escalera. Bi-directional convlstm u-net with densley connected convolutions. In *Proceedings of the IEEE/CVF international conference on computer vision workshops*, pages 0–0, 2019. 8
- [3] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5470–5479, 2022. 2
- [4] Zhiqin Chen, Thomas Funkhouser, Peter Hedman, and Andrea Tagliasacchi. Mobilenerf: Exploiting the polygon rasterization pipeline for efficient neural field rendering on mobile architectures. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16569–16578, 2023. 2
- [5] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5939–5948, 2019. 2, 3
- [6] Nianchen Deng, Zhenyi He, Jiannan Ye, Budmonde Duinkharjav, Praneeth Chakravarthula, Xubo Yang, and Qi Sun. Fov-nerf: Foveated neural radiance fields for virtual reality. *IEEE Transactions on Visualization and Computer Graphics*, 28(11):3854–3864, 2022. 1
- [7] Shivam Duggal, Zihao Wang, Wei-Chiu Ma, Sivabalan Manivasagam, Justin Liang, Shenlong Wang, and Raquel Urtasun. Mending neural implicit modeling for 3d vehicle reconstruction in the wild. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1900–1909, 2022. 2
- [8] Stefan Elfwing, Eiji Uchibe, and Kenji Doya. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural Networks*, 107:3–11, 2018. 5
- [9] Rizal Fathony, Anit Kumar Sahu, Devin Willmott, and J Zico Kolter. Multiplicative filter networks. In *International Conference on Learning Representations*, 2021. 2, 5, 9
- [10] Kyle Gao, Yina Gao, Hongjie He, Denning Lu, Linlin Xu, and Jonathan Li. Nerf: Neural radiance field in 3d vision, a comprehensive review. *arXiv preprint arXiv:2210.00379*, 2022. 1, 2
- [11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 6
- [12] Milad Soltany Kadarvish, Hesam Mojtabaei, Hossein Entezari Zarch, Amirhossein Kazerouni, Alireza Morsali, Azra Abtahi, and Farokh Marvasti. Ensemble neural representation networks. *arXiv preprint arXiv:2110.04124*, 2021. 2
- [13] Jaechang Kim, Yunjoo Lee, Seunghoon Hong, and Jungseul Ok. Learning continuous representation of audio for arbitrary scale super resolution. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3703–3707. IEEE, 2022. 2
- [14] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. 2014. 5
- [15] Sylwester Kloczek, Łukasz Maziarka, Maciej Wołczyk, Jacek Tabor, Jakub Nowak, and Marek Śmieja. Hypernetwork functional image representation. In *International Conference on Artificial Neural Networks*, pages 496–510. Springer, 2019. 2, 3
- [16] Alan Lapedes and Robert Farber. Nonlinear signal processing using neural networks: Prediction and system modelling. Technical report, 1987. 2
- [17] Ke Li, Tim Rolff, Susanne Schmidt, Reinhard Bacher, Simone Frintrop, Wim Leemans, and Frank Steinicke. Bringing instant neural graphics primitives to immersive virtual reality. In *2023 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, pages 739–740. IEEE, 2023. 1
- [18] Sicheng Li, Hao Li, Yue Wang, Yiyi Liao, and Lu Yu. Stern-erf: Accelerating nerf rendering via smooth viewpoint trajectory. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20701–20711, 2023. 2
- [19] Tianyang Li, Xin Wen, Yu-Shen Liu, Hua Su, and Zhizhong Han. Learning deep implicit functions for 3d shapes with dynamic code clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12840–12850, 2022. 2
- [20] Beth Logan et al. Mel frequency cepstral coefficients for music modeling. In *Ismir*, volume 270, page 11. Plymouth, MA, 2000. 7
- [21] Yunfan Lu, Zipeng Wang, Minjie Liu, Hongjian Wang, and Lin Wang. Learning spatial-temporal implicit neural representations for event-guided video super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1557–1567, 2023. 1
- [22] Shishira R Maiya, Sharath Girish, Max Ehrlich, Hanyu Wang, Kwot Sin Lee, Patrick Poirson, Pengxiang Wu, Chen Wang, and Abhinav Shrivastava. Nirvana: Neural implicit representations of videos with adaptive networks and autoregressive patch-wise modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14378–14387, 2023. 1
- [23] Julien NP Martel, David B Lindell, Connor Z Lin, Eric R Chan, Marco Monteiro, and Gordon Wetzstein. Acorn: adaptive coordinate networks for neural scene representation. *ACM Transactions on Graphics (TOG)*, 40(4):1–13, 2021. 2
- [24] Ishit Mehta, Michaël Gharbi, Connelly Barnes, Eli Shechtman, Ravi Ramamoorthi, and Manmohan Chandraker. Modulated periodic activations for generalizable local functional representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14214–14223, 2021. 2, 3
- [25] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020. 2, 9
- [26] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view syn-

- thesis. *Communications of the ACM*, 65(1):99–106, 2021. 1
- [27] Amirali Molaei, Amirhossein Aminimehr, Armin Tavakoli, Amirhossein Kazerouni, Bobby Azad, Reza Azad, and Dorit Merhof. Implicit neural representation in medical imaging: A comparative survey. *arXiv preprint arXiv:2307.16142*, 2023. 1
- [28] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *arXiv preprint arXiv:2201.05989*, 2022. 1, 2
- [29] Joseph Ortiz, Alexander Clegg, Jing Dong, Edgar Sucar, David Novotny, Michael Zollhoefer, and Mustafa Mukadam. isdf: Real-time neural signed distance fields for robot perception. *arXiv preprint arXiv:2204.02296*, 2022. 2
- [30] Giambattista Parascandolo, Heikki Huttunen, and Tuomas Virtanen. Taming the waves: sine as activation function in deep neural networks. 2016. 2
- [31] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 165–174, 2019. 3
- [32] Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. D-nerf: Neural radiance fields for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10318–10327, 2021. 2
- [33] Nasim Rahaman, Aristide Baratin, Devansh Arpit, Felix Draxler, Min Lin, Fred Hamprecht, Yoshua Bengio, and Aaron Courville. On the spectral bias of neural networks. In *International Conference on Machine Learning*, pages 5301–5310. PMLR, 2019. 2, 3
- [34] Sameera Ramasinghe and Simon Lucey. Beyond periodicity: Towards a unifying framework for activations in coordinate-mlps. In *European Conference on Computer Vision*, pages 142–158. Springer, 2022. 5, 9
- [35] Daniel Rebain, Mark J Matthews, Kwang Moo Yi, Gopal Sharma, Dmitry Lagun, and Andrea Tagliasacchi. Attention beats concatenation for conditioning neural fields. *arXiv preprint arXiv:2209.10684*, 2022. 2, 3
- [36] Christian Reiser, Songyou Peng, Yiyi Liao, and Andreas Geiger. Kilonerf: Speeding up neural radiance fields with thousands of tiny mlps. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14335–14345, 2021. 2
- [37] Daniel Rho, Junwoo Cho, Jong Hwan Ko, and Eunbyung Park. Neural residual flow fields for efficient video representations. *arXiv preprint arXiv:2201.04329*, 2022. 2
- [38] Vishwanath Saragadam, Daniel LeJeune, Jasper Tan, Guha Balakrishnan, Ashok Veeraraghavan, and Richard G Baraniuk. Wire: Wavelet implicit neural representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18507–18516, 2023. 2, 5, 7, 9, 11
- [39] Vishwanath Saragadam, Jasper Tan, Guha Balakrishnan, Richard G Baraniuk, and Ashok Veeraraghavan. Miner: Multiscale implicit neural representation. In *European Conference on Computer Vision*, pages 318–333. Springer, 2022. 2
- [40] Liyue Shen, John Pauly, and Lei Xing. Nerp: implicit neural representation learning with prior embedding for sparsely sampled image reconstruction. *IEEE Transactions on Neural Networks and Learning Systems*, 2022. 2
- [41] Vincent Sitzmann, Julien Martel, Alexander Bergman, David Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. *Advances in neural information processing systems*, 33:7462–7473, 2020. 1, 2, 3, 7, 9
- [42] Yannick Strümpler, Janis Postels, Ren Yang, Luc Van Gool, and Federico Tombari. Implicit neural representations for image compression. In *European Conference on Computer Vision*, pages 74–91. Springer, 2022. 2
- [43] Matthew Tancik, Pratul Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *Advances in Neural Information Processing Systems*, 33:7537–7547, 2020. 2, 5, 9
- [44] Jiaxiang Tang. Torch-npg: A pytorch implementation of instant-npg, 2022. 11
- [45] Jiaxiang Tang, Xiaokang Chen, Jingbo Wang, and Gang Zeng. Compressible-composable nerf via rank-residual decomposition. *Advances in Neural Information Processing Systems*, 35:14798–14809, 2022. 11
- [46] Radu Timofte, Shuhang Gu, Jiqing Wu, Luc Van Gool, Lei Zhang, Ming-Hsuan Yang, Muhammad Haris, et al. Ntire 2018 challenge on single image super-resolution: Methods and results. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2018. 5, 7, 8
- [47] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 6450–6459, 2018. 7
- [48] Peng Wang, Yuan Liu, Zhaoxi Chen, Lingjie Liu, Ziwei Liu, Taku Komura, Christian Theobalt, and Wenping Wang. F2-nerf: Fast neural radiance field training with free camera trajectories. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4150–4159, 2023. 2
- [49] Qi Wu, David Bauer, Yuyang Chen, and Kwan-Liu Ma. Hyperirn: A fast and predictive hypernetwork for implicit neural representations via knowledge distillation. *arXiv preprint arXiv:2304.04188*, 2023. 2
- [50] Dejia Xu, Peihao Wang, Yifan Jiang, Zhiwen Fan, and Zhangyang Wang. Signal processing for implicit neural representations. *Advances in Neural Information Processing Systems*, 35:13404–13418, 2022. 1
- [51] Gizem Yüce et al. A structured dictionary perspective on implicit neural representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE, 2022. 5