

# SASNet: Spatially-Adaptive Sinusoidal Networks for INRs

Haoan Feng<sup>1</sup>, Diana Aldana<sup>2</sup>, Tiago Novello<sup>2</sup>, Leila De Floriani<sup>1</sup>

<sup>1</sup>University of Maryland, <sup>2</sup> IMPA

## Abstract

Sinusoidal neural networks (SIRENs) are powerful implicit neural representations (INRs) for low-dimensional signals in vision and graphics. By encoding input coordinates with sinusoidal functions, they enable high-frequency image and surface reconstruction. However, training SIRENs is often unstable and highly sensitive to frequency initialization: small frequencies produce overly smooth reconstructions in detailed regions, whereas large ones introduce spurious high-frequency components that manifest as noise in smooth areas such as image backgrounds. To address these challenges, we propose **SASNet**, a Spatially-Adaptive Sinusoidal Network that couples a frozen frequency embedding layer, which explicitly fixes the network’s frequency support, with jointly learned spatial masks that localize neuron influence across the domain. This pairing stabilizes optimization, sharpens edges, and suppresses noise in smooth areas. Experiments on 2D image and 3D volumetric data fitting as well as signed distance field (SDF) reconstruction benchmarks demonstrate that SASNet achieves faster convergence, superior reconstruction quality, and robust frequency localization—assigning low- and high-frequency neurons to smooth and detailed regions respectively—while maintaining parameter efficiency. Code available at [here](#)<sup>1</sup>.

## 1. Introduction

Implicit neural representations (INRs) have emerged as powerful tools for modeling low-dimensional signals in computer vision and graphics [21, 28, 40]. They encode signals implicitly in the weights of neural networks by mapping coordinates directly to signal values. In particular, sinusoidal networks (SIRENs) [41] employ sinusoidal activations, allowing them to capture fine details by modeling high frequencies. These properties make SIRENs particularly well-suited for tasks that require accurate high-frequency reconstruction, such as image fitting [16, 18, 32, 39], super-resolution [7, 35], signed distance field (SDF) modeling [10, 21, 24, 25, 38, 41], and morphing [4, 6, 37].

<sup>1</sup>[https://github.com/Fengyee/SASNet\\_inr](https://github.com/Fengyee/SASNet_inr)

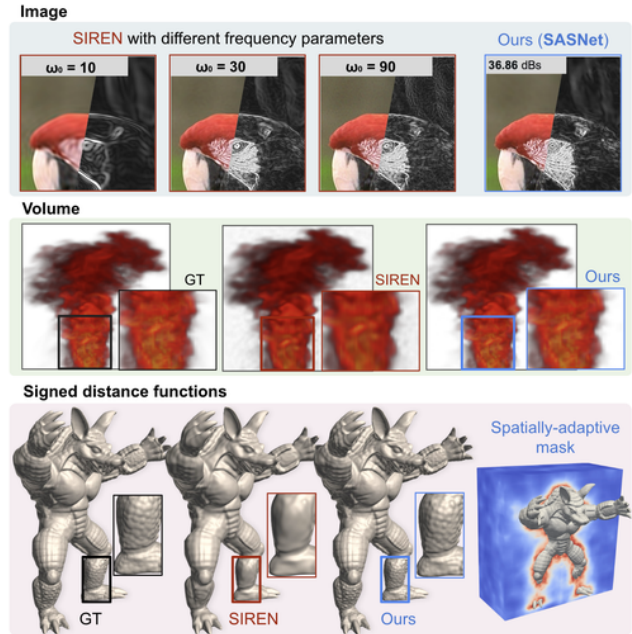


Figure 1. We present **SASNet**, a robust INR that introduces spatial localization into SIRENs, addressing challenges in frequency initialization, hyperparameter selection, and noisy reconstruction. The first row compares SIREN [41] with different frequency parameters  $\omega_0$  and SASNet on a  $\times 16$  super-resolution task. Low  $\omega_0$  values yield blurry reconstructions, while higher  $\omega_0$  values produce ringing artifacts and noisy backgrounds; in contrast, SASNet achieves sharper and cleaner results. Similar results are observed in the 3D volumetric fitting task (second row). Finally, SASNet achieves superior SDF fitting quality (Armadillo’s leg) by learning spatially-adaptive masks that focus neuron activations near the zero-level set (bottom right).

Despite their expressive power, SIRENs are highly sensitive to hyperparameters and provide limited control over frequency support, often leading to reconstruction artifacts. Their performance depends critically on the input frequency parameter  $\omega_0$ , which determines the network’s spectral range. As shown in Fig. 1 (top), a small  $\omega_0$  produces clean but overly smooth reconstructions, while a large  $\omega_0$  improves sharpness but introduces high-frequency noise in smooth regions. We refer to this undesired activation of high-frequency components in low-frequency regions as

*frequency leakage*. Recovering high-frequency details by prolonged training often further destabilizes optimization and causes overfitting, degrading both generalization and derivative accuracy. These localization issues limit the practical use of SIRENs. Grid-based networks [23, 42] alleviate this problem by interpolating spatial features with a hash grid followed by an MLP, but representing fine details still requires very high-resolution grids, increasing memory and computation costs and limiting scalability.

To address these issues, we propose **SASNet** (**S**patially-**A**daptive **S**inusoidal **N**eural **N**etwork), an architecture that introduces robust and localized frequency control by making SIREN activations spatially dependent. SASNet adopts a frozen frequency embedding layer, following Novello et al. [26], to provide explicit control over the network’s spectral support. To achieve frequency localization, SASNet modulates SIREN neuron activations by employing *spatially-adaptive masks* generated from a lightweight hash-grid MLP. These masks are applied to each layer via element-wise multiplication, regulating the influence of different frequency bands and constraining neuron responses to specific spatial regions (see Fig. 1, second row). Both networks are trained jointly, reducing redundancy and overfitting while concentrating neuron contributions in unfitted regions. Fig. 1 showcases SASNet’s results on 2D image fitting and SDF reconstruction tasks, demonstrating significant improvements over prior methods in accuracy, speed, and spatial consistency. Our main contributions are:

- We introduce **SASNet**, which combines a dedicated **frequency embedding layer** [26] with learned spatial modulation to bring localized frequency control to SIRENs.
- We propose a lightweight multi-scale hash-grid MLP that learns **spatially-adaptive masks** to localize the influence of different frequency bands and neuron groups, reducing high-frequency leakage, overfitting in smooth regions, and redundant neuron contributions.
- We show that this design yields more stable optimization and faster convergence than prior methods while improving reconstruction quality and spatial consistency across image fitting, volumetric reconstruction, and SDF tasks.

## 2. Related work

**Implicit neural representations** (INRs) have become fundamental in computer vision and graphics for modeling low-dimensional signals [3, 9, 13, 17, 20, 21, 28, 33, 40, 44]. They encode spatial coordinates directly through neural network parameters, enabling differentiable and resolution-independent signal reconstruction. XINC [27] introduces neuron contribution maps to visualize how individual neurons in INRs contribute to the output. Adopting this view, we propose a three-category taxonomy that classifies INRs into *global*, *local*, and *hybrid* representations.

In **global INRs**, each neuron contributes to the entire spatial domain. Early ReLU-based MLPs exhibit a strong spectral bias [29], favoring low frequencies and struggling to capture high-frequency details [29, 31]. To alleviate this limitation, FFN [43] introduced positional encoding, while alternative activation functions such as sine [41], Gaussian [30], and sinc [36] have also been explored. SIREN [41] is the most widely used INR due to its high representational capacity across diverse signal domains. However, its training is highly sensitive to the frequency parameter  $\omega_0$ . Recent work [25] introduces improved initialization and weight-bounding strategies for explicit spectral control, while FINER [18] proposes a dynamic scaling factor for adaptive frequency modulation. Other approaches mitigate spectral bias via weight reparameterization [39] or extend frequency expressiveness using auxiliary techniques [14, 20]. Despite such advances, global INRs lack spatial localization, which reduces parameter efficiency and often produces blurry or noisy reconstructions (see Fig. 1).

**Local INRs** restrict neuron influence to specific spatial regions, improving scalability and localized detail modeling. Examples include sinc-activated networks [36] and WIRE [35], which employ Gabor wavelets to encode spatial locality but require careful initialization and are prone to overfitting. Inspired by Gaussian splatting [15], recent works [47, 48] use Gaussian kernels for compact image fitting, improving inference efficiency through parameter compression and gradient-based initialization. Although these models mitigate some limitations of global INRs, they lack explicit frequency control, often overfitting local textures, and typically require a larger parameter count.

**Hybrid INRs** integrate the strengths of global and local formulations by combining multi-scale features or decomposing the domain into sub-regions. InstantNGP [23] employs multi-scale hash-grid encodings followed by an MLP for 3D scene representation, often requiring large grids and expensive training. MINER [34] decomposes the domain into hierarchical tiles and iteratively fits residuals. NeuRBF [7] merges fixed frequency embeddings with radial basis functions (RBFs) to improve local fitting, while SAPE [12] expands model capacity using residual-dependent masks that introduce high-frequency embeddings in poorly reconstructed regions. More recently, Walker *et al.* [44] introduce a learned spatial gating mechanism that dynamically selects which levels of a multi-resolution hash grid contribute at each location, enabling spatially-adaptive frequency allocation that reduces artifacts in smooth regions. Despite their advances, these methods can still overfit, show limited expressiveness under compact configurations, and degrade signal derivative accuracy due to ReLU-based activations.

Our **SASNet** addresses these issues by combining the frequency control of SIRENs [26] with hash-grid MLP’s spatial modulation, resulting in a robust INR that achieves frequency localization, reduces overfitting in smooth regions of the signal, converges faster, and delivers superior reconstruction without increasing the number of parameters.

### 3. Method

Our goal is to design a network capable of representing the fine details captured by SIRENs [41] while avoiding the injection of unwanted high-frequencies into smooth regions of the signal. For this, we introduce the *spatially-adaptive sinusoidal network* (SASNet), illustrated in Fig. 2, which integrates a hash-grid MLP that modulates the neurons of a SIREN through learned adaptive masks, enabling localized frequency control. This allows preserving high-frequency detail in complex regions while maintaining smoothness in low-frequency areas, leading to a robust representation.

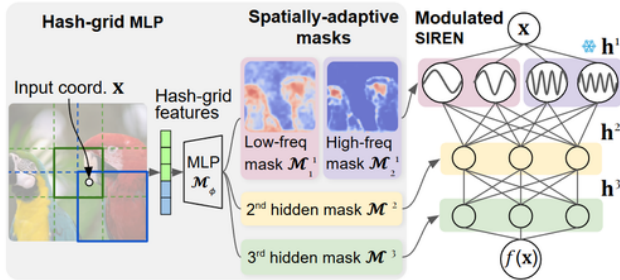


Figure 2. **Overview of SASNet.** A hash-grid encodes the input coordinate  $\mathbf{x}$  into features, which are decoded via a ReLU MLP into spatially-adaptive masks  $\mathcal{M}^i(\mathbf{x})$ . Each mask modulates the corresponding layer of the sinusoidal MLP through an element-wise (Hadamard) product  $\odot$ , localizing neuron influence across the spatial domain. The frozen frequency embedding layer  $\mathbf{h}^1$  fixes the network spectrum, while the masks localize the neuron influence, stabilizing training and improving reconstruction.

#### 3.1. Problem setting

Let  $\{\mathbf{x}_i, \ell_i\}_{i=1}^N$  be samples of an unknown low-dimensional signal  $\ell$ , we aim to learn a neural network  $f_\theta$  that approximates  $\ell$  while avoiding overfitting. This setup follows the standard INR formulation, where the parameters  $\theta$  are implicitly defined by the loss function:

$$\mathcal{L}(\theta) = \frac{1}{N} \sum_{i=1}^N \|f_\theta(\mathbf{x}_i) - \ell_i\|_2^2 + \lambda \mathcal{R}(\theta), \quad (1)$$

where the first term enforces data fitting, and the regularization term  $\mathcal{R}(\theta)$  constrains  $f_\theta$  to satisfy some property. For instance, for signed distance functions (SDFs) [11],  $\mathcal{R}(\theta)$  enforces the Eikonal constraint  $|\nabla f_\theta| = 1$ .

**Sinusoidal MLPs.** SIREN [41] is a common parameterization for  $f_\theta$ . It is defined as the composition of  $d + 1$  layers

$$f_\theta(\mathbf{x}) = \mathbf{L} \circ \mathbf{S}^d \circ \dots \circ \mathbf{S}^0(\mathbf{x}), \quad (2)$$

where each layer is given by  $\mathbf{S}^i(\mathbf{x}) = \sin(\mathbf{W}^i \mathbf{x} + \mathbf{b}^i)$ , parameterized by a weight matrix  $\mathbf{W}^i \in \mathbb{R}^{n_{i+1} \times n_i}$  and bias  $\mathbf{b}^i \in \mathbb{R}^{n_{i+1}}$ . The first layer  $\mathbf{S}^0$  maps the input coordinates  $\mathbf{x}$  into sinusoids  $\sin(\omega \mathbf{x} + \varphi)$ , where  $\omega := \mathbf{W}^0$  are the *input frequencies*, and  $\varphi := \mathbf{b}^0$  are the *phase shifts*. The final layer  $\mathbf{L}$  is an affine transformation.

The network’s spectrum is defined by the input frequencies  $\omega$  (see Sec. 3.2). Small  $\omega$  limit the representable spectrum, resulting in *underfitting* and overly smooth reconstructions that fail to capture fine details. Conversely, a large  $\omega$  allows sharper reconstructions but introduces *overfitting* and high-frequency leakage into smooth regions, leading to noisy artifacts. This trade-off arises because SIREN neurons are globally coupled; each parameter contributes to the entire domain, as we show in Theorem 1.

**Spatially-adaptive networks.** To overcome these limitations, we propose to modulate SIREN activations with *spatially adaptive masks* produced by a lightweight hash-grid MLP  $\mathcal{M}_\phi$ . This mechanism localizes frequency contributions across space by constraining each neuron’s effective spatial support. Formally, we replace each sinusoidal layer  $\mathbf{S}^i$  with its spatially modulated counterpart:

$$\tilde{\mathbf{S}}^i = \mathcal{M}^i \odot \mathbf{S}^i, \quad \text{where } \mathcal{M}^i(\mathbf{x}) \in [0, 1]^{n_i} \quad (3)$$

are the learned spatial masks and  $\odot$  denotes the Hadamard product. Substituting the modulated neurons from (3) into the MLP definition in (2) gives rise to **SASNet**, a model designed to mitigate overfitting and improve the parameter efficiency of SIRENs via *frequency localization*.

By construction, SASNet consists of a frequency embedding layer (Sec. 3.2), spatially-adaptive masks (Sec. 3.3), and a joint training scheme that optimizes the masks via a multi-scale hash-grid MLP strategy. This design allows SASNet to selectively activate high-frequency components where needed while maintaining smoothness and stability across the signal domain.

#### 3.2. Frequency embedding layer

We show that the SIREN spectrum is fully determined by its *input frequencies*. Following the trigonometric identity in [26], each sinusoidal neuron expands as a sum of sines whose frequencies are integer linear combinations of the input frequencies. Hence, the neuron modulation in (3) serves to spatially localize these generated frequency components.

Precisely, recall that an *ij-neuron*  $h_j^i(\mathbf{x})$  of a SIREN

refers to the  $j$ -th neuron of the  $i$ -th layer  $\mathbf{h}^i(\mathbf{x})$ :

$$h_j^i(\mathbf{x}) = \sin\left(\sum_{k=1}^n W_{jk}^i \underbrace{\sin(\mathbf{y}_k^{i-1})}_{h_k^{i-1}(\mathbf{x})} + b_j^i\right), \quad (4)$$

where  $\mathbf{y}^{i-1}$  denotes  $\mathbf{h}^{i-1}(\mathbf{x})$  before activation. Note that weight  $W_{jk}^i$  determines the influence of neuron  $h_k^{i-1}$  in shaping the next layer. Novello et al. [26] shows that  $h_j^i(\mathbf{x})$  expanded as follows:

**Thm 1** *The neuron  $h_j^i$  admits the following expansion:*

$$h_j^i(\mathbf{x}) = \sum_{\mathbf{k} \in \mathbb{Z}^{n_i}} \alpha_{\mathbf{k}} \sin(\langle \mathbf{k}, \mathbf{y} \rangle + b_j^i), \quad (5)$$

where  $\alpha_{\mathbf{k}} = \prod_l J_{k_l}(W_{jl}^i)$  is the product of Bessel functions.

Theorem 1 shows that the frequencies generated in the second layer  $\mathbf{h}^2(\mathbf{x})$  are integer linear combinations  $\langle \mathbf{k}, \omega \rangle$  of the input frequencies  $\omega$ . Moreover, this result generalizes to deeper layers by inductively applying Theorem 1 across the network depth. Hence, the input frequencies  $\omega$  determine the full network spectrum, given by  $\langle \mathbf{k}, \omega \rangle$ .

**Input frequency initialization.** We initialize  $\omega$  so that the generated frequencies  $\langle \mathbf{k}, \omega \rangle$  cover a broad spectral range while preserving accurate low-frequency reconstruction. We define the network over periodic functions and ensure the samples span a complete period. Since such functions can be written as a Fourier series, *i.e.*, sums of sine terms with integer frequencies, Theorem 1 shows that SIRENs can be formulated in this form. Because real-world signals typically have larger low-frequency amplitudes, we sample most elements of  $\omega$  from the low-frequency range  $[-\mathcal{L}, \mathcal{L}]^{n_0}$  and the rest from a higher-frequency band. This defines a fixed *frequency embedding layer*, while all other weights use the standard initialization scheme [41]. Details are provided in the supplementary (Supp. Sec 3.1).

Unlike standard SIRENs, where frequencies evolve during training, our formulation keeps the spectrum fixed while learning only the amplitudes  $\alpha_{\mathbf{k}}$ . This approach stabilizes optimization and accelerates convergence [26].

### 3.3. Spatially-adaptive masks

Theorem 1 shows that the amplitudes  $\alpha_{\mathbf{k}}$  of the frequencies generated by each hidden neuron  $h_j^i(\mathbf{x})$  are independent of the spatial coordinate  $\mathbf{x}$ . Consequently, high-frequency components may contribute *globally* across the domain, including smooth regions, motivating the introduction of *spatially adaptive masks* to localize these frequency contributions. To enable spatial control, we multiply each neuron  $h_j^i(\mathbf{x})$  by a mask  $\mathcal{M}_j^i(\mathbf{x}) \in [0, 1]$ , modulating its influence over the domain. Thus, SIREN neurons are replaced by

$\tilde{h}_j^i(\mathbf{x}) = h_j^i(\mathbf{x}) \mathcal{M}_j^i(\mathbf{x})$ , and substituting into the expansion in (5) yields the spatially localized form:

$$\tilde{h}_j^i(\mathbf{x}) = \sum_{\mathbf{k} \in \mathbb{Z}^{n_i}} \alpha_{\mathbf{k}} \mathcal{M}_j^i(\mathbf{x}) \sin(\langle \mathbf{k}, \mathbf{y} \rangle + b_j^i). \quad (6)$$

Thus, the proposed modulation makes the frequency amplitudes  $\alpha_{\mathbf{k}} \mathcal{M}_j^i(\mathbf{x})$  spatially dependent, allowing high-frequency responses to be activated only where needed. Fig. 3 shows neurons of SIREN and SASNet. Notice that the mask  $\mathcal{M}^1(\mathbf{x})$  applied to the frequency embedding layer  $\mathcal{M}^1(\mathbf{x}) \odot \sin(\omega \mathbf{x} + \varphi)$  directly controls which frequencies appear in each region, as observed in our experiments.

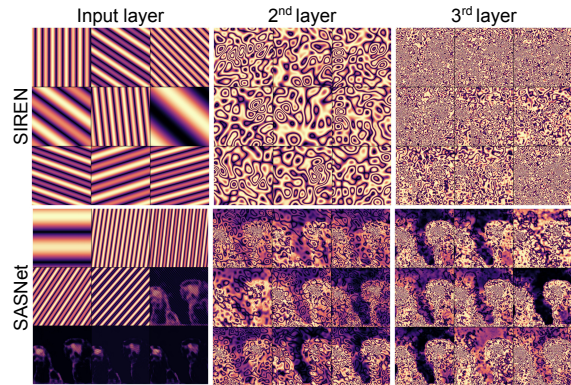


Figure 3. **Neuron visualization** for SIREN and SASNet on the Kodak “two macaws” image. Each column shows 9 neuron activation maps from the input, 2<sup>nd</sup>, and 3<sup>rd</sup> layers. While SIREN exhibits global activations and high-frequency interference, SASNet produces spatially localized and semantically aligned patterns.

**Hash-grid network.** We parametrize the spatially adaptive masks  $\mathcal{M}_\phi$  using a multi-scale hash-grid network [23], which allows the masks to be jointly optimized with the SIREN parameters. To avoid the memory and computational cost of learning a different mask for every neuron  $h_j^i(\mathbf{x})$ , we partition the neurons of each layer  $i$  into  $n$  groups. Each group is assigned a single mask  $\mathcal{M}_j^i(\mathbf{x})$ , which is then *broadcasted* to all neurons within that group, modulating their outputs. Additionally, each input neuron  $\sin(\omega_i \mathbf{x} + \varphi_i)$  is grouped according to its maximum frequency  $\max(|\omega_i|)$ . Specifically, neurons satisfying  $\max(|\omega_i|) < \mathcal{L}$  are assigned to a low-frequency group, while the remaining neurons are partitioned into groups based on the magnitude of  $\max(|\omega_i|)$ . In contrast, neurons in the hidden layers are evenly divided into  $n$  groups.

**Training.** To enforce sparsity, we sum a term to the loss  $\mathcal{L}(\theta)$  that encourages the masks to take values close to zero:

$$\mathcal{L}(\theta) + \lambda \int |\mathcal{M}_\phi(\mathbf{x})|_1 d\mathbf{x}.$$

where  $\lambda$  controls the strength of the regularization. Computed over sampled spatial coordinates  $\mathbf{x}$ , this extra term penalizes the mask values  $\mathcal{M}_j^i(\mathbf{x})$ , encouraging the usage of high-frequency components only when necessary.

## 4. Experiments

We evaluate SASNet on three tasks: 2D image fitting (Sec. 4.1), 3D volumetric data reconstruction (Sec. 4.2), and signed distance function (SDF) reconstruction (Sec. 4.3). We also conduct an ablation study (Sec. 4.4) to assess the contribution of each component in our proposed method.

### 4.1. Image fitting

We evaluate SASNet against state-of-the-art INRs with open-source implementations (summarized in Tab. 1) on the task of image fitting. Each  $512^2$  RGB image of DIV2K [2] is fitted by an INR and compared to the ground truth (GT) using the *peak signal-to-noise ratio* (PSNR) and *structural similarity index measure* (SSIM) [45].

However, these metrics fail to fully capture noisy reconstruction artifacts caused by excessively high-frequency components, particularly in smooth regions. To detect such regions, we identify edges using Canny edge detector [5] and dilate them with a disk kernel, resulting in a binary mask that separates edge and smooth areas, i.e.,  $\mathbb{1}_{\text{edge}}$  and  $\mathbb{1}_{\text{smooth}}$ . To quantify noise, we compute the discrepancy in gradient norms between the input image and the model output in the smooth regions, resulting in the **noisiness** metric:

$$\frac{1}{N_{\text{smooth}}} \sum_{i=1}^N \mathbb{1}_{\text{smooth}}(\mathbf{x}_i) \left| \|\nabla f_{\text{gray}}(\mathbf{x}_i)\|_2 - \|\nabla f_{\text{gray}}(\mathbf{x}_i)\|_2 \right|,$$

where  $\mathbf{x}_i$  are the image pixels,  $N_{\text{smooth}}$  is the number of pixels in the smooth regions, and  $f_{\text{gray}}$  and  $\hat{f}_{\text{gray}}$  denote the ground-truth and predicted grayscale images. The INR gradient  $\nabla f_{\text{gray}}$  is computed analytically using automatic differentiation, while  $\nabla \hat{f}_{\text{gray}}$  is estimated using a second-order finite-difference method. Finally, since the edge-region reconstruction quality is closely tied to image sharpness, we evaluate it using the **edge-region PSNR**:

$$\text{PSNR}_{\text{edge}} = -10 \log_{10} (\text{MSE}(\mathbb{1}_{\text{edge}} \odot \hat{f}, \mathbb{1}_{\text{edge}} \odot f_{\theta})).$$

In all image experiments, the compared methods are trained using their recommended hyperparameter settings.

#### 4.1.1. Quantitative comparisons

Tab. 1 compares SASNet with classical and state-of-the-art INRs on the image fitting task, showing that our method consistently achieves high reconstruction accuracy while maintaining very low noise. All models are trained for 5K epochs, except GaussianImage, which requires 50K epochs to converge. To ensure a fair comparison, all architectures are configured to have a similar number of parameters. SASNet’s spatial mask generation branch introduces only modest computational and GPU memory overhead, which is analyzed in the supplementary (Supp. Sec. 4).

SIREN and NeuRBF reduce noise but underfit fine details, leading to lower PSNR. FFN, WIRE, and SAPE show weaker accuracy, either due to limited frequency capacity or unstable local modeling. FINER provides sharper reconstructions but introduces substantial high-frequency noise. In contrast, SASNet achieves the best overall performance across all metrics. Its spatially adaptive masks avoid unnecessary high-frequency activations in smooth areas while enhancing detail reconstruction where needed, leading to the highest  $\text{PSNR}_{\text{edge}}$  and the lowest noisiness scores. Additional results are provided in the supplementary, where we further evaluate SASNet on medical CT slices and head-shot images, showing that the method generalizes effectively across diverse image modalities (Supp. Sec. 5).

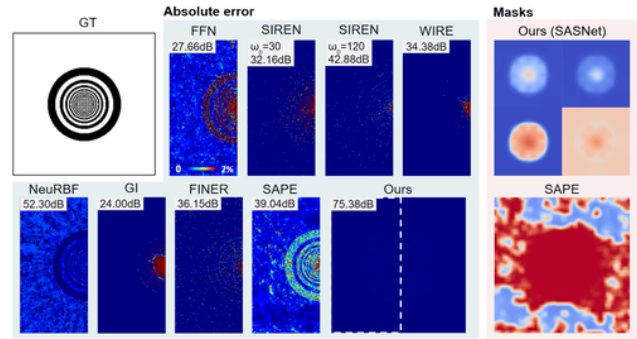


Figure 4. **Comparison of reconstruction error maps and learned masks on a toy image.** The leftmost column shows the GT, and the absolute error maps for all methods. SAPE and SASNet masks are shown on the right. For SASNet, we visualize two frequency-embedding group masks (top) and two hidden-layer group masks (bottom), showing how the model activates neurons only around high-frequency regions. GI refers to GaussianImage.

#### 4.1.2. Qualitative comparisons

To evaluate how effectively SASNet adapts to the spatial frequency distribution of a signal, we test all methods from Tab. 1 on a toy image containing sharp edges, flat regions, and a wide spectrum of frequencies (see GT in Fig. 4). All models use an architecture with approximately 11K parameters. Fig. 4 shows absolute error maps and the masks learned by SAPE and SASNet. Most baselines—including SIREN, FFN, NeuRBF, SAPE, FINER, and WIRE—exhibit ringing artifacts and fail to capture the high-frequency structure in the central region, often producing noisy backgrounds. GaussianImage and WIRE better suppress noise but still struggle to fit sharp details. In contrast, SASNet delivers substantially cleaner reconstructions, achieving a +23 dB PSNR gain over the second-best method by accurately capturing edges while suppressing noise in flat regions. The learned masks further highlight the difference: SAPE’s masks are noisy and spatially inconsistent, whereas SASNet produces well-localized, multi-frequency masks that activate only in high-frequency areas.

Table 1. **Quantitative evaluation of image fitting** on DIV2K [2]. The **best** and **runner-up** values are highlighted in color boxes. **Act func.** denotes the activation function, **FE** indicates the use of a frequency or positional embedding technique, and **BC** denotes an explicit control of the output signal’s band limit, **Type** classifies INRs between “Global”, “Hybrid”, and “Local” localization methods.

Methods	PSNR $\uparrow$	SSIM $\uparrow$	PSNR <sub>edge</sub> $\uparrow$	Noisiness $\downarrow$	Act func.	FE	BC	Type
SIREN [41]	34.64	0.959	31.37	2.525	sin			Global
FFN [43]	31.92	0.910	30.20	9.873	ReLU	✓		Global
SAPE [12]	32.57	0.932	30.31	3.128	ReLU	✓	✓	Hybrid
WIRE [35]	28.76	0.831	25.73	10.324	wavelet			Local
FINER [18]	37.52	0.967	34.02	4.617	$\sin((1+ x )x)$			Global
GaussianImage [47]	37.38	0.973	33.25	-	-			Local
NeuRBF [7]	36.33	0.959	33.92	6.975	ReLU	✓		Hybrid
SASNet (Ours)	39.82	0.979	36.81	2.105	sin	✓	✓	Hybrid

Fig. 5 evaluates the generalization capability of SASNet in a Kodak image. We consider two scenarios: An image patch generated by a trained INR that takes as inputs the trained coordinates shifted by 1/8 of pixel size, and zoom-in  $\times 16$  computed by evaluating the INRs between the training coordinates.

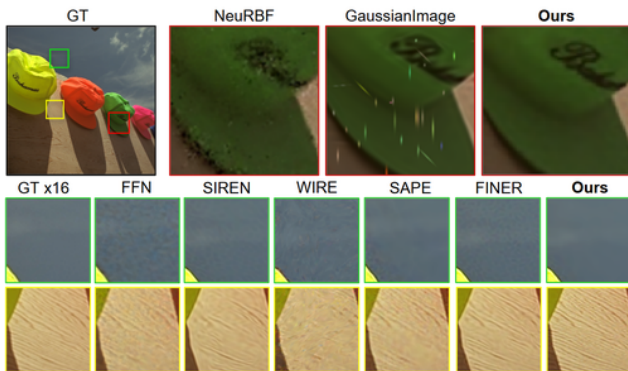


Figure 5. **Qualitative comparison of different methods under pixel shifting and  $\times 16$  super-resolution.** Red rectangles show the zoom-in view of pixel-shifting results: each model is queried at coordinates shifted by 1/8 pixel along both axes to reveal interpolation artifacts. Green rectangles mark smooth background regions, while yellow rectangles highlight high-frequency regions used to assess super-resolution. The bottom rows show the  $\times 16$  reconstructions of these patches, showing that SASNet preserves sharp details in high-frequency areas while avoiding noise in smooth regions.

The top row in Fig. 5 (in red) compares NeuRBF, GaussianImage, and SASNet for the first scenario. The noisy inference result of NeuRBF indicates severe overfitting, while the discrete nature and limited number of Gaussian kernels in GaussianImage lead to special rasterization artifacts. In the bottom row, we evaluate the super-resolution of several SOTA methods in a flat region (in green) and a texture-rich region (yellow squares). SASNet is the only method capable of generalizing in high-frequency regions while preventing unwanted high-frequency noise in smooth areas.

### 4.1.3. Model scalability

We evaluate the scalability of different INRs by varying the number of their trainable parameters and measuring the corresponding PSNR on the Kodak dataset [1]. Model capacity is increased according to each method’s architecture: hidden-layer widths for all INRs, number of Gaussians for GaussianImage, and for SASNet only the SIREN backbone is widened while the hash-grid and mask decoder remain fixed. Fig. 6 shows that NeuRBF [7], FINER [18], and SASNet all benefit from increased capacity, whereas GaussianImage [47] saturates and eventually degrades beyond a certain threshold. SIREN, FFN, SAPE, and WIRE show weaker scaling behavior, with slower PSNR gains or early performance plateaus. Across the entire parameter range, SASNet consistently achieves the highest image-fitting accuracy, highlighting that spatially adaptive masking provides a scalable and parameter-efficient mechanism for capturing complex frequency structures.

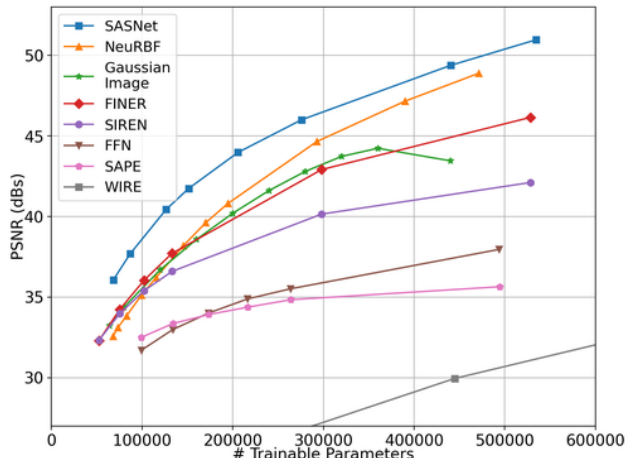


Figure 6. **Model scalability comparison.** Image-fitting accuracy (PSNR) as a function of parameters. While all methods improve with increased capacity, SASNet shows the strongest scalability and maintains the best performance across all model sizes.

## 4.2. Volumetric data representation

We further evaluate the generality of SASNet on 3D volumetric fields, a setting where INRs must represent both smooth global structures and fine high-frequency details. Following the evaluation protocol of FINER [18], we use a smoke-density field  $\rho : \mathbb{R}^3 \rightarrow \mathbb{R}$  from the *ScalarFlow* dataset [8], selecting 8 volumes from a (100, 178, 100) grid. We compare SASNet against five representative INRs: PEMLP [22], SIREN [41], Gauss [30], WIRE [35], and FINER [18]. All models are trained for 5K epochs using the same architecture (3 layers with 256 neurons), except SASNet which uses 224 neurons per layer to keep the parameter count comparable. Tab. 2 reports quantitative results, showing that SASNet outperforms all baselines, achieving the highest PSNR and SSIM, with a noticeable margin over strong methods such as WIRE and FINER.

Table 2. **Quantitative comparison on volumetric data reconstruction** from *ScalarFlow*. The **best** and **runner-up** results are highlighted. SASNet achieves the best reconstruction quality across all models, with the highest PSNR and SSIM.

	PEMLP	Gauss	SIREN	FINER	WIRE	SASNet
PSNR $\uparrow$	40.42	47.11	47.67	49.77	53.06	55.92
SSIM $\uparrow$	0.9434	0.9920	0.9948	0.9959	0.9963	0.9973

Qualitative comparisons in Fig. 7 further illustrate these improvements, showing that SIREN, Gauss, and FINER reconstruct coarse structures but suffer from noise in empty regions due to global frequency leakage. WIRE preserves more high-freq details yet introduces artifacts in smooth areas. In contrast, SASNet produces the sharpest reconstruction while maintaining a clean background, capturing thin filaments and turbulent detail without amplifying noise.

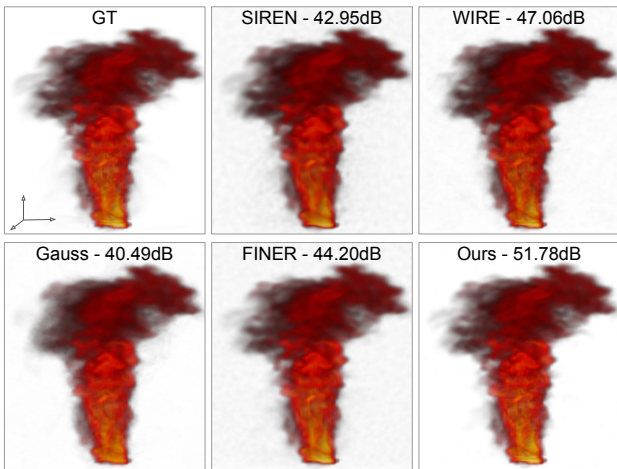


Figure 7. **Volumetric data reconstruction.** SASNet recovers fine-scale structures with sharper detail and reduces background noise more effectively than all baselines.

## 4.3. Signed distance functions

Signed distance functions (SDFs) are fundamental implicit representations used in geometry processing, graphics, and simulation. Recent works show that INRs provide a powerful alternative to grid-based SDFs, preserving differentiability while improving compactness and fidelity [24, 28]. Thus, learning SDFs provides a robust benchmark for evaluating the stability and expressiveness of INR architectures.

**Setup.** Following the open-source FINER pipeline [18], we evaluate four shapes with pre-sampled surface points and estimated normals. All baselines use the same architecture (3 layers  $\times$  256 neurons), while SASNet uses 224 neurons to keep the parameter count comparable. Training uses the coarse-to-fine SDF loss [46], sampling 10K points per iteration for 200K iterations. For evaluation, we extract the zero-level set using Marching Cubes [19] with  $512^3$  grid and report Chamfer distance and IoU.

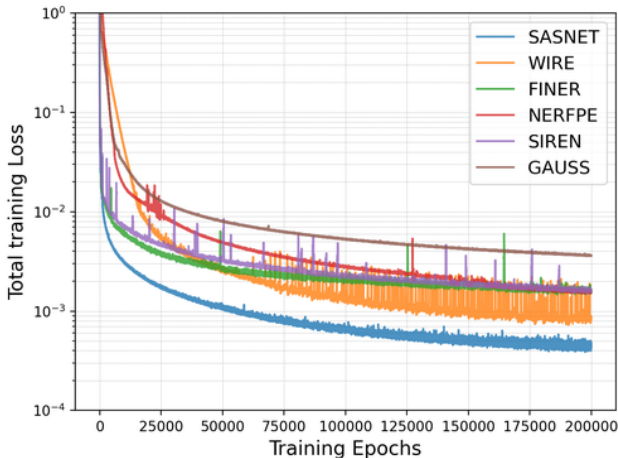


Figure 8. **Training losses for the Armadillo SDF fitting task** on a log-scale, shown as moving averages with a window size of 35. SASNet reaches the final loss attained by the baselines within the first  $\sim 35$ K iterations and achieves the lowest overall loss.

As shown in Tab. 3, SASNet achieves the best reconstruction accuracy across all shapes and metrics. The fixed frequency embedding enables SASNet to represent high-frequency components of the SDF, capturing sharp curvature and intricate geometric features—particularly on complex shapes such as the *Thai Statue*. At the same time, SASNet produces smooth, artifact-free fields in empty regions, avoiding the spurious surfaces that often arise from global INRs. SASNet also converges significantly faster: as illustrated in Fig. 8, it reaches FINER’s final loss level within roughly 35K iterations, and continues improving thereafter. This behavior confirms that the combination of spatially adaptive masking and fixed frequency embedding stabilizes the optimization and accelerates convergence.

Fig. 9 shows reconstruction of the Stanford Dragon model, where SASNet recovers sharp geometric detail

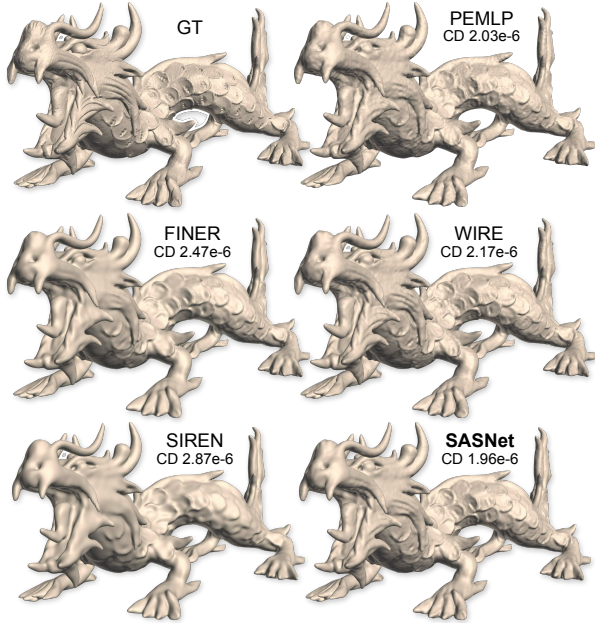


Figure 9. **Qualitative comparison on the SDF reconstruction task** for the *Stanford Dragon*. SASNet produces the most accurate geometry, achieving the lowest Chamfer distance.

while maintaining clean, smooth regions, unlike other INRs, which exhibit noise, blurring, or missing structure. In addition, in Fig. 10, we visualize a cross-section of the learned mask for the highest-frequency band of the frequency embedding layer, showing that SASNet activates those neurons only near the object surface; full multi-view mask visualizations are provided in Supp. Sec. 5.2.

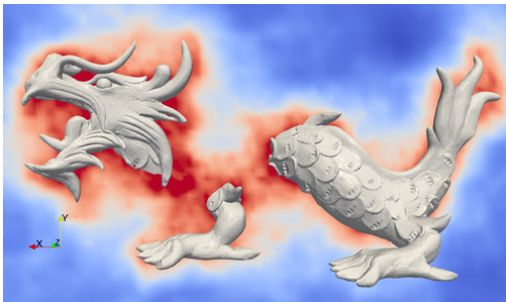


Figure 10. **Sliced 3D mask visualization** of SASNet’s learned high-frequency spatial mask using a cool-warm color map.

#### 4.4. Ablation studies

Our procedure introduces two key components to enhance the performance of SIRENs: a frequency embedding layer and spatially adaptive masks for neuron groups. In Tab. 4 we evaluate the contribution of each component to the baseline SIREN architecture. Specifically, we consider adding the frequency embedding layer initialization (FEmbed.) and masking the input layer ( $\mathbf{h}^1$ ), hidden layers ( $\mathbf{h}^2$ ,  $\mathbf{h}^3$ ), or both.

While initializing the frequency embedding layer provides the most significant improvement ( $\sim 4$  dB), addi-

Table 3. **Quantitative comparison on SDF reconstruction.** The best and runner-up results are highlighted. SASNet achieves the lowest Chamfer distance and highest IoU on all shapes, demonstrating superior geometric accuracy.

Methods	Armadillo	Dragon	Lucy	Thai Statue	Avg.	
Chamfer↓	PEMLP	3.465e-06	2.039e-06	2.387e-06	4.460e-06	3.088e-06
	Gauss	7.283e-06	1.304e-05	1.178e-05	1.162e-05	1.093e-05
	SIREN	3.599e-06	2.871e-06	2.405e-06	4.381e-06	3.314e-06
	WIRE	3.274e-06	2.172e-06	2.419e-06	3.296e-06	2.790e-06
	FINER	4.040e-06	2.477e-06	2.536e-06	4.201e-06	3.313e-06
	SASNet	3.368e-06	1.964e-06	1.927e-06	3.090e-06	2.587e-06
IoU↑	PEMLP	0.9878	0.9803	0.9677	0.9561	0.9730
	Gauss	0.9806	0.9561	0.9581	0.9418	0.9592
	SIREN	0.9872	0.9694	0.9716	0.9561	0.9711
	WIRE	0.9910	0.9747	0.9693	0.9635	0.9746
	FINER	0.9837	0.9728	0.9671	0.9518	0.9689
	SASNet	0.9891	0.9809	0.9813	0.9654	0.9792

Table 4. **Ablation study of SASNet for image fitting task** with different components enabled. **FEmbed.** stands for initializing the frequency embedding layer. **Masks ( $\mathbf{h}^i$ )** stands for masking neuron groups in layer  $i$ .

FEmbed.	Masks			PSNR↑	SSIM↑
	$\mathbf{h}^1$	$\mathbf{h}^2$	$\mathbf{h}^3$		
				36.00	0.921
✓				39.98	0.953
✓	✓			41.30	0.958
✓	✓	✓		41.44	0.959
✓	✓		✓	41.75	0.963
✓		✓		41.08	0.962
✓			✓	40.56	0.960
	✓	✓	✓	37.10	0.932
✓	✓	✓	✓	<b>42.03</b>	<b>0.966</b>

tional masking strategies further improve accuracy. A qualitative comparison showing better reconstruction of high-frequency regions and background noise reduction is presented in the supplementary (Supp. Sec. 5.5).

## 5. Conclusion

We presented SASNet, a novel *Spatially-Adaptive Sinusoidal Network* that addresses key limitations of INRs. By combining a fixed frequency embedding layer with jointly optimized spatially-adaptive masks, SASNet improves frequency localization, enables explicit control over SIREN’s spectral support, and suppresses frequency leakage. Experiments across various benchmarks show that SASNet consistently improves reconstruction quality while reducing noise artifacts, outperforming existing INR architectures in both quantitative and qualitative evaluations. Its gains become smaller when the model is heavily over-parameterized or the target signal is simple to fit. Future work includes reducing the overhead of spatial mask generation and exploring alternative initialization strategies.

## Acknowledgment

This work has been partially supported by the US National Science Foundation under grant numbers IIS-1910766 and IIS-2114451, and Google.

## References

- [1] Kodak lossless true color image suite. <http://https://r0k.us/graphics/kodak/>, 1999. Accessed: 2024-09-01. 6
- [2] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, July 2017. 5, 6
- [3] Yizhak Ben-Shabat, Chamin Hewa Koneputugodage, Sameera Ramasinghe, and Stephen Gould. Neural experts: Mixture of experts for implicit neural representations. *Advances in Neural Information Processing Systems*, 37: 101641–101670, 2024. 2
- [4] Arthur Bizzi, Matias Grynberg, Vitor Matias, Daniel Perazzo, João Paulo Lima, Luiz Velho, Nuno Gonçalves, João Pereira, Guilherme Schardong, and Tiago Novello. Flowing: Implicit neural flows for structure-preserving morphing. In *Conference on Neural Information Processing Systems (NeurIPS)*, volume 38, 2025. 1
- [5] John Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (6):679–698, 1986. 5
- [6] Tung-I Chen, Dae Yeol Lee, Guan-Ming Su, Mohammad Hajjesmaili, and Ramesh K Sitaraman. Nivm: Real-time view morphing via neural implicit function. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pages 8418–8427, 2025. 1
- [7] Zhang Chen, Zhong Li, Liangchen Song, Lele Chen, Jingyi Yu, Junsong Yuan, and Yi Xu. Neurbf: A neural fields representation with adaptive radial basis functions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4182–4194, 2023. 1, 2, 6
- [8] Marie-Lena Eckert, Kiwon Um, and Nils Thuerey. Scalarflow: a large-scale volumetric data set of real-world scalar transport flows for computer animation and machine learning. *ACM Transactions on Graphics (TOG)*, 38(6):1–16, 2019. 7
- [9] Morsali et al. Staf: Sinusoidal trainable activation functions for implicit neural representation. *arXiv*, 2025. 2
- [10] Haoan Feng, Xin Xu, and Leila De Floriani. Impliciterrain: a continuous surface model for terrain data analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshop*, pages 899–909, 2024. 1
- [11] Amos Gropp, Lior Yariv, Niv Haim, Matan Atzmon, and Yaron Lipman. Implicit geometric regularization for learning shapes. *arXiv preprint arXiv:2002.10099*, 2020. 3
- [12] Amir Hertz, Or Perel, Raja Giryes, Olga Sorkine-Hornung, and Daniel Cohen-Or. Sape: Spatially-adaptive progressive encoding for neural optimization. *Advances in Neural Information Processing Systems*, 34:8820–8832, 2021. 2, 6
- [13] Suizhi Huang, Xingyi Yang, Hongtao Lu, and Xinchao Wang. Few-shot implicit function generation via equivariance. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 16262–16272, 2025. 2
- [14] Amirhossein Kazerouni, Reza Azad, Alireza Hosseini, Dorit Merhof, and Ulas Bagci. Incode: Implicit neural conditioning with prior knowledge embeddings. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1298–1307, 2024. 2
- [15] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4), July 2023. URL <https://repo-sam.inria.fr/fungraph/3d-gaussian-splatting/>. 2
- [16] Jinseok Kim and Tae-Kyun Kim. Arbitrary-scale image generation and upsampling using latent diffusion model and implicit neural decoder. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9202–9211, 2024. 1
- [17] Namhoon Kim and Sara Fridovich-Keil. Grids often outperform implicit neural representations. *NeurIPS*, 2025. 2
- [18] Zhen Liu, Hao Zhu, Qi Zhang, Jingde Fu, Weibing Deng, Zhan Ma, Yanwen Guo, and Xun Cao. Finer: Flexible spectral-bias tuning in implicit neural representation by variable-periodic activation functions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2713–2722, 2024. 1, 2, 6, 7
- [19] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *Seminal graphics: pioneering efforts that shaped the field*, pages 347–353. 1998. 7
- [20] 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
- [21] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4460–4470, 2019. 1, 2
- [22] 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. *Communications of the ACM*, 65(1):99–106, 2021. 7
- [23] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics*, 41(4):1–15, 2022. 2, 4
- [24] Tiago Novello, Guilherme Schardong, Luiz Schirmer, Vinícius da Silva, Hélio Lopes, and Luiz Velho. Exploring differential geometry in neural implicit. *Computers & Graphics*, 108:49–60, 2022. ISSN 0097-8493. doi: <https://doi.org/10.1016/j.cag.2022.09.003>. URL <https://www.sciencedirect.com/science/article/pii/S0097849322001649>. 1, 7

- [25] Tiago Novello, Vinícius da Silva, Guilherme Schardong, Luiz Schirmer, Hélio Lopes, and Luiz Velho. Neural implicit surface evolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 14279–14289, October 2023. 1, 2
- [26] Tiago Novello, Diana Aldana, Andre Araujo, and Luiz Velho. Tuning the frequencies: Robust training for sinusoidal neural networks, 2025. URL <https://arxiv.org/abs/2407.21121>. 2, 3, 4
- [27] Namitha Padmanabhan, Matthew Gwilliam, Pulkit Kumar, Shishira R Maiya, Max Ehrlich, and Abhinav Shrivastava. Explaining the implicit neural canvas: Connecting pixels to neurons by tracing their contributions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10957–10967, 2024. 2
- [28] 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. 1, 2, 7
- [29] Nasim Rahaman, Aristide Baratin, Devansh Arpit, Felix Dräxler, Min Lin, Fred A. Hamprecht, Yoshua Bengio, and Aaron C. Courville. On the spectral bias of neural networks. In *International Conference on Machine Learning*, 2018. 2
- [30] 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. 2, 7
- [31] Sameera Ramasinghe, Lachlan E MacDonald, and Simon Lucey. On the frequency-bias of coordinate-mlps. *Advances in Neural Information Processing Systems*, 35:796–809, 2022. 2
- [32] Pradyumna Reddy, Zhifei Zhang, Zhaowen Wang, Matthew Fisher, Hailin Jin, and Niloy Mitra. A multi-implicit neural representation for fonts. *Advances in Neural Information Processing Systems*, 34:12637–12647, 2021. 1
- [33] Reza Rezaeian, Moein Heidari, Reza Azad, Dorit Merhof, Hamid Soltanian-Zadeh, and Ilker Hacihaliloglu. S12a-inr: Single-layer learnable activation for implicit neural representation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 26065–26074, 2025. 2
- [34] Vishwanath Saragadam, Jasper Tan, Guha Balakrishnan, Richard G Baraniuk, and Ashok Veeraraghavan. Miner: Multiscale implicit neural representations. In *European Conference on Computer Vision*, 2022. 2
- [35] 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. 1, 2, 6, 7
- [36] Hemanth Saratchandran, Sameera Ramasinghe, Violetta Shevchenko, Alexander Long, and Simon Lucey. A sampling theory perspective on activations for implicit neural representations. *arXiv preprint arXiv:2402.05427*, 2024. 2
- [37] Guilherme Schardong, Tiago Novello, Hallison Paz, Iurii Medvedev, Vinícius Da Silva, Luiz Velho, and Nuno Gonçalves. Neural implicit morphing of face images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7321–7330, 2024. 1
- [38] Luiz Schirmer, Tiago Novello, Vinícius da Silva, Guilherme Schardong, Daniel Perazzo, Hélio Lopes, Nuno Goncalves, and Luiz Velho. Geometric implicit neural representations for signed distance functions. *Computers & Graphics*, 125: 104085, 2024. 1
- [39] Kexuan Shi, Xingyu Zhou, and Shuhang Gu. Improved implicit neural representation with fourier reparameterized training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 25985–25994, 2024. 1, 2
- [40] Vincent Sitzmann, Michael Zollhöfer, and Gordon Wetzstein. Scene representation networks: Continuous 3d-structure-aware neural scene representations. *Advances in Neural Information Processing Systems*, 32, 2019. 1, 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, 4, 6, 7
- [42] Towaki Takikawa, Joey Litalien, Kangxue Yin, Karsten Kreis, Charles Loop, Derek Nowrouzezahrai, Alec Jacobson, Morgan McGuire, and Sanja Fidler. Neural geometric level of detail: Real-time rendering with implicit 3d shapes. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11358–11367, 2021. 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, 6
- [44] Thomas Walker, Octave Mariotti, Amir Vaxman, and Hakan Bilen. Spatially-adaptive hash encodings for neural surface reconstruction. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 2963–2972. IEEE, 2025. 2
- [45] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004. 5
- [46] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces. *Advances in Neural Information Processing Systems*, 34:4805–4815, 2021. 7
- [47] Xinjie Zhang, Xingtong Ge, Tongda Xu, Dailan He, Yan Wang, Hongwei Qin, Guo Lu, Jing Geng, and Jun Zhang. Gaussianimage: 1000 fps image representation and compression by 2d gaussian splatting. In *European Conference on Computer Vision*, 2024. 2, 6
- [48] Yunxiang Zhang, Alexandr Kuznetsov, Akshay Jindal, Kenneth Chen, Anton Sochenov, Anton Kaplanyan, and Qi Sun. Image-gs: Content-adaptive image representation via 2d gaussians. *arXiv preprint arXiv:2407.01866*, 2024. 2

# SASNet: Spatially-Adaptive Sinusoidal Networks for INRs

Haoan Feng<sup>1</sup>, Diana Aldana<sup>2</sup>, Tiago Novello<sup>2</sup>, Leila De Floriani<sup>1</sup>  
<sup>1</sup>University of Maryland, College Park, <sup>2</sup> IMPA

hfengac@umd.edu, diana.aldana@impa.br,  
 tiago.novello@impa.br, deflo@umd.edu

## Contents

<b>1. Proof of Theorem 1</b>	<b>1</b>
<b>2. Contribution map</b>	<b>2</b>
<b>3. Implementation details</b>	<b>2</b>
3.1. Frequency embedding layer initialization . . .	2
3.2. Model configurations . . . . .	2
<b>4. Computation performance and overhead</b>	<b>3</b>
<b>5. More experimental results</b>	<b>3</b>
5.1. Influence of the hyperparameter $\omega_0$ . . . . .	3
5.2. Visualizations for the SDF . . . . .	3
5.3. Additional experiments on CT medical images	4
5.4. Additional experiments on FFHQ dataset . . .	4
5.5. Additional ablation study results . . . . .	4
<b>6. Extreme cases</b>	<b>7</b>

## 1. Proof of Theorem 1

In the main paper, we claim that the  $j$ -th neuron of the  $i+1$ -th layer  $h_j^{i+1}$  admits the following expansion:

$$h_j^{i+1}(\mathbf{x}) = \sum_{\mathbf{k} \in \mathbb{Z}} \alpha \sin(\langle \mathbf{k}, \mathbf{y} \rangle + b_j^{i+1}) \quad (1)$$

where  $\mathbf{y} = \mathbf{W}^i \mathbf{h}^i(\mathbf{x}) + \mathbf{b}^i$ , and  $\alpha = \prod_l J_{k_l}(W_{j_l}^{i+1})$  are the products of Bessel functions of the first kind at the  $j$ -th row of the hidden matrix  $\mathbf{W}^{i+1}$ .

Considering  $h_j^{i+1} = \sin(\sum_{k=1}^m W_{jk}^{i+1} \sin(y_k) + b_j^{i+1})$ , we now prove Eq. (1) by induction on  $m$ . For  $m = 1$ ,

$$\begin{aligned} h_j^{i+1} &= \sin(W_{j1}^{i+1} \sin(y_1) + b_j^{i+1}) \\ &= \sin(W_{j1}^{i+1} \sin(y_1)) \cos(b_j^{i+1}) + \\ &\quad \cos(W_{j1}^{i+1} \sin(y_1)) \sin(b_j^{i+1}) \end{aligned}$$

$$\begin{aligned} &= \sum_{k \in \mathbb{Z} \text{ odd}} J_k(W_{j1}^{i+1}) \sin(ky_1) \cos(b_j^{i+1}) + \\ &\quad \sum_{k \in \mathbb{Z} \text{ even}} J_k(W_{j1}^{i+1}) \cos(ky_1) \sin(b_j^{i+1}) \\ &= \sum_{k \in \mathbb{Z} \text{ odd}} J_k(W_{j1}^{i+1}) \sin(ky_1 + b_j^{i+1}) + \\ &\quad \sum_{k \in \mathbb{Z} \text{ even}} J_k(W_{j1}^{i+1}) \sin(ky_1 + b_j^{i+1}) \\ &= \sum_{k \in \mathbb{Z}} \alpha \sin(ky_1 + b_j^{i+1}). \end{aligned}$$

The first equality follows from the trigonometric identity  $\sin(a+b) = \sin(a)\cos(b) + \cos(a)\sin(b)$ , the second follows from standard identities for Bessel functions of the first kind (see [2, p. 361]). For the third, we use the properties  $\sin(a)\cos(b) = \frac{\sin(a+b) + \sin(a-b)}{2}$  and  $J_{-k}(a) = (-1)^k J_k(a)$  to rewrite the summations.

Now assume that Eq. (1) holds for  $m-1$ . Then,

$$\begin{aligned} h_j^{i+1} &= \sin\left(\sum_{k=1}^m W_{jk}^{i+1} \sin(y_k) + b_j^{i+1}\right) \\ &= \sin\left(\sum_{k=1}^{m-1} W_{jk}^{i+1} \sin(y_k) + b_j^{i+1}\right) \cos\left(W_{jm}^{i+1} \sin(y_m)\right) + \\ &\quad \cos\left(\sum_{k=1}^{m-1} W_{jk}^{i+1} \sin(y_k) + b_j^{i+1}\right) \sin\left(W_{jm}^{i+1} \sin(y_m)\right) \\ &= \sum_{\substack{\mathbf{k} \in \mathbb{Z}^{m-1} \\ l \in \mathbb{Z} \text{ even}}} \alpha_{\mathbf{k}} J_l(W_{jm}^{i+1}) \sin\left(\sum_{s=1}^{m-1} k_s y_s + b_j^{i+1}\right) \cos(l y_m) + \\ &\quad \sum_{\substack{\mathbf{k} \in \mathbb{Z}^{m-1} \\ l \in \mathbb{Z} \text{ odd}}} \alpha_{\mathbf{k}} J_l(W_{jm}^{i+1}) \cos\left(\sum_{s=1}^{m-1} k_s y_s + b_j^{i+1}\right) \sin(l y_m) \\ &= \sum_{\mathbf{k} \in \mathbb{Z}^m} \alpha \sin(\langle \mathbf{k}, \mathbf{y} \rangle + b_j^{i+1}) \end{aligned}$$

Again, the second equality follows from the angle-sum identity for sine, and the third uses the induction hypothesis. This completes the proof.

## 2. Contribution map

Analyzing the contribution of each neuron is an effective approach to understand a network’s fitting capability and parameter redundancy. According to Padmanabhan et al. [10], for a neuron  $h_j^i$ , its contribution to each pixel coordinate  $\mathbf{x}$  is defined by a mapping from its weights  $\mathbf{W}_j^i$  to the neuron output scalar  $h_j^i(\mathbf{x})$ , i.e.,  $\sum_l h_j^i(\mathbf{x}) \cdot \mathbf{W}_{jl}^{i+1}$ . Thus, the contribution map of  $h_j^i$  is given by accumulating the pixel-wise contribution across the entire image. These maps illustrate the spatial representation focus of neurons<sup>1</sup>.

In the main paper, Fig. 3 presents contribution maps of nine neurons from SIREN [12], WIRE [11], and SASNet, representing different INR categories: *Global*, *Local*, and *Hybrid* (see more examples in the Related work). The maps reveal that neurons in *global* INRs tend to be redundant and “*space-agnostic*”, whereas those in *local* INRs exhibit sparser and more spatially localized contributions. Additionally, the global nature of INRs gives rise to model overfitting - neurons fitting high-frequency regions introduce noise into low-frequency areas. In such regions, e.g., image backgrounds, other neurons may need to counteract the effect of high-frequency noise by learning complementary signals to suppress these artifacts. This inefficiency in INRs is one of the main issues addressed in this paper.

## 3. Implementation details

### 3.1. Frequency embedding layer initialization

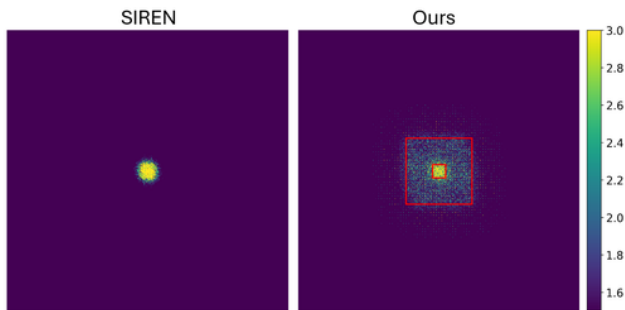


Figure 1. Comparing the frequency spectra of SIREN and our SASNet before training. Two red rectangles indicate band limits of low- and high-frequency embedding neurons.

As shown in Fig. 1, given a predefined band limit  $\mathcal{B}$  and a low-frequency range  $\mathcal{L}$ , we split the frequency domain for selecting  $\omega$  into two concentric rectangular regions giving rise to the low- and high-frequency neurons. Considering image-fitting as an example, low-frequency neuron multipliers  $\mathbf{k}_{low} \in \mathbb{Z}^2$  are randomly sampled from  $[-\mathcal{L}, \mathcal{L}]^2$ . To ensure better coverage, high-frequency multipliers  $\mathbf{k}_{high}$  are sampled on an integer grid within  $[-\mathcal{B}, \mathcal{B}]^2$ , excluding the low-frequency region  $[-\mathcal{L}, \mathcal{L}]^2$ . To reduce redundancy, we constrain sampling to positive integers along the  $x$ -axis, leveraging a symmetry of the frequency domain, where  $(k_x, -k_y)$  is equivalent to  $(-k_x, k_y)$ . By adjusting  $\mathcal{B}$  and balancing low-

<sup>1</sup>Publicly available at <https://github.com/namithap10/xinc>

and high-frequency neurons, SASNet explicitly controls the output frequencies, improving high-frequency detail reconstruction. Choosing an appropriate  $\mathcal{B}$  mitigates overfitting and enhances the reconstruction of high-frequency details.

### 3.2. Model configurations

For the image fitting task, we follow each method’s open-source implementation unless stated otherwise. For the sinusoidal baselines, i.e., FINER, SIREN, and SASNet, we increase  $\omega_0$  from 30 to 43 because the images used here are of higher resolution than those in prior work, and this value gives a good trade-off between PSNR and reconstruction noisiness. All models are optimized with Adam using a fixed learning rate of  $10^{-4}$ , except as noted below.

Tab. 1 summarizes the task-specific SASNet hyperparameters used in our image fitting, volumetric fitting, and SDF reconstruction experiments. In all cases, the low-frequency mask is fixed to one, the remaining masks are initialized around 0.5, and the hash-grid and decoder use a learning rate of  $5 \times 10^{-4}$ .

Table 1. SASNet model configurations used in the supplementary experiments. For the SDF task, the band limit is 30 for Armadillo and Dragon, and 40 for Lucy and Thai Statue.

Setting	Image	Volume	SDF
Hardware	RTX A6000	RTX A6000	RTX A6000
Batch size	262144	1780000	10000
Input dim	2	3	3
$\omega_0$	43	30	30
# trainable params	128865	138181	130917
FEmb layer size	256	256	256
Hidden layer size	232	192	224
Band limit	60	30	30 (40)
Low-freq range	12	6	6
Low-freq percentage	0.5	0.5	0.5
# high-freq bands	4	4	4
# neuron groups	8	8	8
Hash-grid			
feature dim	2	2	2
feature std	0.01	0.01	0.01
feature bias	0	0	0
codebook bitwidth	10	11	10
num levels	10	12	10
Mask decoder	ReLU-MLP with one hidden layer of size 48		

Additional implementation details for the compared methods are listed below:

- FINER [8]: the bias scale is set to 0.
- FFN [13] and SAPE [5]: we use the implementation in SAPE [5]<sup>2</sup>. For each training step, instead of sampling coordinates, we feed all coordinates for supervision. The number of frequencies is set to 256 according to its performance on the Kodak dataset.
- WIRE [11]: we use the implementation that each network layer uses a complex Gabor function (except the output layer). The learning rate is fixed to  $5e-4$ . For the image-fitting task, the hyperparameter  $\omega$  is set to 20, and  $\sigma$  is set to 30, as suggested by its implementation. For the toy example, the

<sup>2</sup>Publicly available at <https://github.com/amirhertz/sape>.

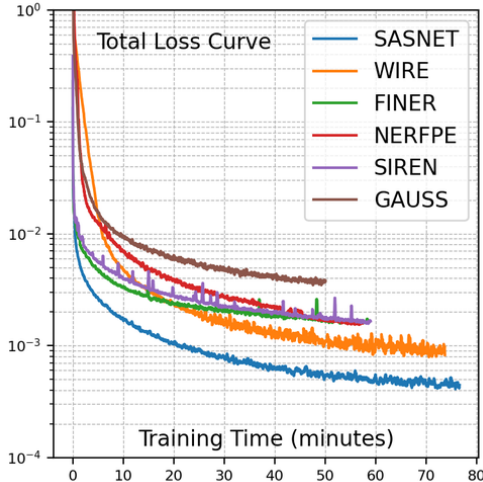


Figure 2. Training loss plotted against wall-clock time on the 2D image fitting task. SASNet reaches the same loss level substantially faster, highlighting its improved time-to-accuracy despite a modest per-step overhead.

hyperparameter  $\omega$  is set to 4, and the  $\sigma$  is set to 10 according to a simple hyperparameter search.

- GaussianImage [14]: without applying quantization, we fitted each image with 16640 Gaussians for 50K training epochs.
- NeuRBF [4]: we tune the hyperparameters and set  $\log_2(\text{hashmap\_size})$  to 14 and  $n_{\text{levels}}$  to 10.
- SASNet: the full task-specific config. is summarized in Tab. 1.

For volumetric data and SDF fitting, we use the implementation of FINER [8]<sup>3</sup> and reuse its training and evaluation scripts on the four provided training point clouds. The corresponding SASNet hyperparameters for these two tasks are listed in Tab. 1.

## 4. Computation performance and overhead

Table 2. Efficiency comparison on the 2D image fitting task. We report average step time (AST) and peak GPU memory usage (PMem) for models with comparable numbers of trainable parameters. Lower values indicate better efficiency.

Model	AST↓	PMem↓
SIREN	<b>29,224</b>	<b>2075.50</b>
MSIREN	95,438	6077.07
INCODE	59,907	3642.59
SASNet	43,107	3031.60

We report **average step time** (AST,*us*) and **peak GPU mem usage** (PMem, MiB) from PyTorch Profiler (averaged over 5 runs) with similar #trainable params. For the image fitting task (Tab. 2), we use batch size 262,144; INCODE[7] freezes the `feature_extractor` and MSIREN[9] assumes precomputed

<sup>3</sup>Publicly available at: <https://github.com/liuzhen0212/FINER>

Table 3. Quantitative evaluation on the Kodak [1] dataset.

Methods	PSNR ↑	SSIM ↑	PSNR <sub>edge</sub> ↑	Noisiness ↓
SIREN [12]	36.60	0.951	33.28	4.079
FFN [13]	33.00	0.868	29.87	10.353
SAPE [5]	33.37	0.872	31.65	5.373
WIRE [11]	29.34	0.804	25.23	15.375
FINER [8]	37.72	0.973	34.67	5.174
GI [14]	37.58	0.971	32.59	-
NeuRBF [4]	37.51	0.971	34.78	6.258
SASNet	40.43	0.986	38.12	3.993

latent codes and uses  $32^2$  tiles with stride 16. For the 3D shape fitting task, we further profile SASNet and SIREN. SASNet increases AST from 17,557 to 22,175 and PMem from 98.0 to 118.2, indicating only a modest computational overhead.

This overhead mainly comes from the multi-scale hash-grid queries used to generate spatial masks. In practice, these gathers are often latency-limited because their memory accesses exhibit poor spatial locality and are therefore not well coalesced, making runtime sensitive to cache misses and memory-level parallelism. However, per-step efficiency alone does not fully reflect optimization behavior. When we redraw the training curve in the main figure as Fig. 2 with wall-clock time, SASNet reaches the same training loss in approximately 10 minutes, whereas SIREN requires 58 minutes. Overall, SASNet offers a favorable trade-off between model complexity and training efficiency: it incurs a moderate increase in runtime and memory usage, but converges substantially faster in practice. A more detailed runtime breakdown and optimization study is left for future work. In addition, the signal-conditioned modulation strategy used in INCODE suggests a possible direction for reducing this overhead further, namely predicting spatial masks with an INCODE-style module instead of generating them explicitly.

## 5. More experimental results

### 5.1. Influence of the hyperparameter $\omega_0$

As shown in Fig. 1 of the main paper, SIREN [12] is highly sensitive to the choice of the hyperparameter  $\omega_0$ , making its selection non-trivial. Smaller values lead to blurry reconstructions without introducing high-frequency noise, whereas higher values reduce spectral bias and produce sharper outputs, but resulting in increased noisiness in smooth regions.

In contrast, SASNet leverages spatially-adaptive masks to separate frequency band expressions, preventing high-frequency neuron outputs from propagating into smooth regions. As a result, the choice of  $\omega_0$  has minimal impact on the final reconstruction accuracy, as illustrated in Fig. 3, where PSNR remains stable across different  $\omega_0$  values.

### 5.2. Visualizations for the SDF

Fig. 4 first visualizes the SDF error map for the Armadillo example. We compare a model equipped with the fixed frequency embedding but without spatial masks against the full SASNet model. Without spatial localization, the reconstruction error is

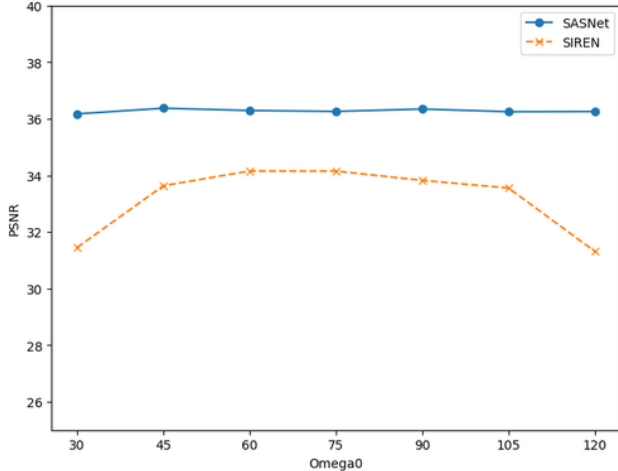


Figure 3. Sensitivity comparison between SIREN and our SASNet to the change of hyperparameter  $\omega_0$ .

distributed more broadly over the surface, especially around geometrically detailed regions, which indicates that globally available high-frequency components are not used selectively. In contrast, SASNet suppresses these artifacts and produces a substantially cleaner error distribution, showing that the learned masks help restrict high-frequency responses to the regions where they are most needed.

To further explain this behavior by the Dragon example in the main paper, Fig. 5 visualizes the learned spatial mask corresponding to the *highest-frequency band* of the frequency embedding layer. The figure shows the target SDF isosurface for Dragon overlaid on the volumetric mask produced by the multi-scale hash-grid decoder. The mask values are rendered as a volumetric field using the same color encoding adopted for the 2D image masks. High mask responses concentrate near sharp geometric ridges, thin structures, and high-curvature surface regions, while remaining weak in smooth or empty space. This qualitative result confirms that SASNet allocates high-frequency capacity selectively, which helps explain the reduced error observed in Fig. 4.

### 5.3. Additional experiments on CT medical images

In addition to the two RGB image datasets used in the main paper, we further compare SASNet with FINER [8] and SIREN [12] on a publicly available CT medical image dataset<sup>4</sup>, which contains 100 CT scan images from the TCIA cancer imaging archive [3]. Each scan has a resolution of  $512^2$  and consists of a single-channel grayscale image, normalized to  $[0, 1]$ . We conduct the image-fitting experiments using the same configuration as in the main paper. These results further demonstrate that SASNet generalizes well across modalities and resolutions.

Tab. 4 presents the quantitative evaluations, showing that SASNet achieves higher fitting accuracy, particularly in preserving edge details. Notably, for single-channel images, the model configuration used in previous experiments is sufficiently large for all

<sup>4</sup><https://www.kaggle.com/datasets/kmader/siim-medical-images>

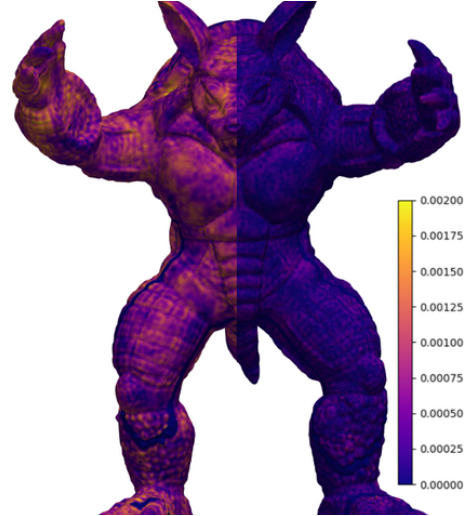


Figure 4. SDF error map for the Armadillo example. Left: a model with fixed frequency embedding but without spatial masks. Right: SASNet. The error map shows that spatially-adaptive masking suppresses widespread surface artifacts and localizes the remaining error to a smaller set of challenging regions.

Table 4. Quantitative evaluation on a medical image dataset. For each metric,  $\uparrow/\downarrow$  indicates the numerical value is *higher/lower* the method performs better.

Methods	PSNR $\uparrow$	SSIM $\uparrow$	PSNR <sub>edge</sub> $\uparrow$	Noisiness $\downarrow$
SIREN [12]	47.07	0.98	42.94	3.482
FINER [8]	48.49	0.99	45.56	3.488
SASNet (Ours)	50.21	0.99	49.58	3.681

methods to fit the data without overfitting. Consequently, the noisiness across all methods remains low. This is further observable in Fig. 6, where both the model outputs and their error maps are visualized.

### 5.4. Additional experiments on FFHQ dataset

The Flickr-Faces-HQ (FFHQ) dataset [6] contains a large collection of high-quality human face images at a resolution of  $1024^2$ . For this experiment, 100 images are randomly selected from the training image set. Similar to the medical image experiment, the same network configuration is applied to SIREN [12], FINER [8], and our SASNet.

Tab. 5 presents the quantitative evaluations, while Fig. 7 provides a visual comparison among these methods. SASNet achieves higher fitting accuracy (as observed in the error maps), and this PSNR improvement does not result in an increased noisiness level, in contrast to FINER.

### 5.5. Additional ablation study results

In the main paper, we conduct an ablation study to evaluate the contribution of each module, namely the frequency embedding layer and spatially-adaptive masks at different layers. Fig. 8

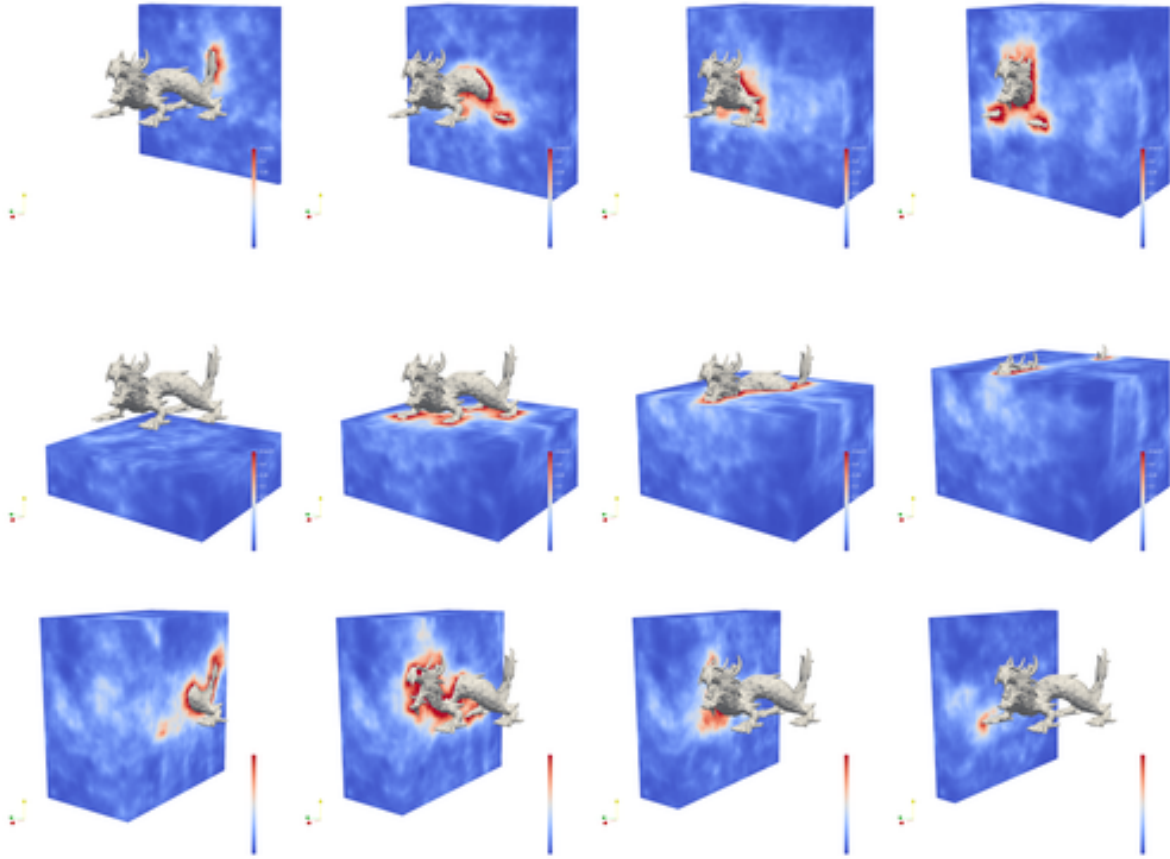


Figure 5. Learned 3D spatial mask for SASNet’s highest-frequency band on the Dragon SDF. Across multiple viewpoints, high responses are concentrated near the object surface and around geometrically complex regions.

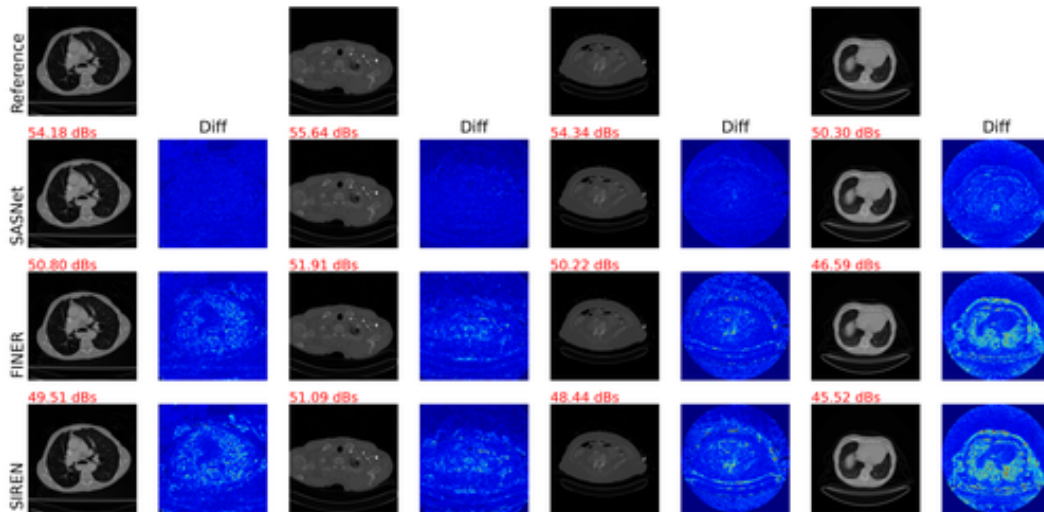


Figure 6. Medical CT scans’ fitting results by SASNet, FINER, and SIREN. For each of the four examples, PSNR values are annotated on the upper left corner of the model outputs. Besides each model’s output image, the *Diff* column shows the error map with a *jet* color encoding (up to 2% absolute difference).

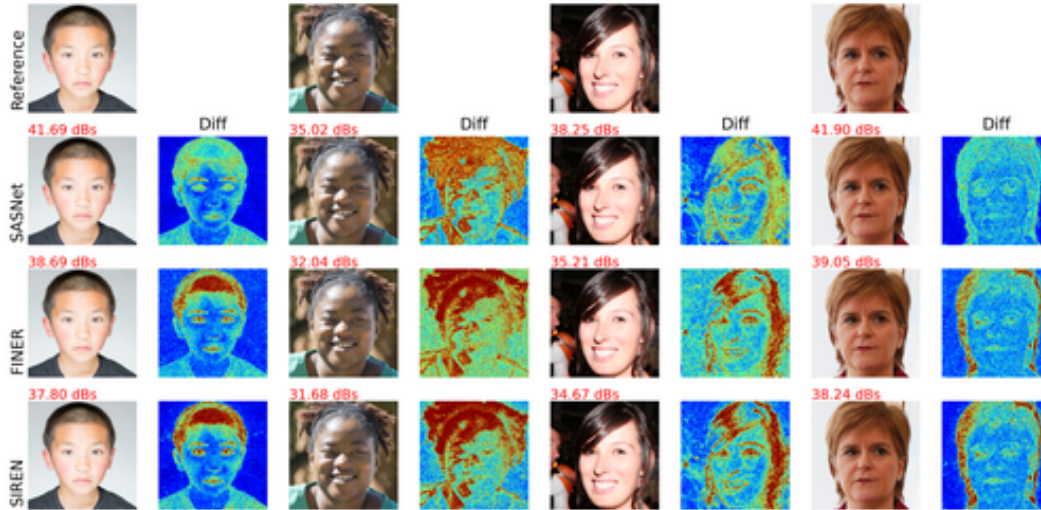


Figure 7. Fitting results of FFHQ human faces by SASNet, FINER, and SIREN. For each of the four examples, PSNR values are annotated on the upper left corner of model outputs. Besides each model’s output image, the *Diff* column shows the error map with a *jet* color encoding (up to 2% absolute difference).

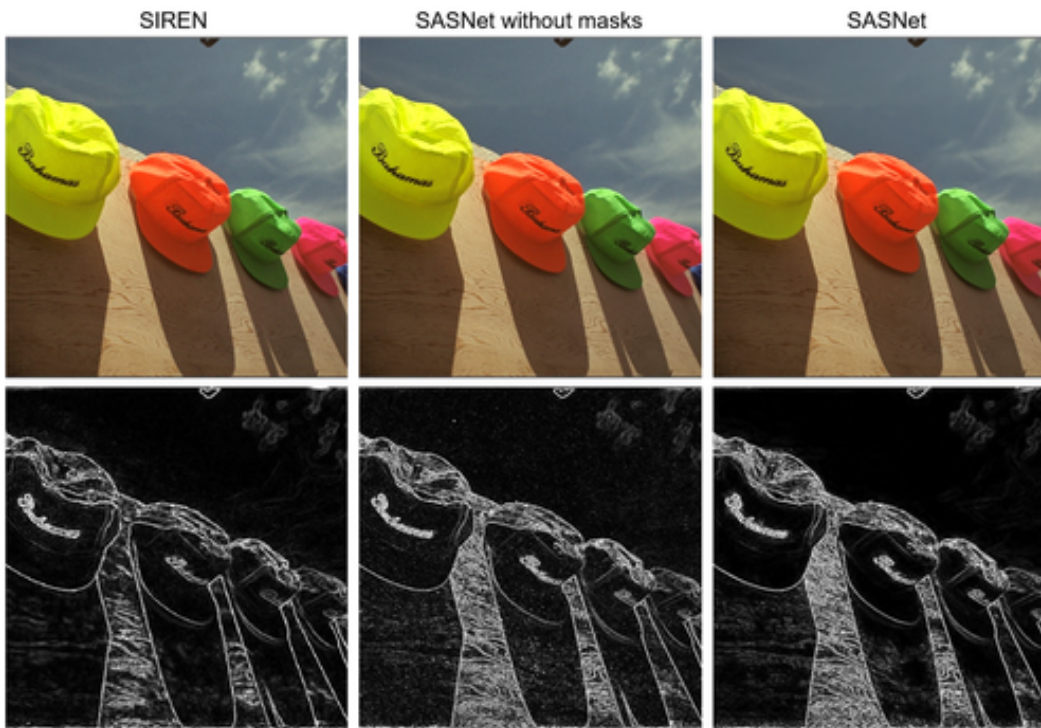


Figure 8. Visual comparison between SIREN, SASNet without masks, and full SASNet. First row shows the image reconstructed and the second row shows the corresponding gradient intensity obtained by network automatic differentiation (*autograd* in PyTorch).

presents a visual comparison among SIREN, SASNet with only frequency embedding (without masks), and the full SASNet. The results demonstrate that while the frequency embedding layer enhances fitting accuracy, particularly for sharper reconstructions, it also introduces additional noise in the image background. By ap-

plying spatially-adaptive masks, the quality of background reconstruction significantly improves, as indicated by the reduction in image gradient intensity.

Fig. 9 visualizes the jointly optimized spatial masks of SASNet and SAPE for the same input image as in Fig. 8. The learned

Table 5. Quantitative evaluation on the FFHQ dataset. For each metric,  $\uparrow/\downarrow$  indicates the numerical value is *higher/lower* the method performs better.

Methods	PSNR $\uparrow$	SSIM $\uparrow$	PSNR <sub>edge</sub> $\uparrow$	Noisiness $\downarrow$
SIREN [12]	37.58	0.95	34.20	4.178
FINER [8]	38.16	0.95	35.09	4.975
SASNet (Ours)	39.69	0.95	37.64	4.278

masks for different frequency bands effectively align with the spatial complexity of the image, enabling the model to focus its fitting capacity on complex, high-frequency regions while preventing resource waste and overfitting in simpler regions.

Additionally, a hidden mask is selected from each hidden layer, and contribution maps of the corresponding neuron groups are visualized. These results indicate that spatial masks allow different regions to encode distinct frequency bands, leading to lower network redundancy in SASNet compared to SIREN. In smooth and easy-to-fit regions, only a subset of neurons contributes to reconstruction, whereas in complex, texture-rich regions, most neurons participate in signal reconstruction, demonstrating the effectiveness of spatially-adaptive frequency control.

## 6. Extreme cases

Under extreme cases, as shown in Fig. 10, INRs’ performance can be limited for certain classes of image content. Both images primarily consist of high-frequency repeated patterns that span almost the entire domain, and with minimal pixel variation within each “color strip”. This structure is particularly well-suited for GaussianImage, as each Gaussian kernel can efficiently cover a larger portion of the domain with minimal computational effort. Conversely, INR methods suffer from spectral bias and an insufficient number of parameters to effectively capture such a broad frequency spectrum, leading to reduced fitting accuracy in these scenarios. These examples illustrate the inherent limitations of INRs under extreme global-frequency conditions.

## References

- [1] Kodak lossless true color image suite. <http://r0k.us/graphics/kodak/>, 1999. Accessed: 2024-09-01. 3
- [2] Milton Abramowitz and Irene A Stegun. *Handbook of mathematical functions with formulas, graphs, and mathematical tables*, volume 55. US Government printing office, 1948. 1
- [3] Brad Albertina, Mark Watson, Chandra Holback, Rose Jarosz, Shanah Kirk, Yueh Lee, Kimberly Rieger-Christ, and John Lemmerman. The cancer genome atlas lung adenocarcinoma collection (tcga-luad). 2016. 4
- [4] Zhang Chen, Zhong Li, Liangchen Song, Lele Chen, Jingyi Yu, Junsong Yuan, and Yi Xu. Neurbf: A neural fields representation with adaptive radial basis functions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4182–4194, 2023. 3
- [5] Amir Hertz, Or Perel, Raja Giryes, Olga Sorkine-Hornung, and Daniel Cohen-Or. Sape: Spatially-adaptive progressive encoding for neural optimization. *Advances in Neural Information Processing Systems*, 34:8820–8832, 2021. 2, 3
- [6] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019. 4
- [7] Amirhossein Kazerouni, Reza Azad, Alireza Hosseini, Dorit Merhof, and Ulas Bagci. Incode: Implicit neural conditioning with prior knowledge embeddings. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1298–1307, 2024. 3
- [8] Zhen Liu, Hao Zhu, Qi Zhang, Jingde Fu, Weibing Deng, Zhan Ma, Yanwen Guo, and Xun Cao. Finer: Flexible spectral-bias tuning in implicit neural representation by variable-periodic activation functions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2713–2722, 2024. 2, 3, 4, 7
- [9] 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. 3
- [10] Namitha Padmanabhan, Matthew Gwilliam, Pulkit Kumar, Shishira R Maiya, Max Ehrlich, and Abhinav Shrivastava. Explaining the implicit neural canvas: Connecting pixels to neurons by tracing their contributions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10957–10967, 2024. 2
- [11] 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, 3
- [12] 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. 2, 3, 4, 7
- [13] 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, 3
- [14] Xinjie Zhang, Xingtong Ge, Tongda Xu, Dailan He, Yan Wang, Hongwei Qin, Guo Lu, Jing Geng, and Jun Zhang. Gaussianimage: 1000 fps image representation and compression by 2d gaussian splatting. In *European Conference on Computer Vision*, 2024. 3

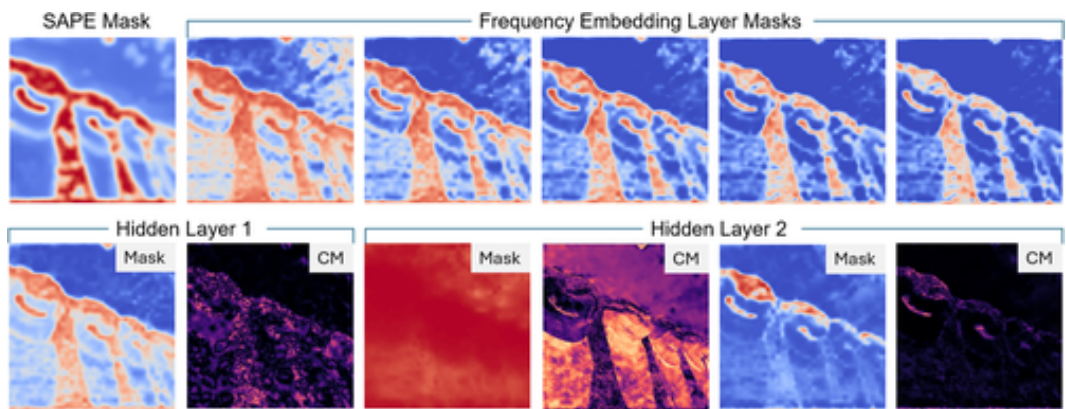


Figure 9. SAPE mask and our masks of the image-fitting task. CM: contribution map. The first hidden layer mask shows that this group has a global contribution, as also shown in the CM. However, some details on edges and texture regions can be missed, and therefore, the other group focuses only on these small regions.

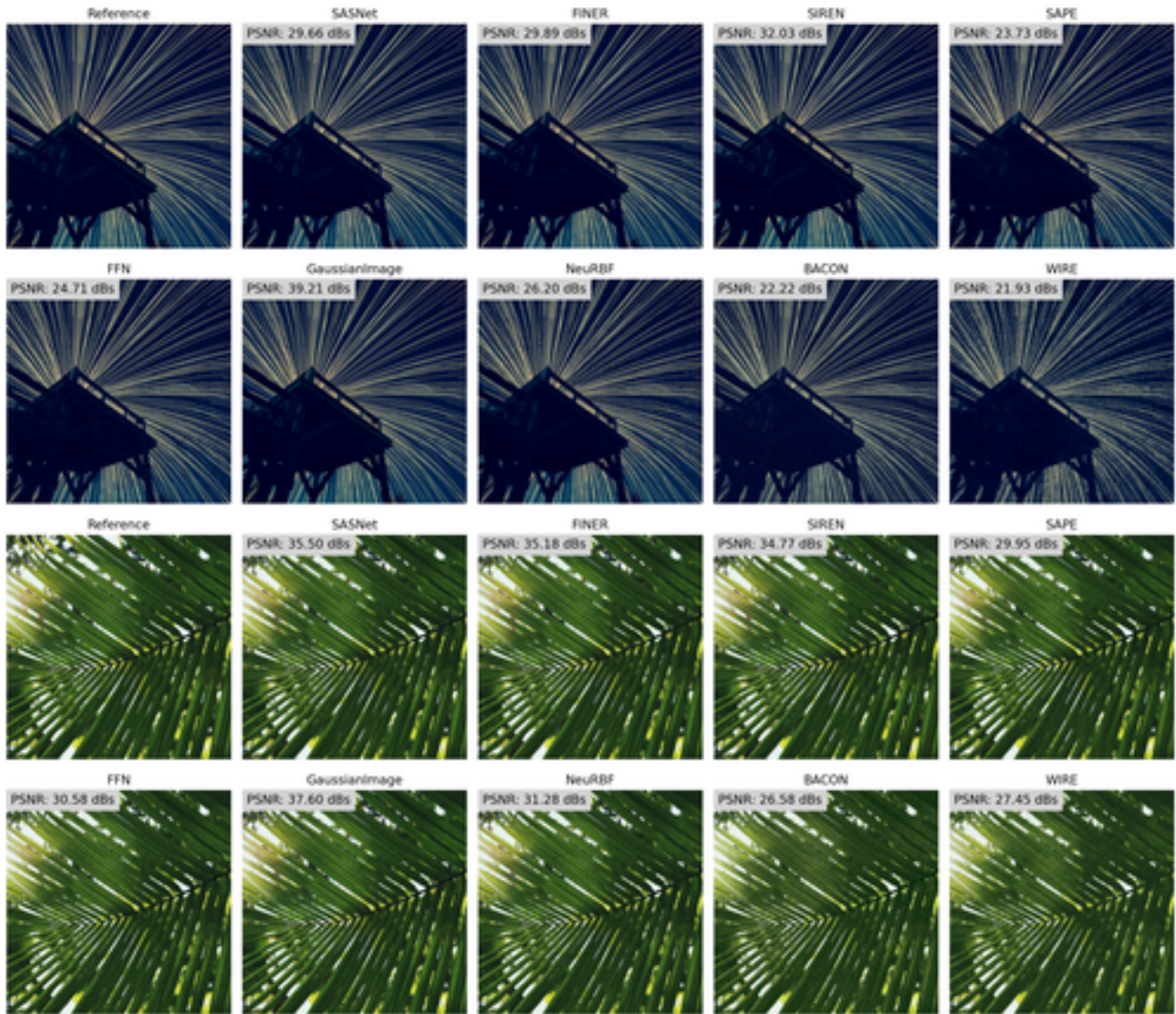


Figure 10. Example images from the DIV2K dataset showing when performances of INRs are limited. Both images are of complex signal components and higher frequencies than others, and contain hard edges covering almost the whole domain.