SPAN-INR: SPATIAL ASSIGNMENT NETWORKS FOR DEEP SINUSOIDAL IMPLICIT REPRESENTATIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Real signals are detailed in some places and smooth in others, so a sinusoidal implicit neural representation (INR) has to produce different harmonics at different locations. Existing sinusoidal INRs, however, spend their parameters on a separate matrix in every layer, which decides which channels can combine (*reach*), and do little to control what is produced where (*assignment*): the scale and offset of the argument entering each sine, which by the Jacobi–Anger expansion decide how high the harmonic orders go and the balance between odd and even harmonics, are the same at every location or are adjusted only by coarse masks on groups of features. In this paper, we first show that in our setting reach is cheap and assignment is what has to be learned. Analytically, which harmonic interactions are available depends on the layer matrix only through its connectivity, and the scale of each of its rows can be taken over by a gain; empirically, a single dense mixing matrix shared by all stages and frozen at its random initialization loses only 0.14 dB against a fully learned one, whereas restricting it to its diagonal loses 5.81 dB, and a model with an independent matrix per stage loses 3.48 dB at the same budget. Based on this observation, we propose a new sinusoidal INR, called Spatial Assignment Network (SPAN-INR), which spends its parameters on assignment rather than on reach: most of them go to coordinate controllers, one per stage, that predict a gain and a shift for every channel at every location, while the single shared mixing matrix that provides the reach is small by comparison. Since a new stage needs only a new controller and no new matrix, the same budget buys more stages, and every stage is one more place to assign harmonics locally. Experiments show that SPAN-INR achieves 38.90 dB PSNR on image fitting with 126K trainable parameters and the best results among the compared methods on super-resolution, denoising, 3D occupancy and novel-view synthesis. Analyses of the fitted models further confirm that the gain tracks the local spectral centroid in the later stages, the shift tracks the odd/even balance, and the empirical neural tangent kernel is 9.0 times narrower at edges than in smooth regions, against 4.8 times for SASNet.

1 INTRODUCTION

Implicit neural representations (INRs) represent a signal with a neural network that maps coordinates to signal values. Sinusoidal INRs use multilayer perceptrons with sine activations to capture fine detail (Sitzmann et al., 2020), and repeated sinusoidal composition generates harmonic combinations of the incoming frequencies, so a compact network can represent a wide range of frequencies (Yüce et al., 2022; Novello et al., 2025). Many signals, however, place detailed textures and sharp edges next to smooth regions. Accurate reconstruction therefore requires representing different levels of detail across locations.

Existing INR methods introduce explicit controls over the network's features through latent-conditioned modulation (Mehta et al., 2021), coordinate-dependent amplitude masks (Feng et al., 2025), frequency reparameterization (Shukla et al., 2026), or band-localized activations (Ko et al., 2026). These methods differ in the form of control they provide and where it enters the network. Finer spatial control can help adjust feature contributions at each location, preserving detailed textures while maintaining smooth regions. Figure 1 shows that SPAN-INR preserves fine texture and follows the ground-truth luminance profile more faithfully (a), achieves lower log-power error across

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

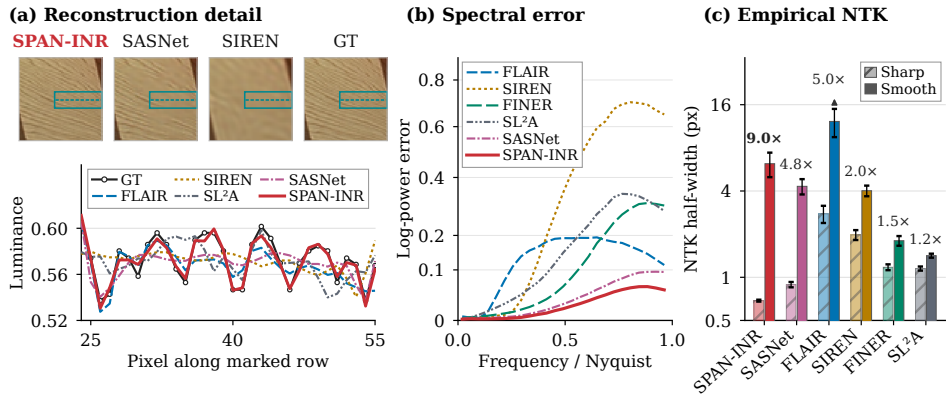


Figure 1: Fine-detail reconstruction and spatial adaptation. (a) SPAN-INR recovers fine details in the crops and follows the ground-truth luminance profile more faithfully than SASNet, FLAIR, SIREN, and SL^{2}A. SASNet's grouped masks scale several features together; with incoming features fixed, they cannot adjust their relative contributions. SPAN-INR provides gain and shift for each mixed feature channel. (b) SPAN-INR achieves the lowest spectral error across all measured frequency bands among the compared methods . (c) Its empirical NTK exhibits the narrowest coupling at edges and broader coupling in smooth regions, indicating spatially adaptive fitting; the number above each pair is the smooth-to-sharp half-width ratio. Panels (b,c) summarize the full-resolution 24 Kodak images.

frequency bands (b), and exhibits a narrower empirical neural tangent kernel (NTK) (Jacot et al., 2018) in sharp regions and a broader one in smooth regions (c).

Building on the amplitude–phase expansion of Novello et al. (2025), we distinguish two aspects of sinusoidal composition: **channel reach** (Reach) and **spatial spectral assignment** (Assignment). Applying sine to a weighted sum of sine features can generate harmonic combinations across channels. Reach describes which incoming channels the mixing matrix connects for these interactions. Assignment describes how harmonic contributions vary across coordinates. The mixing weights influence these contributions, while coordinate-dependent gain and shift provide explicit spatial control: gain rescales the mixed features and changes their harmonic coefficients, while shift changes the phase offset and the balance between even and odd harmonic orders (Proposition 1). These controls allow the same channel connections to support different harmonic contributions at different locations.

We introduce the **Spatial Assignment Network (SPAN-INR)**, which combines shared channel mixing with stage-specific spatial control (Fig. 2). Reach is provided by a single learned dense mixing matrix W, shared by all stages. Assignment is performed by coordinate controllers: each stage has its own controller, a small multilayer perceptron (MLP) that reads the input coordinate and predicts a gain field and a shift field, which set the scale and offset of every mixed feature at that location. Successive stages repeat this computation, and a linear output layer maps the final features to the signal. Through gain and shift, each controller adjusts how the mixed features enter subsequent sinusoidal composition, allowing the shared connections to support different harmonic contributions across the signal. Sharing keeps the number of mixing parameters independent of depth, leaving more of a fixed parameter budget for coordinate controllers and additional stages. A local spectral approximation relates the resulting harmonic coefficients and phase variation to local frequency content (Appendix A).

Our experiments study both Assignment and Reach. For Assignment, removing the coordinate controllers, gain, or shift reduces reconstruction accuracy. For Reach, freezing the shared dense W at random initialization reduces image-fitting PSNR by only 0.14 dB when the controllers and output layer are trained, whereas diagonal mixing reduces it by 5.81 dB. These results support learning spatial control while reusing fixed cross-channel connections. Analyses of fitted models further show that a dense W keeps the internal spectra closer to the target than a diagonal W, while the empirical NTK is narrower at edges than in smooth regions, showing spatially adaptive fitting. Overall, SPAN-INR achieves strong results across image fitting, super-resolution, denoising, 3D occupancy, and novel-view synthesis.

Contributions. This paper makes three contributions:

(i) **Analysis of Reach and Assignment.** We distinguish channel reach (Reach) from spatial spectral assignment (Assignment), characterize how mixing, gain, and shift shape harmonic contributions, and identify the spatial control retained when the mixing matrix is fixed.

(ii) **Spatial Assignment Network.** We introduce SPAN-INR, which provides Reach through shared dense mixing and allocates most of its parameters to Assignment through stage-specific coordinate controllers for gain and shift. This provides spatial control throughout sinusoidal composition while keeping the number of mixing parameters independent of depth.

(iii) **Empirical evidence for the design.** Experiments across five reconstruction tasks demonstrate SPAN-INR's effectiveness. Ablations show the importance of gain, shift, and cross-channel mixing, with frozen random dense mixing achieving accuracy close to learned mixing. Spectral and empirical NTK analyses further characterize spatial adaptation.

2 Related work

INRs encode signals as continuous functions of coordinates.

Activation-based INRs. SIREN (Sitzmann et al., 2020) uses sine activations. Gauss (Ramasinghe & Lucey, 2022) uses Gaussian responses, WIRE (Saragadam et al., 2023) combines oscillation and localization through complex Gabor activations, and FINER (Liu et al., 2024) varies the sine period with its input magnitude. HOSC (Serrano et al., 2024) applies a scaled sine inside a hyperbolic tangent. SL^{2}A-INR (SL^{2}A) (Rezaeian et al., 2025) learns activation functions through Chebyshev polynomial coefficients, while QuadINR (Zhou et al., 2025) uses piecewise quadratic activations with rich harmonic content. These methods reshape the spectral content available to an INR, but rarely specify how spectral contributions vary across spatial locations.

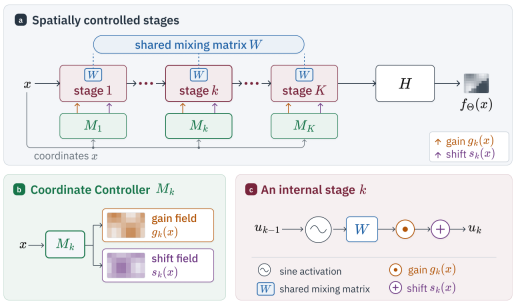


Figure 2: SPAN-INR. (a) Stages share a learned mixing matrix W (dotted links) and have separate coordinate controllers M_k. (b) Each controller MLP predicts gain and shift fields. (c) A stage applies sine, shared mixing, elementwise gain and shift.

Spectral control and sinusoidal composition. BACON (Lindell et al., 2022) prescribes bandwidths through bounded-frequency filters in a multiplicative coordinate network. TUNER (Novello et al., 2025) characterizes sinusoidal composition through integer combinations of input frequencies, with amplitudes determined by the learned weights and phases. It fixes integer input frequencies and constrains hidden weights to control the spectrum. Spatially heterogeneous signals also require frequency contributions to vary across regions: enlarging a global frequency range to capture detail can introduce unwanted oscillations in smooth areas (Novello et al., 2025; Feng et al., 2025).

Spatially adaptive INRs Spatial adaptation uses local encodings or coordinate-dependent modulation. SASNet (Feng et al., 2025) freezes its input-frequency embedding and uses a hash-grid MLP to predict multiplicative masks for groups of sinusoidal neurons. The masks localize feature contributions and, downstream, affect harmonic generation, but supply no separate coordinate-dependent shift. FLAIR (Ko et al., 2026) combines a Gaussian-windowed, band-selective activation with wavelet-energy scores computed from the target signal to guide regional frequency selection. These methods provide spatial adaptation, but offer limited fine-grained control over how spectral contributions are assigned during deep sinusoidal composition. Building on TUNER's amplitude–phase expansion, SPAN-INR uses coordinate-dependent gain and shift to control the harmonic coefficients and operating phase of subsequent sine layers. Assignment therefore receives most of the learned capacity, while channel reach is provided by a shared dense matrix W; empirically, even a frozen random W performs close to a learned one.

3 METHOD: SPATIALLY CONTROLLED SINUSOIDAL COMPOSITION

A standard sinusoidal INR (Sitzmann et al., 2020) uses an MLP whose hidden-layer weights determine both which channels can interact and their harmonic contributions. Since these weights are shared across spatial locations, they provide limited explicit control over where different harmonic contributions should appear. Yet the level of detail in a signal can vary substantially across space, requiring spectral contributions to be assigned differently at different locations. SPAN-INR separates these roles: a shared mixing matrix provides channel reach, while separate, stage-specific coordinate controllers perform spatial spectral assignment by modulating harmonic contributions across locations.

3.1 SPAN-INR ARCHITECTURE

A sinusoidal INR models a signal as $f_\Theta : \Omega \subset \mathbb{R}^d \to \mathbb{R}^p$, where $x \in \Omega$ is a d-dimensional coordinate, p is the output dimension, and Θ denotes the learnable parameters. SPAN-INR consists of K stages with C feature channels. Each stage comprises a shared mixing matrix W and a stage-specific coordinate controller M_k, implemented as an MLP. Given a coordinate $x \in \Omega$, the controller M_k predicts channel-wise gain and shift fields:

$$\big(g_k(x), s_k(x)\big) = M_k(x), \qquad g_k(x), s_k(x) \in \mathbb{R}^C. \tag{1}$$

The initial state $u_0(x) \in \mathbb{R}^C$ contains the input coordinates x, one coordinate per channel. For $k = 1, \ldots, K$, the state $u_k(x)$ is updated as

$$q_k(x) = W \sin\big(\omega_0 u_{k-1}(x)\big), \qquad u_k(x) = g_k(x) \odot q_k(x) + s_k(x). \tag{2}$$

Here, $W \in \mathbb{R}^{C\times C}$ is a learned dense mixing matrix shared across all stages, $q_k(x)$ denotes the mixed sinusoidal features, and $\omega_0 > 0$ is a fixed sine scale. The sine is applied element-wise, and $\odot$ denotes element-wise multiplication.

For $k < K$, the controlled state $u_k(x)$ serves as the input to the sine operation of the next stage. The final state $u_K(x)$ is mapped to the output by an linear layer $H : \mathbb{R}^C \to \mathbb{R}^p$:

$$f_\Theta(x) = H\big(u_K(x)\big). \tag{3}$$

Spatial control allows each mixed feature's scale and offset to differ across locations and stages. With the mixed features held fixed, separate channel gains can change their relative contributions, whereas a common nonzero multiplicative factor preserves their proportions. Shift adds a channel-specific offset, including where a mixed feature is zero. These adjustments provide finer control across feature channels, with spatial variation learned by the controllers.

3.2 SPATIAL SPECTRAL ASSIGNMENT

When a controlled state enters another sine, gain and shift change its input scale and phase offset, affecting the resulting harmonic contributions. **Spatial spectral assignment** describes how these contributions vary across coordinates. To examine this effect, fix a stage $1 \le k < K$ and channel j. Let $\varphi_m(x) = \omega_0 u_{k-1,m}(x)$ be the phase of incoming channel m, collected in $\varphi(x)$. The following sine evaluates

$$\sin\big(\omega_0 u_{k,j}(x)\big) = \sin\left(b(x) + \sum_{m=1}^{C} d_m(x) \sin \varphi_m(x)\right), \tag{4}$$

where $d_m(x) = \omega_0 g_{k,j}(x) W_{jm}$ and $b(x) = \omega_0 s_{k,j}(x)$. The coefficients $d_m(x)$ combine gain with the mixing weights, while $b(x)$ is the phase offset supplied by shift. Their effects on harmonic contributions follow from the amplitude–phase expansion of Novello et al. (2025). Let $\kappa \in \mathbb{Z}^C$ collect one integer harmonic order per incoming phase, and let J_n denote the Bessel function of the first kind of integer order n.

Proposition 1 (Coordinate-dependent harmonic coefficients)**.** *At each coordinate, the controlled sine admits the expansion*

$$\sin\big(\omega_0 u_{k,j}(x)\big) = \sum_{\kappa\in\mathbb{Z}^C} A_\kappa(x) \sin\big(\kappa^\top \varphi(x) + b(x)\big), \qquad A_\kappa(x) = \prod_{m=1}^{C} J_{\kappa_m}\big(d_m(x)\big). \tag{5}$$

To summarize the distribution of harmonic orders, treat the phases as independent variables. The complex exponential of the same sine argument has squared coefficient magnitudes $A_\kappa(x)^2$ summing to one. Their ***harmonic-order moment*** *is*

$$r_{k,j}(x) := \sum_{\kappa \in \mathbb{Z}^C} \|\kappa\|_2^2 A_\kappa(x)^2 = \frac{\omega_0^2}{2} g_{k,j}(x)^2 \|W_{j,:}\|_2^2, \tag{6}$$

where $W_{j,:}$ is row j and $\|\cdot\|_2$ denotes the Euclidean norm.

Proposition 1 identifies distinct effects of gain and shift on sinusoidal composition. Gain changes the input scale and reshapes the harmonic coefficients, with their mean squared order scaling with $g_{k,j}(x)^2$ for fixed mixing weights. Shift changes the phase offset and the relative contributions of even and odd total orders $\sum_m \kappa_m$ in the sine response, without changing the moment in Eq. (6). Separate controllers learn these adjustments for each channel across locations and stages, even when W is held fixed. Detailed derivations and the local spectral approximation are provided in Appendices A.3 and A.4.

3.3 Channel reach and shared mixing

Gain and shift control harmonic contributions from the connected channels. **Reach** describes which incoming channels are available for this interaction: mixed feature j receives the channels $\mathcal{R}_j = \{m : W_{jm} \neq 0\}$. A dense matrix connects all incoming channels, whereas a diagonal matrix connects each feature only to its corresponding channel. Weight values also affect the coefficients analyzed above; reach describes the available connections. The following proposition examines the same controlled sine in Eq. (4) over independent incoming phases, with the mixing weights held fixed.

Proposition 2 (Harmonic reach with fixed mixing)**.** *Fix W, a row j, and independent phase variables $\theta = (\theta_1, \ldots, \theta_C) \in [0, 2\pi)^C$. For scalar gain g and shift s, define*

$$F_j(\theta; g, s) = \sin\left(\omega_0 s + \omega_0 g \sum_{m=1}^{C} W_{jm} \sin\theta_m\right). \tag{7}$$

Fourier coefficients of F_j in θ with $\kappa_m \neq 0$ outside $\mathcal{R}_j$ vanish. For almost every pair (g, s), every coefficient involving only connected channels is nonzero.

Proposition 2 shows that a shared dense W retains the available harmonic interactions in the independent-phase expansion. A random matrix with continuously distributed entries is dense with probability one, so freezing it retains this interaction set without learning the mixing weights. Diagonal mixing removes cross-channel interactions from the expansion. Their magnitudes and realization in space still depend on the weights and spatial phases (tested in Section 5.1). The proof is in Appendix A.2. Together, the propositions motivate allocating learned capacity to spatial spectral assignment: shared dense mixing retains channel reach, while stage-specific gain and shift control harmonic contributions across coordinates. Sharing also leaves more of a fixed budget for controllers and additional stages.

Training. We train W, the controllers and the output layer jointly for each signal. Image fitting minimizes the mean squared reconstruction error; frozen-mixing ablations keep W at its random initialization. Task-specific input and output mappings and optimization settings are reported with the experiments. Detailed explantion can be found in Appendix B.

4 Experiments

We evaluate SPAN-INR on image fitting, super-resolution, denoising, 3D occupancy and novel-view synthesis. Our baselines include SIREN (Sitzmann et al., 2020), Gauss (Ramasinghe & Lucey, 2022), BACON (Lindell et al., 2022), WIRE (Saragadam et al., 2023), FINER (Liu et al., 2024), HOSC (Serrano et al., 2024), SL^{2}A-INR (Rezaeian et al., 2025), QuadINR (Zhou et al., 2025), SASNet (Feng et al., 2025), and FLAIR (Ko et al., 2026).

Table 1: Image fitting on Kodak, and super-resolution and denoising on DIV2K. Best values are in bold; PSNR in dB. Super-resolution compares $\times 2$ and $\times 4$ upscaling with a model trained for $\times 2$; denoising fits images with Gaussian noise ($\sigma = 0.1$) and is evaluated against the clean images. All methods have matched parameter budgets.

Method	Image Fitting			Super-Resolution						Image Denoising		
				$\times 2$			$\times 4$					
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
QuadINR	25.97	0.687	0.513	27.19	0.765	0.425	26.17	0.696	0.559	25.40	0.691	0.471
WIRE	26.43	0.686	0.469	23.81	0.637	0.550	23.25	0.631	0.616	23.99	0.668	0.406
BACON	27.12	0.697	0.364	25.39	0.644	0.585	24.71	0.597	0.685	24.59	0.617	0.351
Gauss	28.50	0.755	0.284	26.22	0.693	0.459	25.38	0.634	0.607	22.12	0.491	0.489
SIREN	29.47	0.819	0.315	27.09	0.770	0.432	26.09	0.700	0.565	25.30	0.693	0.466
SL^2A	30.52	0.819	0.206	26.31	0.710	0.453	25.44	0.661	0.563	25.80	0.677	0.307
HOSC	31.08	0.830	0.169	25.71	0.610	0.457	24.89	0.533	0.559	21.95	0.474	0.509
FLAIR	31.25	0.881	0.194	27.21	0.802	0.294	26.29	0.757	0.399	25.22	0.701	0.319
FINER	32.52	0.868	0.130	26.84	0.742	0.497	25.89	0.680	0.620	25.70	0.700	0.434
SASNet	36.95	0.932	0.081	29.00	0.823	0.246	27.70	0.750	0.407	26.65	0.700	0.274
SPAN-INR	**38.90**	**0.955**	**0.041**	**29.55**	**0.839**	**0.186**	**28.13**	**0.763**	**0.343**	**27.83**	**0.798**	**0.205**

4.1 Experimental settings

Implementation. We implement SPAN-INR in PyTorch (Paszke et al., 2019). All stages share a learned mixing matrix W, and each stage uses an MLP-based coordinate controller M_k. For all baseline methods, we use the official implementations when available and otherwise follow the configurations specified in their respective papers. All methods are configured with approximately matched parameter budgets for each task. Additional implementation details, task-specific configurations, evaluation protocols, and ablation settings are provided in Appendix B.

Evaluation metrics. For image-based tasks, we report peak signal-to-noise ratio (PSNR) (Wang & Bovik, 2009), structural similarity (SSIM) (Wang et al., 2004), and learned perceptual image patch similarity (LPIPS) (Zhang et al., 2018). We use the same metrics to evaluate rendered novel views for NeRF. For 3D occupancy, we report intersection over union (IoU), averaged over the evaluated shapes.

4.2 Results

Image fitting. We fit the full-resolution Kodak images[1] using approximately 126K parameters. Table 1 summarizes the average reconstruction performance. SPAN-INR achieves 38.90 dB PSNR, together with the highest SSIM and lowest LPIPS among the compared methods. Figure 3 presents a qualitative comparison on Kodak 20. In the enlarged region, SPAN-INR more faithfully reconstructs the curved panel seams and surrounding fine structures of the engine cowling, whereas several baselines blur or distort these details.



Figure 3: Image fitting on Kodak 20. Ground truth (GT) and reconstructions with PSNR (dB); insets enlarge the boxed region. SPAN-INR attains the highest PSNR and recovers finer detail.

Super-resolution. We train on DIV2K (Agustsson & Timofte, 2017) for $\times 2$ super-resolution using approximately 65K parameters, and evaluate the fitted model at both $\times 2$ and $\times 4$. SPAN-INR achieves 29.55 and 28.13 dB PSNR, respectively, with higher reported SSIM and lower reported

[1] https://r0k.us/graphics/kodak/



Figure 4: $\times 2$ super-resolution on DIV2K. Reconstructions of the butterfly image with PSNR (dB); SPAN-INR attains the highest PSNR of the compared methods.

LPIPS than the listed baselines at both scales (Table 1). Figure 4 shows the $\times 2$ reconstruction of the butterfly image.

Denoising. We corrupt DIV2K images with Gaussian noise at $\sigma = 0.1$ and fit the noisy RGB values using approximately 128K parameters. Reconstruction quality is measured against the clean images. As shown in Table 1, SPAN-INR achieves 27.83 dB PSNR, together with higher reported SSIM and lower reported LPIPS than the listed baselines.

Table 2: 3D occupancy and novel-view synthesis. Mean IoU@0.5 over three shapes and NeRF metrics over five Blender scenes. PSNR is in dB.

	Occupancy IoU↑		NeRF		
Method	256^3	512^3	PSNR↑	SSIM↑	LPIPS↓
QuadINR	0.9423	0.9405	28.80	0.951	0.031
WIRE	0.9383	0.9360	28.52	0.952	0.032
BACON	0.9925	0.9699	27.93	0.933	0.049
Gauss	0.9145	0.9040	28.58	0.952	0.041
SIREN	0.9400	0.9380	28.97	0.955	0.031
SL^2A	0.9932	0.9805	29.49	0.959	0.025
HOSC	0.9207	0.9181	28.97	0.951	0.032
FLAIR	0.9757	0.9704	29.68	0.959	0.027
FINER	0.9764	0.9686	29.40	0.958	0.026
SASNet	0.9955	0.9851	27.68	0.941	0.041
SPAN-INR	**0.9972**	**0.9867**	**29.99**	**0.963**	**0.023**

3D occupancy. We reconstruct Armadillo, Dragon and Thai Statue at 256^3 and 512^3 resolutions. All methods use approximately 300K parameters and minimize binary cross-entropy under the same training protocol. Table 2 reports IoU averaged across the three shapes. SPAN-INR achieves 0.9972 and 0.9867 at the two resolutions, respectively, giving the highest mean IoU among the listed methods. Figure 5 shows the reconstructed meshes.

Novel-view synthesis. For novel-view synthesis with NeRF (Mildenhall et al., 2021), the baseline evaluation uses five Blender scenes, with 25 training views at 200×200 resolution and approximately 500K parameters. Table 2 reports the five-scene means. SPAN-INR achieves 29.99 dB PSNR, 0.963 SSIM and 0.023 LPIPS, the best values among the listed methods. Figure 6 shows a held-out view.

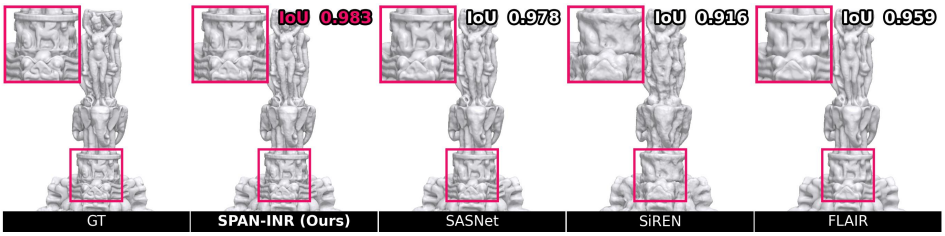


Figure 5: Thai Statue occupancy at 512^3. Ground truth and reconstructed meshes with whole-shape IoU@0.5; SPAN-INR attains the highest IoU and recovers finer surface detail.

5 Ablation Study

5.1 Component Ablations

Figure 7a retrains each variant on the 24 Kodak images at approximately 126K parameters, the budget of SPAN-INR, changing one component at a time and resizing each variant to the same budget.

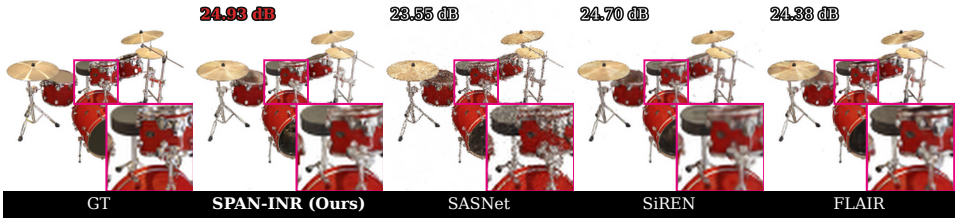
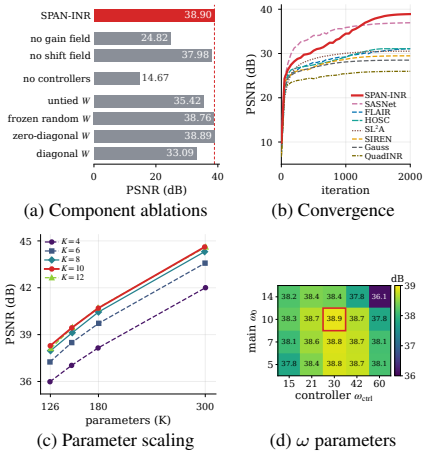


Figure 6: Novel-view synthesis on Blender drums. A held-out test view with PSNR (dB); SPAN-INR attains the highest PSNR and recovers finer detail.

Spatial control. Removing the gain field costs 14.08 dB, and removing the shift field costs 0.92 dB. Without the controllers, W is still learned and the budget goes to a wider recurrence, yet PSNR falls to 14.67 dB. Both fields contribute, and gain carries most of the effect, consistent with Proposition 1: gain sets the harmonic content of the next sine, while shift sets its phase. Section 6 examines both roles in the trained models.

Channel mixing. What matters in W is that channels reach each other. Keeping only the diagonal of W, so that no channel reaches another, costs 5.81 dB, whereas removing only the diagonal costs 0.01 dB. Once channels are connected, learning the exact values of W adds little: freezing a dense W at its random initialization costs 0.14 dB.

Weight sharing. A separate mixing matrix per stage does not help: at the same parameter count, the untied-W variant is 3.48 dB below SPAN-INR.



(a) Component ablations (b) Convergence

(c) Parameter scaling (d) ω parameters

Figure 7: Ablations and parameter studies. (a) Matched-budget ablations of spatial control, channel mixing, and weight sharing. (b) Mean PSNR over training iterations. (c) Parameter scaling over stage count and feature width at different budgets. (d) Mean PSNR over the 24 Kodak images for each ω_0 and ω_{ctrl}; the red box marks SPAN-INR.

5.2 Convergence, Scaling and Sine Scales

Convergence. Figure 7b tracks mean PSNR over training iterations on the 24 Kodak images. During early training, SPAN-INR achieves higher PSNR than all baselines except SASNet. SASNet rises faster initially, passing 33 dB at recorded iteration 400 compared with iteration 850 for SPAN-INR, but it levels off at 36.95 dB. SPAN-INR overtakes SASNet from iteration 1,250 onward and reaches 38.90 dB after 2,000 iterations.

Parameter scaling. SPAN-INR consists of K spatially controlled stages with C feature channels. We examine how its performance scales as the parameter budget increases across different depths and feature widths. Figure 7c varies both the parameter budget and the number of stages on Kodak images 1–8. Mean PSNR increases consistently as the parameter budget grows, showing that SPAN-INR scales with model capacity. At a given budget, allocating more capacity to additional stages generally improves reconstruction, with performance increasing up to around $K = 10$ before saturating.

Choosing ω_0 and ω_{ctrl}. SPAN-INR uses two sine scales: ω_0 sets the scale of the stage-wise sinusoidal composition, and ω_{ctrl} sets the sine scale inside each coordinate controller. Figure 7d varies ω_0 from 5 to 14 and ω_{ctrl} from 15 to 60 (20 settings, all 24 Kodak images). The default ($\omega_0 = 10$, $\omega_{\text{ctrl}} = 30$) gives the highest mean PSNR, and performance stays close to it across all ω_{ctrl} values when $\omega_0 \leq 10$; it drops at larger ω_0.

6 Analysis

Section 5 showed which parts of SPAN-INR are needed. This section looks at what they do inside the models fitted to the 24 Kodak images (Figure 8).

6.1 Spatial spectral assignment

By Proposition 1, gain controls the harmonic-order moment of the next sine, while shift controls its phase and the even/odd balance. We examine whether these quantities are associated with the measured local spatial spectrum. In every 32×32 window of an image we measure the gain and the shift, the next sine's local frequency and even/odd balance, and the image's edge energy. Then correlate them across the windows of that image (Figure 8a). The gain correlates positively with the local frequency and with edge energy, and both correlations strengthen with depth (blue). The shift correlates strongly with the even/odd balance from the second stage on (orange).

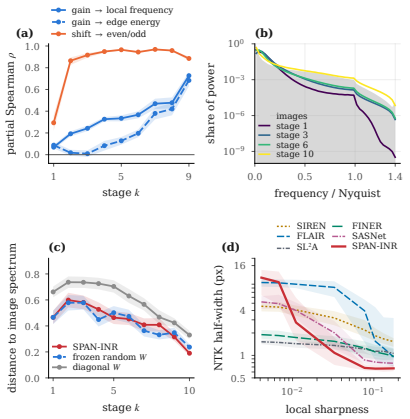


Figure 8: Spectral control and spatial adaptation. (a) Correlation of gain and shift with the next sine's spectrum and the image's edges. (b) State spectra; grey: images. (c) Distance to the image spectrum for three forms of W. (d) NTK half-width versus local sharpness. Results use 24 Kodak images. Shading: 95% confidence intervals in (a,c); interquartile ranges in (d).

6.2 Frequency growth and channel reach

Frequency growth. Figure 8b shows how the power of the state u_k is spread over spatial frequency at stages 1, 3, 6 and 10. The first stage only mixes sinusoids of the coordinates, so its power lies almost entirely at low frequencies. Each later stage passes the state through the sine again, which adds higher frequencies. From the second stage on, the distance between the two spreads shrinks and reaches its minimum at the end (Figure 8c, red).

Channel reach. Figure 8c plots the same distance for models that differ from SPAN-INR only in W (Section 5.1). A frozen random W stays close to the learned one throughout, so learning W contributes little to the spectrum's growth. A diagonal W removes all cross-channel combinations (Proposition 2). Its state carries more mid- and high-frequency power than the image from the first stage on and stays farther from the image spectrum at every stage; the finer-scale variation of its early states is visible in Figure 9.

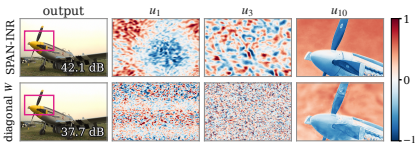


Figure 9: Hidden-state patterns on Kodak 20. Top: SPAN-INR; bottom: diagonal W. Left: reconstructions and PSNR. State panels show the first principal component over the 34 channels in the marked crop, normalized separately for display.

6.3 Spatial adaptation

The empirical neural tangent kernel (NTK) (Jacot et al., 2018) measures how strongly a gradient step at one coordinate changes the output at another, so the width of its profile shows how local each update is. Figure 8d plots this width against the target's local sharpness. SPAN-INR's updates are local where the image is sharp and broad where it is smooth: the half-width is 0.69 pixels at sharp locations and 6.20 at smooth ones, a smooth/sharp ratio of 9.0, against 4.8 for SASNet and at most 2.0 for SIREN, FINER and SL^{2}A.

7 Conclusion

We presented SPAN-INR, a sinusoidal INR that separates reach from assignment: a single shared mixing matrix W connects the channels, and a coordinate controller per stage predicts a gain and a shift that set the harmonic order and phase of the next sine. In our experiments, reach requires only connectivity, since a frozen random W nearly matches a learned one, whereas assignment has to be learned: removing the gain field or the controllers lowers PSNR substantially. This offers a new design choice for sinusoidal INRs, spending parameters on spatial assignment rather than on a separate matrix per layer; SPAN-INR achieves the best results among the compared methods on five reconstruction tasks at compact budgets.

AI USE STATEMENT

We received assistance from AI agents with the refinement and polishing of the manuscript, figures, code, theoretical analyses and proofs, experimental design, and result interpretation. The authors directed the work, reviewed and verified all AI-assisted material, and remain responsible for the final content of the paper.

ETHICS STATEMENT

This work evaluates a general-purpose implicit neural representation on public benchmarks (Kodak, DIV2K, the Armadillo, Dragon and Thai Statue scans, and the Blender scenes) and collects no new data. Any use of the method should respect the licenses of the data involved. We are aware of no other ethical risks specific to it.

REPRODUCIBILITY STATEMENT

All details needed to reproduce the results, including model configurations, optimization settings, task protocols, and proofs, are given in the main paper and the appendices. Code will be released upon acceptance.

REFERENCES

Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *2017 IEEE conference on computer vision and pattern recognition workshops (CVPRW)*, pp. 1122–1131. IEEE, 2017.

John R Carson. Notes on the theory of modulation. *Proceedings of the institute of radio engineers*, 10(1):57–64, 1922.

Haoan Feng, Diana Aldana, Tiago Novello, and Leila De Floriani. Sasnet: Spatially-adaptive sinusoidal networks for inrs. *arXiv preprint arXiv:2503.09750*, 2025.

Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. *Advances in neural information processing systems*, 31, 2018.

Sukhun Ko, Seokhyun Youn, Dahyeon Kye, Kyle Min, Chanho Eom, and Jihyong Oh. Flair: Frequency-and locality-aware implicit neural representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4877–4887, 2026.

Henry J Landau and Henry O Pollak. Prolate spheroidal wave functions, fourier analysis and uncertainty—iii: the dimension of the space of essentially time-and band-limited signals. *Bell System Technical Journal*, 41(4):1295–1336, 1962.

David B Lindell, Dave Van Veen, Jeong Joon Park, and Gordon Wetzstein. Bacon: Band-limited coordinate networks for multiscale scene representation. In *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 16231–16241. IEEE, 2022.

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 variableperiodic activation functions. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2713–2722. IEEE, 2024.

Ishit Mehta, Michaël Gharbi, Connelly Barnes, Eli Shechtman, Ravi Ramamoorthi, and Manmohan Chandraker. Modulated periodic activations for generalizable local functional representations. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 14194–14203. IEEE, 2021.

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.

Tiago Novello, Diana Aldana, Andre Araujo, and Luiz Velho. Tuning the frequencies: Robust training for sinusoidal neural networks. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3071–3080. IEEE, 2025.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

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

Reza Rezaeian, Moein Heidari, Reza Azad, Dorit Merhof, Hamid Soltanian-Zadeh, and Ilker Hacihaliloglu. Sl 2 a-inr: Single-layer learnable activation for implicit neural representation. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 26065–26074. IEEE, 2025.

Vishwanath Saragadam, Daniel LeJeune, Jasper Tan, Guha Balakrishnan, Ashok Veeraraghavan, and Richard G Baraniuk. Wire: Wavelet implicit neural representations. In *2023 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 18507–18516. IEEE, 2023.

Danzel Serrano, Jakub Szymkowiak, and Przemyslaw Musialski. Hosc: A periodic activation function for preserving sharp features in implicit neural representations. *arXiv preprint arXiv:2401.10967*, 2024.

Harinandan Shukla, Rajarshi Verma, and Jitin Singla. Fire: Frequency reparameterization as a preconditioner for periodic implicit neural representations. *arXiv preprint arXiv:2606.29414*, 2026.

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.

Zhou Wang and Alan C Bovik. Mean squared error: Love it or leave it? a new look at signal fidelity measures. *IEEE signal processing magazine*, 26(1):98–117, 2009.

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.

Gizem Yüce, Guillermo Ortiz-Jiménez, Beril Besbinar, and Pascal Frossard. A structured dictionary perspective on implicit neural representations. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19206–19216. IEEE, 2022.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *2018 IEEE/CVF conference on computer vision and pattern recognition*, pp. 586–595. IEEE, 2018.

Wenyong Zhou, Boyu Li, Jiachen Ren, Taiqiang Wu, Zhilin Ai, Zhengwu Liu, and Ngai Wong. Quadinr: Hardware-efficient implicit neural representations through quadratic activation. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 72(11):1600–1604, 2025.

A PROOFS AND DERIVATIONS

Definition 1 (Local spectrum and assignment)**.** Fix a spatial window w_Δ of width Δ, frequency bins $\mathcal{B}$, and a signal f. Define

$$V_f(x,\beta) = \int f(t) w_\Delta(t-x) e^{-2\pi i \beta \cdot t}\, dt, \qquad p_\beta(x) = \frac{|V_f(x,\beta)|^2}{\sum_{\gamma\in\mathcal{B}} |V_f(x,\gamma)|^2},$$

where the denominator is nonzero; sampled signals use the corresponding windowed DFT. Variation of $p_\beta(x)$ across windows is *spatial spectral assignment*. Comparisons use the same window, bins, and normalization. This definition describes the represented signal; a network can realize such variation through coordinate modulation or interference among global components.

A.1 HARMONIC COMPOSITION

The analysis builds on the amplitude–phase expansion of Novello et al. (2025). For a neuron

$$h(x) = \sin\left(b + \sum_{m=1}^{M} a_m \sin \varphi_m(x)\right),$$

the following finite-input identity corresponds to TUNER's Lemma 1.

Lemma 1 (Harmonic composition)**.** *For finite M, real constants a_m, b, and real phase functions $\varphi_m(x)$,*

$$\begin{aligned} h(x) &= \sum_{\kappa\in\mathbb{Z}^M} c_\kappa \sin\big(b + \kappa^\top \varphi(x)\big), \\ c_\kappa &= \prod_{m=1}^{M} J_{\kappa_m}(a_m), \end{aligned} \tag{8}$$

Here $J_n(a)$ is the Bessel function of the first kind of integer order n. It is the coefficient of $e^{in\theta}$ in the Jacobi–Anger expansion of $e^{ia\sin\theta}$ below: a sets the input sinusoid's amplitude and n selects its harmonic order.

Proof of Lemma 1. Apply the Jacobi–Anger identity

$$e^{ia\sin\theta} = \sum_{n\in\mathbb{Z}} J_n(a) e^{in\theta}$$

to each factor of $e^{ib}\prod_m e^{ia_m \sin\varphi_m(x)}$. Multiplying the absolutely convergent expansions and taking the imaginary part yields Eq. (8). □

Theorem A (Deep-network expansion, from TUNER (Novello et al., 2025))**.** *For a finite-depth sinusoidal MLP whose coordinate dependence enters through a sinusoidal input layer with frequencies ω, hidden neurons admit amplitude–phase expansions at integer combinations of ω. Their coefficients are products of Bessel factors determined by hidden weights and expansion indices; biases enter the phases.*

The expansion lists candidate frequencies; terms at the same frequency combine into one Fourier coefficient.

A.2 HARMONIC REACH UNDER FIXED MIXING

Fix row $w = W_{j,:}$ and write $a = \omega_0 g$ and $b = \omega_0 s$. For independent phases $\theta \in [0, 2\pi)^C$, let

$$F(\theta) = \sin\left(b + a\sum_{m=1}^{C} w_m \sin\theta_m\right), \qquad B_\kappa(a) = \prod_{m=1}^{C} J_{\kappa_m}(a w_m).$$

The Fourier convention is

$$\widehat{F}(\kappa) = (2\pi)^{-C} \int_{[0,2\pi)^C} F(\theta) e^{-i\kappa^\top \theta}\, d\theta,$$

where $i^2 = -1$. Expanding the positive and negative complex exponentials of the sine gives

$$\widehat{F}(\kappa) = B_\kappa(a) \begin{cases} \sin b, & \sum_m \kappa_m \text{ even}, \\ -i \cos b, & \sum_m \kappa_m \text{ odd}. \end{cases} \tag{9}$$

Proof of Proposition 2. If $w_m = 0$ and $\kappa_m \neq 0$, the factor $J_{\kappa_m}(0)$ vanishes. Thus every coefficient involving a phase outside $\mathcal{R}_j$ is zero. For a supported index, the inactive factors are $J_0(0) = 1$. Every active factor $J_{\kappa_m}(aw_m)$ is a nonzero analytic function of a: for $n \geq 0$, its power series starts with $(aw_m/2)^n/n!$, and negative orders follow from $J_{-n} = (-1)^n J_n$. Its real zeros are therefore discrete. The union of these zero sets over finitely many active channels and countably many orders is countable. Outside this set, all supported products $B_\kappa(a)$ are nonzero simultaneously. Excluding $b \in (\pi/2)\mathbb{Z}$ also makes both parity factors nonzero. The exceptional pairs (a, b) have two-dimensional Lebesgue measure zero. Since $\omega_0 > 0$, the same statement holds for (g, s). If $\mathcal{R}_j$ is empty, only the zero index remains, with coefficient $\sin b$, and the conclusion still holds. □

For a finite matrix whose entries have continuous distributions, the probability that any entry is zero is zero; independence of the entries is unnecessary. Each dense row therefore has the full independent-phase index set $\mathbb{Z}^C$. Reusing these rows preserves this index set as a possibility at every stage whose output enters a subsequent sine. A diagonal row instead supports indices on its own coordinate axis only.

Scope. Genericity refers to the scalar parameter pair (g, s), not to coordinates or trained controller outputs. A controller may take exceptional values, including $g = 0$. The coefficient magnitudes and their joint controllability depend on the weights, and all arguments aw_m share the same scalar a. Moreover, substituting spatial phase functions can collapse distinct phase harmonics: if $w = (1, -1)$ and $\varphi_1(x) = \varphi_2(x)$, the spatial response is the constant $\sin b$. Coordinate-dependent controllers can also introduce spatial content independently of incoming-state mixing. These distinctions prevent interpreting Proposition 2 as a theorem about reconstruction accuracy, realized spatial bandwidth, or equivalence to arbitrary untied matrices.

SUPPORT OF MIXED FEATURES

Lemma 2 (Support of mixed features)**.** *Let fixed finite trigonometric features v_i have frequency supports S_i. For $q_j = \sum_i W_{ji} v_i$,*

$$\operatorname{supp}(q_j) \subseteq \bigcup_{i : W_{ji} \neq 0} S_i.$$

A subsequent sine of a constant affine transform of q_j has candidate frequencies in the integer span of this union. A dense row exposes every incoming channel to this combination.

Proof. Write each fixed incoming feature as a finite trigonometric sum $v_i(x) = \sum_{\beta \in S_i} c_{i,\beta} e^{i\beta \cdot x}$, with equal physical frequencies already aggregated. The coefficient of q_j at β is

$$c^q_{j,\beta} = \sum_{i : W_{ji} \neq 0} W_{ji} c_{i,\beta}.$$

This proves the support inclusion. For a fixed connection pattern and a frequency present in at least one connected channel, cancellation imposes a proper linear constraint on the allowed real row weights. The finite union of these exceptional sets has Lebesgue measure zero. The same argument covers countably many coefficients of absolutely convergent incoming series. Generic equality here holds with incoming features fixed; trained recurrent features can themselves depend on the shared W.

Applying the sine power series to a constant affine transform of a finite trigonometric sum generates only integer combinations of its frequencies. For example, $\sin(a \sin\theta)$ generates odd harmonics, illustrating that the candidate set can exceed realized support. Thus the mixing matrix W pools a union of supports and the sine forms frequency combinations. With constant fields and a diagonal mixing matrix W, channels initialized from individual carriers stay in their respective integer-generated sets. Dense mixing makes all incoming channels available for combination, while the amplitudes of the resulting components depend on the weights.

□

State dependence of a stage. For a fixed coordinate x, write the stage update with incoming feature vector u as $T_k(u; x) = D_{g_k(x)} W \sin(\omega_0 u) + s_k(x)$, where D_v is diagonal with entries v. The fields depend on x, so differentiation with respect to u gives

$$\boxed{\frac{\partial T_k}{\partial u}(u; x) = \omega_0 D_{g_k(x)} W D_{\cos(\omega_0 u)}.} \tag{10}$$

A diagonal W, including the identity, gives a diagonal state Jacobian. A dense W permits cross-channel dependence, with gains and sine derivatives determining the active connections. This statement concerns dependence on the incoming state at fixed x; the coordinate controllers can still depend on all components of x.

A.3 Coordinate-dependent harmonic coefficients

Fix a nonfinal stage $1 \le k < K$, channel j, and coordinate x. Write $d_m(x) = \omega_0 g_{k,j}(x) W_{jm}$ and $b(x) = \omega_0 s_{k,j}(x)$.

Proof of Proposition 1. Eq. (5) follows from Lemma 1 at fixed x. To obtain the coefficient moments, introduce independent phases $\theta \in [0, 2\pi)^C$ and define

$$\Psi_x(\theta) = \exp\left(i \left[b(x) + \sum_m d_m(x) \sin\theta_m \right] \right).$$

Its Fourier coefficients are $e^{ib(x)} A_\kappa(x)$. Since $|\Psi_x| = 1$, Parseval's identity under the normalized uniform phase measure gives unit squared coefficient mass. Moreover,

$$\partial_{\theta_m} \Psi_x = i\, d_m(x) \cos\theta_m\, \Psi_x,$$

whose mean squared magnitude is $d_m(x)^2/2$. Derivative Parseval and summation over m give Eq. (6). □

Additional consequences of Proposition 1

For a fixed stage, channel and coordinate, Proposition 1 gives unit coefficient mass and the harmonic-order moment. For any $N > 0$, its moment bound yields

$$\sum_{\|\kappa\|_2 > N} A_\kappa(x)^2 \le \frac{\|d(x)\|_2^2}{2N^2}.$$

This follows by bounding each term in the tail by $\|\kappa\|_2^2 A_\kappa(x)^2 / N^2$ and using Eq. (6). The gain-sign relation follows from $J_n(-z) = (-1)^n J_n(z)$: reversing gain multiplies A_κ by $(-1)^{\sum_m \kappa_m}$. The sum $\sum_m \kappa_m$ and $\|\kappa\|_1$ have the same parity.

Gain and shift for a single tone. For an incoming phase θ and scalar modulation index z,

$$\sin(z \sin\theta) = 2 \sum_{n \ge 1,\, n \text{ odd}} J_n(z) \sin(n\theta).$$

Changing z changes the harmonic coefficients, which need not vary monotonically. A phase offset b gives

$$\sin(z \sin\theta + b) = \cos b\, \sin(z \sin\theta) + \sin b\, \cos(z \sin\theta). \tag{11}$$

The first term contains odd sine harmonics; the second contains a constant term and even cosine harmonics. Thus b controls their balance: $b = 0$ selects the odd branch, while $b = \pi/2$ selects the constant/even branch.

For a multi-channel input $t(\varphi) = \sum_m d_m \sin \varphi_m$, shifting every phase by π changes t to $-t$. Consequently, $\sin t$ contains indices with odd $\sum_m \kappa_m$, while $\cos t$ contains those with even $\sum_m \kappa_m$. This parity concerns the phase variables and need not match odd/even bins in the spatial spectrum.

Real-output energy also depends on the phase offset. For a single tone,

$$\left\langle \sin^2(z \sin \theta + b) \right\rangle_\theta = \frac{1 - \cos(2b) J_0(2z)}{2}.$$

Thus unit complex-coefficient mass is compatible with changing real-output energy. At $b = 0$ and nonzero z, the output-energy fraction above the fundamental is

$$\eta_{\geq 3}(z) = 1 - \frac{J_1(z)^2}{\sum_{n \geq 1,\, n \text{ odd}} J_n(z)^2}.$$

This fraction is approximately 0.002 at $z = 1$ and remains nonzero for small nonzero z. For a generic shift, even harmonics also appear. For the complex expansion, the squared coefficient mass outside orders $-1, 0, 1$ is $1 - J_0(z)^2 - 2J_1(z)^2 = z^4/32 + O(z^6)$. Single-tone frequency modulation yields an infinite Bessel series (Carson, 1922); effective bandwidth reflects a tolerance on its tails, rather than an exact cutoff.

A.4 Local spectral approximation

Write $h_{k+1,j}(x) = \sin(\omega_0 u_{k,j}(x))$ for the controlled sine response, and let Ω denote the coordinate domain.

For a term in Eq. (5), define its phase as $\psi_\kappa(x) = \kappa^\top \varphi(x) + b(x)$. The local angular-frequency vector at coordinate x_0 is

$$\xi_\kappa(x_0) = \nabla \psi_\kappa(x_0) = \sum_m \kappa_m \nabla \varphi_m(x_0) + \nabla b(x_0). \tag{12}$$

Here ∇ differentiates with respect to the coordinates. Spatially varying shift contributes through $\nabla b(x_0)$; frequencies in cycles per coordinate unit are $\xi_\kappa(x_0)/(2\pi)$.

Let $B_\Delta(x_0) \subset \Omega$ be a closed ball, and assume $A_\kappa \in C^1$ and $\psi_\kappa \in C^2$ on a neighborhood of that ball. Freezing the coefficient and linearizing the phase gives

$$\widetilde{h}_\kappa(x) = A_\kappa(x_0) \sin\big(\psi_\kappa(x_0) + \xi_\kappa(x_0)^\top (x - x_0)\big). \tag{13}$$

For $x \in B_\Delta(x_0)$, the pointwise error satisfies

$$\begin{aligned} &\left| A_\kappa(x) \sin \psi_\kappa(x) - \widetilde{h}_\kappa(x) \right| \leq E_\kappa, \\ &\qquad E_\kappa = \Delta \sup_{B_\Delta(x_0)} \|\nabla A_\kappa\|_2 + \frac{|A_\kappa(x_0)| \Delta^2}{2} \sup_{B_\Delta(x_0)} \|\nabla^2 \psi_\kappa\|_{\text{op}}. \end{aligned} \tag{14}$$

Proof of the local approximation bound. Write $\ell_\kappa(x) = \psi_\kappa(x_0) + \xi_\kappa(x_0)^\top (x - x_0)$. Add and subtract $A_\kappa(x_0) \sin \psi_\kappa(x)$. The triangle inequality and $|\sin u - \sin v| \leq |u - v|$ give

$$\begin{aligned} \left| A_\kappa(x) \sin \psi_\kappa(x) - \widetilde{h}_\kappa(x) \right| \leq\ & |A_\kappa(x) - A_\kappa(x_0)| \\ & + |A_\kappa(x_0)|\, |\psi_\kappa(x) - \ell_\kappa(x)|. \end{aligned}$$

The mean-value bound controls the first term by $\Delta \sup \|\nabla A_\kappa\|_2$. Taylor's theorem controls the phase remainder by $\Delta^2 \sup \|\nabla^2 \psi_\kappa\|_{\text{op}}/2$, proving Eq. (14). □

Small coefficient variation and phase curvature are sufficient for a small bound on the observation scale. To approximate the full response, retain a finite index set Λ and define its uniform remainder

$$\varepsilon_{\text{tail}} = \sup_{x \in B_\Delta(x_0)} \left| h_{k+1,j}(x) - \sum_{\kappa \in \Lambda} A_\kappa(x) \sin \psi_\kappa(x) \right|.$$

Then

$$\sup_{B_\Delta(x_0)} \left| h_{k+1,j} - \sum_{\kappa \in \Lambda} \widetilde{h}_\kappa \right| \le \varepsilon_{\text{tail}} + \sum_{\kappa \in \Lambda} E_\kappa.$$

Both the summed approximation errors and the omitted terms must be controlled. Bounded modulation indices on the compact ball give uniform absolute convergence of the Bessel expansion, allowing a finite truncation with a small remainder. The squared-coefficient tail bound above concerns independent-phase coefficient mass and differs from this uniform real-signal remainder.

For an integrable window supported in the ball, the corresponding windowed-transform error is at most this uniform signal-error bound times the window's L^1 norm. Each retained local sinusoid contributes shifted copies of the window transform centered at $\pm\xi_\kappa(x_0)/(2\pi)$ in the spatial-frequency convention used above. Spatial coefficient variation can add modulation sidebands, while shift variation contributes to phase gradients. Coincident frequencies and overlapping window responses combine before their squared magnitude is measured; their squared coefficients therefore cannot generally be added as local spectral energies.

For each nonfinal stage $1 \le k < K$, the diagnostic λ_k is the channel-average harmonic-order moment of the next sine:

$$\lambda_k(x) = \frac{\omega_0^2}{2C} \sum_j g_{k,j}(x)^2 \|W_{j,:}\|_2^2.$$

This harmonic-order moment and the measured spatial-frequency centroid are different quantities. The final stage has no following sine, and its fields feed the output layer directly.

LOCAL-BANDWIDTH HEURISTIC

For an interval of length T and band $[-B, B]$ in cycles per unit coordinate, time–bandwidth concentration (Landau & Pollak, 1962) gives approximately $2TB$ well-concentrated modes, subject to concentration tolerance and asymptotic corrections. Divide a signal domain into regions of lengths L_r with chosen bandwidths B_r. Before accounting for gates and boundaries, the corresponding allocation counts are

$$D_{\text{global}} \approx 2LB_{\max}, \qquad D_{\text{local}} \approx 2\sum_r L_r B_r, \qquad \frac{D_{\text{global}}}{D_{\text{local}}} \approx \nu,$$

where $L = \sum_r L_r$, $\overline{B} = \sum_r L_r B_r / L > 0$, and $\nu = B_{\max}/\overline{B}$. Equality $\nu = 1$ means that the chosen bandwidth statistic is constant across regions; statistical stationarity is a stronger condition.

These counts describe concentrated linear function classes. A compact gate introduces spectral tails and incurs representation and boundary-error costs, while an individual target may have a sparse representation within a much larger class. The analogous two-dimensional count uses frequency area. We therefore use ν only as a diagnostic of uneven local bandwidth; the counting argument supplies no INR parameter or approximation-rate bound.

A.5 EQUIVALENCE UNDER ROW SCALING

Gain scales each mixed feature as a whole, so at stage k the effective mixing matrix is

$$W_{\text{eff},k}(x) = D_{g_k(x)} W, \tag{15}$$

where D_v is the diagonal matrix with entries v: each row of W is rescaled, and the relative weights within a row are kept. A stage-specific rescaling of the rows of a shared matrix is therefore already available to the controllers, which is the class of untied matrices the next result covers.

Proposition 3 (Equivalence under row scaling)**.** *Consider an untied model with matrices $\widetilde{W}_k = D_{a_k} W$, constant vectors $a_k \in \mathbb{R}^C$, and fields $\widetilde{g}_k, \widetilde{s}_k$. With the same initial state, sine scale, and output layer, a shared-matrix model produces identical stage updates using*

$$\boxed{g_k(x) = a_k \odot \widetilde{g}_k(x), \qquad s_k(x) = \widetilde{s}_k(x),} \tag{16}$$

provided its controllers can rescale each gain output by a constant without changing the shift outputs.

Proof of Proposition 3. At each stage,

$$D_{\widetilde{g}_k(x)}\widetilde{W}_k = D_{\widetilde{g}_k(x)}D_{a_k}W = D_{a_k\odot\widetilde{g}_k(x)}W.$$

The same identity applies to the coordinate mixture at the first stage, giving identical $u_1(x)$. The subsequent updates agree for identical incoming feature states, so induction gives identical outputs. For the controller with gain and shift outputs, multiply the first C output-weight rows and their biases by the corresponding entries of a_k, leaving all shift-output rows and biases unchanged. Negative and zero entries are allowed. □

The construction realizes the controller-class closure required in Proposition 3: gain is scaled componentwise while the shift is held fixed. It establishes equivalence of represented functions for row-scaled mixing matrices, with initialization, regularization, and optimization trajectories allowed to differ.

SHARING ASYMMETRY

Corollary 1 (Sharing asymmetry)**.** *Sharing a dense mixing matrix W preserves the complete channel connection graph at every stage. For equal-architecture coordinate controllers, tying the fields restricts their stagewise tuples to the shared-field subset, strictly so whenever at least two stages can represent different fields.*

A dense shared mixing matrix W supplies the same complete channel connection graph at every stage. For equal-architecture coordinate controllers, every shared tuple $((g, s), \ldots, (g, s))$ is an admissible untied tuple. If the networks can produce two different fields and there are at least two stages, the untied family contains a tuple outside the shared subset. This proves strictness for the intermediate field tuples. The final represented function classes also depend on the remaining parameters, which can compensate for field tying; fixed-budget comparisons additionally change network widths. These constraints distinguish the effects of sharing mixing weights and sharing the coordinate controllers.

SMALL-DRIVE APPROXIMATION

Corollary 2 (Small-drive approximation)**.** *For $t(x) = \omega_0 g(x)q(x)$ and $b(x) = \omega_0 s(x)$,*

$$|\sin(b + t) - \sin b - t\cos b| \le \tfrac{1}{2}t^2, \qquad |\sin t - t| \le \tfrac{1}{6}|t|^3.$$

Thus small drive suppresses higher nonlinear terms continuously.

Taylor's theorem bounds the first-order remainder by $t^2/2$, since $|\sin'' | \le 1$. At $b = 0$, the quadratic term vanishes and the third derivative is bounded by one, yielding $|t|^3/6$. The response approaches an affine function of the state continuously as $t \to 0$. Coordinate-dependent fields can still introduce spatial frequencies in this regime; there is no unit-drive cutoff, and later propagation depends on the gains and mixing weights.

HYPOTHESIS 1 (DRIVE ATTENUATION)

In a near-linear chain with constant attenuation factors and no additive sources, multiplying a stage output by $m_k \in [0, 1]$ scales the next stage's input-dependent drive by the same factor. Repeated attenuation carries a product of these factors downstream. This mechanism suggests that output attenuation can weaken subsequent nonlinear transformations. Establishing its role in trained baselines requires intermediate-drive measurements: spatial masks add frequencies, and learned weights or additive paths can compensate for attenuation. Per-stage gains provide direct control over the input-dependent modulation scale before each subsequent sine.

B EXPERIMENTAL DETAILS

B.1 MODEL CONFIGURATIONS

Table 3 summarizes the model configurations used for the five experimental tasks.

Table 3: Task configurations. K is the number of stages, C the feature width and h the controller hidden width; ω_0 and ω_{ctrl} are the stage and controller sine scales. For 3D occupancy, ω_0 is 30 at 256^3 and 25 at 512^3.

Task	K	C	h	ω_0	ω_{ctrl}	Trainable parameters
Image fitting	10	34	81	10	30	125,871
Super-resolution	8	90	32	10	30	65,109
Denoising	8	53	81	2	15	127,587
3D occupancy	8	315	64	30 / 25	30	∼300K
Novel-view synthesis	4	290	96	18	30	∼500K

Optimization. For image fitting, denoising, and super-resolution, we use 2,000 full-batch iterations with mean squared error and cosine learning-rate decay. Image fitting uses AdamW with an initial learning rate of 3×10^{-3} for all parameters. Denoising uses AdamW with 2.1×10^{-3} for coordinate controllers and 3×10^{-3} for the remaining parameters. Super-resolution uses RAdam with 4.2×10^{-3} for coordinate controllers and 6×10^{-3} for the remaining parameters. For 3D occupancy, we use balanced minibatches of 200,000 voxels and 3,000 Adam iterations with a binary cross-entropy loss on the occupancy logits. SPAN-INR uses Adam with a learning rate of 2×10^{-4} at both resolutions. For novel-view synthesis, we use 30,000 Adam iterations with a mean squared error loss on the rendered pixel colors, a learning rate of 3×10^{-4} (2.1×10^{-4} for the coordinate controllers), decayed to one tenth over training.

B.2 Task protocols

Image fitting. We use all 24 Kodak images at native resolution. The objective is RGB reconstruction error at the observed pixel coordinates.

Super-resolution. We use 40 DIV2K images, resized to a longest side of 2,048 pixels. A $\times 4$ average-pooled image supplies the observation. Training compares each observed pixel with the average of four predictions on the $\times 2$ grid. The fitted representation is evaluated at both $\times 2$ and $\times 4$ magnification.

Denoising. We resize 40 DIV2K images to a longest side of 512 pixels, add Gaussian noise with $\sigma = 0.1$, and clip the observations to $[0, 1]$. Training minimizes error against the noisy image; evaluation uses the clean image.

3D occupancy. We reconstruct Armadillo, Dragon and Thai Statue at 256^3 and 512^3 resolutions. Evaluation computes full-grid IoU at an occupancy threshold of 0.5.

Novel-view synthesis. The baseline NeRF evaluation uses five Blender scenes: chair, drums, ficus, materials and mic. Training uses 25 views at 200×200 resolution, and evaluation uses the 200 test views of each scene.

B.3 Ablation settings

Matched-budget ablations (Figure 7a). All variants use the architecture of Section 3.1, $\omega_0 = 10$, controller sine scale 30, and 2,000 full-batch iterations on native-resolution Kodak, same as the configuration of image fitting in Section B.1. Optimization uses mean squared error, AdamW at 3×10^{-3} with zero weight decay, cosine learning-rate decay and gradient-norm clipping at 1. The diagonal variant learns only the diagonal entries of shared W and keeps $K = 10$, $C = 34$ and $h = 81$, giving approximately 125K trainable parameters, about 1% fewer than the dense reference. All PSNR comparisons are paired by image. Two-sided Wilcoxon tests use the 24 per-image differences without multiplicity adjustment.

B.4 Convergence and scaling protocols

Convergence trajectories (Figure 7b). All methods are trained with matched parameter budgets, and each recorded PSNR is averaged over the 24 native-resolution Kodak images. Methods are

evaluated every 50 iterations during 2,000 full-batch updates; threshold crossings refer to the first recorded point where the mean curve exceeds the threshold.

Scaling grid (Figure 7c). We use Kodak images 1–8 and the construction of the component ablations. We vary the number of stage $K \in \{4, 6, 8, 10, 12\}$ and choose feature width C to approach budgets of 126K, 150K, 180K and 300K trainable parameters. Controller width $h = 81$, stage sine scale $\omega_0 = 10$, controller sine scale $\omega_{ctrl} = 30$, and the optimization protocol of the component ablations remain fixed. All 20 configurations are evaluated on the same eight images. The plotted values average the best recorded PSNR within 2,000 iterations. Depth changes the number of controllers and, at a fixed budget, their share of the parameters; the sweep therefore evaluates depth and parameter allocation together.

C Qualitative comparisons

This appendix shows reconstructions and error maps for image fitting on Kodak and super-resolution on DIV2K, the settings of Table 1. Each row shows one window: the context panel marks its location, followed by the ground truth and the five methods, each labelled with the PSNR of the crop. Each method panel is split on the diagonal, with the reconstruction above it and the absolute error below it, in grayscale on one scale per row (brighter is larger).

C.1 Image fitting

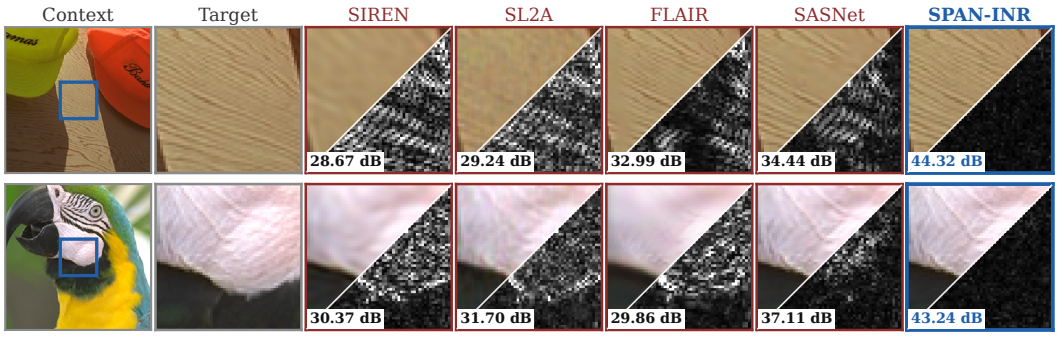


Figure 10: Image fitting on kodim03 (top) and kodim23 (bottom). Each row shows a 56×56 window. Top: the window of highest local spectral centroid in kodim03, on the wood grain of the wall; the baselines leave the grain pattern in their error. Bottom: a window on the white cheek of the macaw in kodim23; SIREN, SL^{2}A and FLAIR leave error along the boundary with the dark plumage, and SASNet across the feather texture. In both rows, SPAN-INR's error is close to unstructured.

C.2 Super-resolution

Models are fit on the $4\times$-downsampled DIV2K image and evaluated at $\times 2$ and $\times 4$.

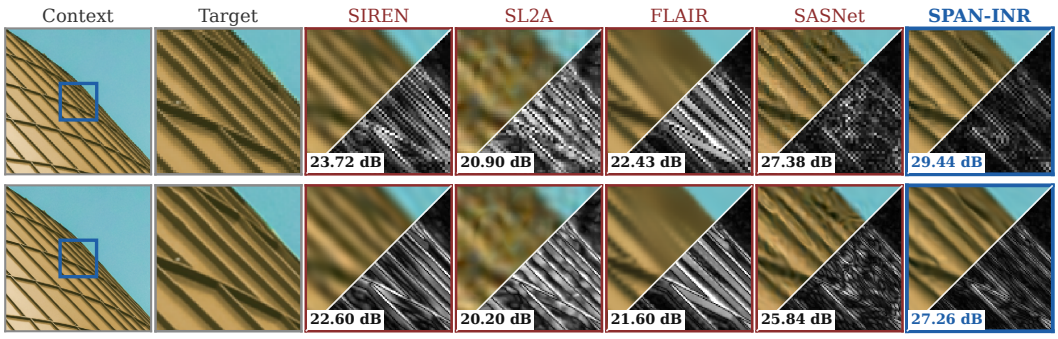


Figure 11: Super-resolution on DIV2K image 22 at $\times 2$ (top) and $\times 4$ (bottom). The window is the image's window of highest local spectral centroid on the $\times 2$ ground truth (56×56), shown at $\times 4$ as the same scene window (112×112), on the seams of the building's cladding. SIREN and FLAIR blur the seams, SL^{2}A adds colour blotches, and SASNet leaves scattered artefacts between them; SPAN-INR has the lowest error at both scales.