The Surprising Effectiveness of Noise Pretraining for Implicit Neural Representations

Supplementary Material

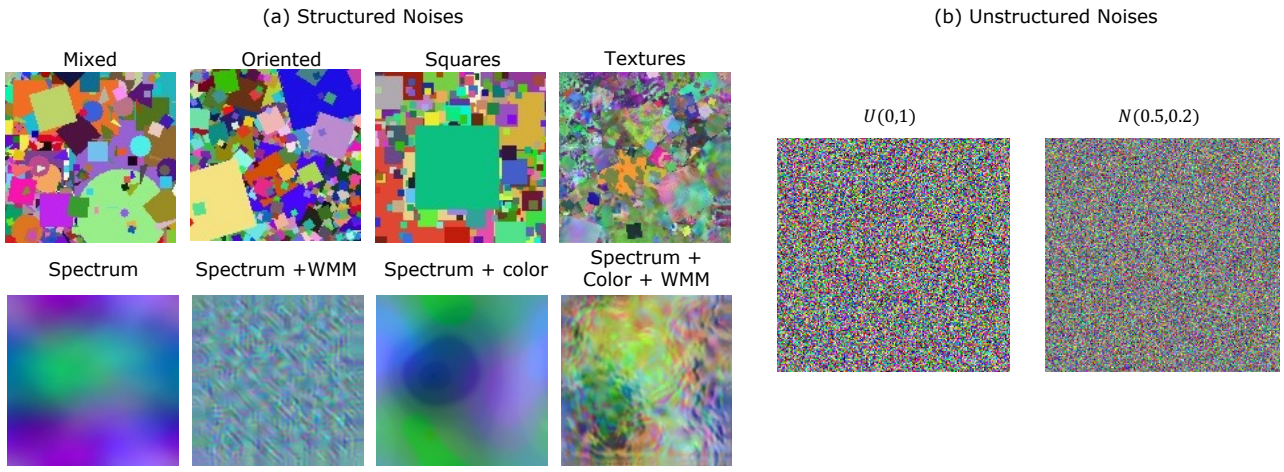

Figure 11. **Example images used** SNP **training dataset**(a) Structured noise images, from dead leaves and statistical image generation models , from Looking at Noise [4]. (b) Unstructured noise images, Uniform and gaussian noise.

5. Structured and Unstructured noises used for pretraining

Figure 11 showcase example images used for SNP pretraining for both structured (from Baradad et.al [4]) and unstructured noises (uniform and Gaussian). For more details on how structured noise images are generated, we refer the reader to Baradad et.al [4].

6. SNP pretrain and fitting procedures

We first provide the reader with an overview of conventional INRs followed by detailed training and fitting procedures for SNP INRs.

Conventional INRs. Typically an INR, denoted as $f_\theta(\cdot)$ with parameters θ learns a mapping between d dimensional input coordinates, $\mathbf{x} \in \mathcal{R}^d$, and D dimensional signal, $\mathbf{y} \in \mathcal{R}^D$. INR parameters θ are iteratively optimized by minimizing an objective function, such as mean squared error, to yield the optimal learned signal representation f_{θ^*}.

$$\theta^* = \arg\min_\theta || f_\theta(\mathbf{x}) - \mathbf{y}(\mathbf{x}) ||_2^2 \tag{1}$$

SNP INRs. SNP follows the design of the STRAINER [44] INR framework. As described in STRAINER [44], at pretraining time, SNP shares initial $L-1$ *encoder* layers while jointly fitting N signals, each with their respective signal specific decoder. At test-time, SNP retains the pretrained encoder weights as an initialization along with randomly initialized decoder and proceeds to fit an unseen signal.

We define the initial encoder layers of SNP as $f_\theta(\cdot)$. During pretraining stage, for each train signal $\mathbf{y}^i$, we define signal specific decoders denoted by $g^i_\psi(\cdot)$. We define SNP INR $h_\Theta(\cdot)$ below in Eq. 2

$$h_\Theta(\mathbf{x}) = g^i_\psi \circ f_\theta(\mathbf{x}) \tag{2}$$

where Θ is the collective set of encoder weights θ and decoder weights $\psi^1 \cdots \psi^N$.

During pretraining, we fit N signals of the same noise-type jointly to SNP for T steps, following Eq 3, to obtain learned encoder initialization θ^*

$$\Theta^* = \arg\min_\Theta \sum_{i=1}^{i=N} || g^i_\psi \circ f_\theta(\mathbf{x}) - \mathbf{y^i}(\mathbf{x}) ||_2^2 \tag{3}$$

where $\Theta^* = [\theta^*, \psi^{1*}, \cdots, \psi^{N*}]$

At test-time, SNP is initialized learned encoder weights $f_{\theta=\theta^*}$ along with a randomly initialized decoder g_ψ. Finally, we fit the test-signal for T steps to the pretrained SNP INR as shown in Eq. 4.

$$[\theta^*, \psi^*] \leftarrow \arg\min_{\theta,\psi} || g_\psi \circ f_{\theta=\theta^*}(\mathbf{x}) - \mathbf{y}(\mathbf{x}) ||_2^2 \tag{4}$$

7. Training specifications

7.1. Image and Video fitting

Image fitting. We describe the pretraining and fitting procedure for SNP INRs. For pretraining, we follow the process outlined in STRAINER [44] and use $N = 10$ images (which has previously shown success in learning high quality initialization in STRAINER) for each type of noise (illustrated in Fig. 11). At test-time, we retain the pretrained features as initialization of the *encoder* or first $L-1$ layers and randomly initialize the last *decoder* layer. We fit each test image for 2000 iterations using Adam optimizer with learning rate of 10^{-4}. All hidden layers are of width 256.

Video fitting. We adopt ResField as our video-fitting backbone to apply the SNP initialization. ResField's parameters are decomposed into temporally shared weight and per-frame low-rank residuals. We transfer the weights learned during SNP noise pre-training to the shared branch, while initializing the residual factors to zero [16, 32], i.e., $A \sim \mathcal{N}(0, \sigma^2)$, $B = 0$, $\Delta W = BA = 0$ at the first iteration. We employ an MLP with four hidden layers of width 256 for the shared weights and rank-10 residual weights. Each video is optimized for 10^5 iterations using the Adam optimizer (learning rate 5×10^{-4}) and a cosine-annealing learning-rate schedule. We perform pretraining on noise images rather than noise videos, and then successfully transfer the resulting model weights from the image domain to the video domain. This strategy is highly practical, as it substantially reduces both pre-training time and the complexity of pretraining data generation.

Image and Video Denoising. For image and video denoising, we simulate a Poisson process to add synthetic noise of photon count of 30, resulting in a read-out noise of 2 dB to each image/video frame. At test-time we initialize the SNP encoder using the corresponding weights from fitting tasks.

We use the Alpine INR framework [43] for pretraining and test-time signal fitting for SNP INRs.

8. More results on image and video fitting

Table 1 (in main text) presented a brief snapshot of our experimental evaluation. We provide a more detailed analysis in Tables 4, 6, and Table 7 and report mean values of PSNR, SSIM, and LPIPS [51] and 1 std dev. for randomly initialized SIREN, data-driven baselines such as STRAINER, and SNP variants for image fitting on CelebA-HQ,AFHQ, and OASIS-MRI datasets. We observe that SNP pretrained on uniform and Gaussian noise strikingly, achieves a PSNR of $80db+$, outperforming all other methods. On structured noise initializations, SNP: Spectrum performs best, almost matching STRAINER. Figures 13 and 12 also show image fitting trajectories for AFHQ and OASIS-MRI datasets respectively and we observe that SNP pretrained on all noise types to quickly converge to high quality signal representations. Notably, SNP: Uniform achieves SIREN level performance $20\times$ faster and STRAINER level performance $4\times$ faster when evaluated on CelebA-HQ (see Fig. 3 in main text) and on AFHQ and OASIS-MRI datasets as shown in Fig. 12 and Fig. 13 in attached supplement.

Table 4. **Image fitting results on CelebA-HQ.** We present mean PSNR(in db), SSIM and LPIPS with 1 std.dev. for all test methods at $T = 2000$ iterations, from vanilla randomly initialized (SIREN), to data driven methods such as Metalearned [41], STRAINER [44], and SNP variants trained on structured and unstructured noises.

Method	CelebA		
	PSNR↑	SSIM↑	LPIPS↓
SIREN [39]	44.9 ± 2.13	0.991 ± 0.007	0.0026 ± 0.002
STRAINER [44]	57.8 ± 3.46	0.999 ± 0.001	0.00002 ± 0.07
Meta-learned [41]	53.1 ± 3.36	0.994 ± 0.053	$0.0046 \pm 6.2\text{e-}5$
SNP: Dead leaves oriented	55 ± 2.8	0.999 ± 0.0006	0.00096 ± 0.001
SNP: Dead leaves texture	55 ± 2.9	0.999 ± 0.0006	0.0009 ± 0.001
SNP: Dead leaves mixed	54.6 ± 3.0	0.999 ± 0.0007	0.0011 ± 0.001
SNP: Dead leaves squares	54.7 ± 2.7	0.999 ± 0.0007	0.0012 ± 0.001
SNP: Spectrum	56.4 ± 3.1	0.999 ± 0.0006	0.0005 ± 0.001
SNP: Spectrum + Color	51 ± 2.5	0.998 ± 0.0011	0.0039 ± 0.003
SNP: Spectrum + WMM	54.4 ± 3.1	0.999 ± 0.0007	0.0011 ± 0.001
SNP: Spectrum + WMM + Color	55.2 ± 3.4	0.999 ± 0.0007	0.0008 ± 0.0017
SNP: Gaussian	80 ± 11.2	$0.999 \pm 4.20\text{e-}5$	$5.50\text{e-}6 \pm 1.5\text{e-}5$
SNP: Uniform	$\mathbf{85.7 \pm 12.6}$	$\mathbf{0.999 \pm 2.90}\text{e-}5$	$\mathbf{3.40}\text{e-}6 \pm \mathbf{1.10}\text{e-}5$

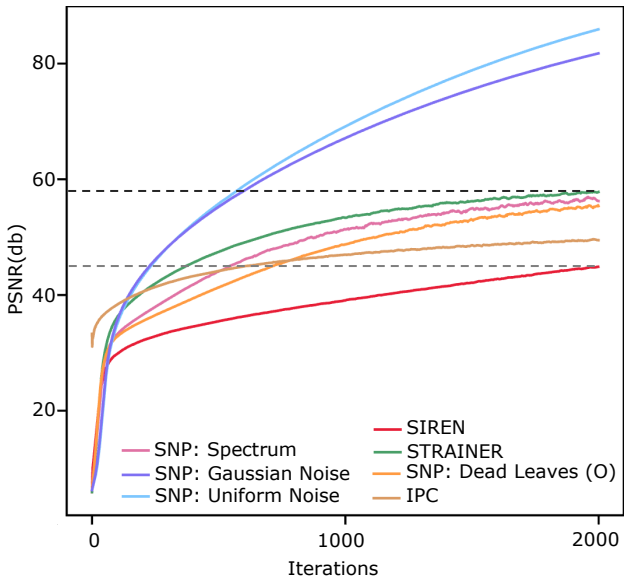

Figure 12. **Signal fitting results on AFHQ images.** SNP initialized using various noises demonstrate strong signal representation ability and convergence compared to widely known baseline methods such as SIREN [39], IPC [25], and STRAINER [44]. Interestingly, SNP trained with Gaussian and Uniform noises performed significantly better than all other approaches. converging rapidly to perfect fitting.

Table 5 summarizes video fitting results. To illustrate

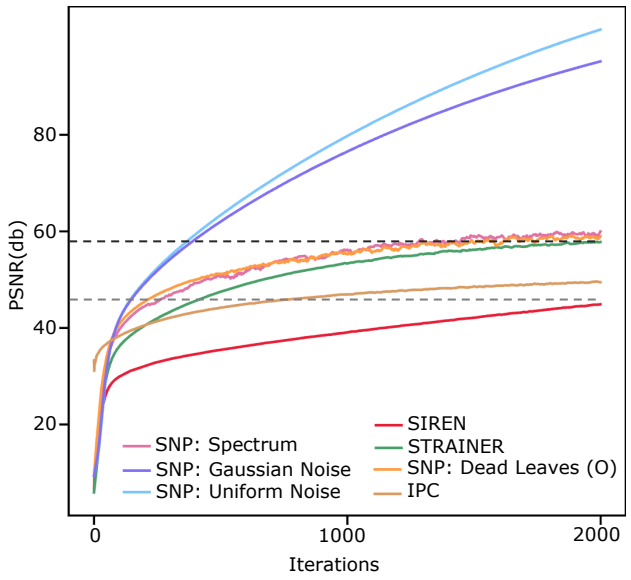

Figure 13. **Signal fitting results on OASIS-MRI images.** SNP initialized using various noises demonstrate strong signal representation ability and convergence compared to widely known baseline methods such as SIREN [39], IPC [25], and STRAINER [44]. Interestingly, SNP trained with Gaussian and Uniform noises performed significantly better than all other approaches. converging rapidly to perfect fitting.

convergence efficiency, Fig 14 plots the learning curve averaged across the dataset; notably, our SNP-initialized model attains the baseline's 100k-iteration performance in just 20k iterations. The corresponding learning curves for individual videos are detailed in Fig. 15.

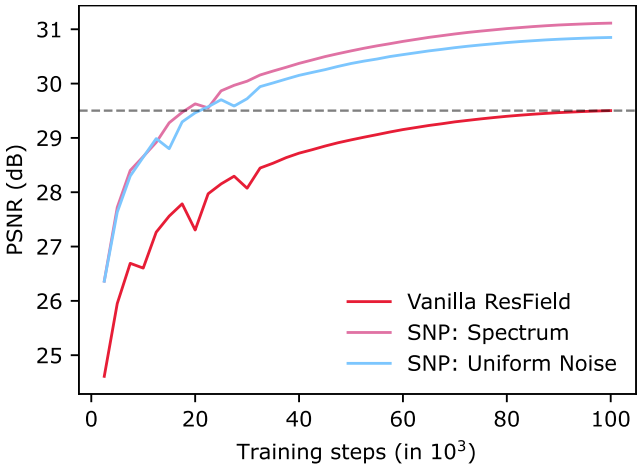

Figure 14. **Video fitting results on the all video in our dataset.**. SNP initialization accelerates the training of ResField by approximately a factor of five compared to the vanilla model across all videos.

9. Number of images used for learning SNP encoder

We study how the signal fitting performance changes based on number of training images used for pretraining SNP. We run a simple evaluation of SNP pretrained using $N = 1$ images (SNP-1) and $N = 10$ images (SNP-10) and fit to 10 samples from the CelebA-HQ dataset and plot reconstruction PSNR as a function of training iterations, shown in Fig 16. We observe that SNP pretrained on even 1 unstructured noise image (uniform and Gaussian) achieves comparable performance to SNP-10 pretrained on spectrum and dead leaves images. However, SNP-10 pretrained on 10 unstructured noise images perform significantly better in comparison, out performing SNP-1 (Uniform) by $\approx 40db+$.

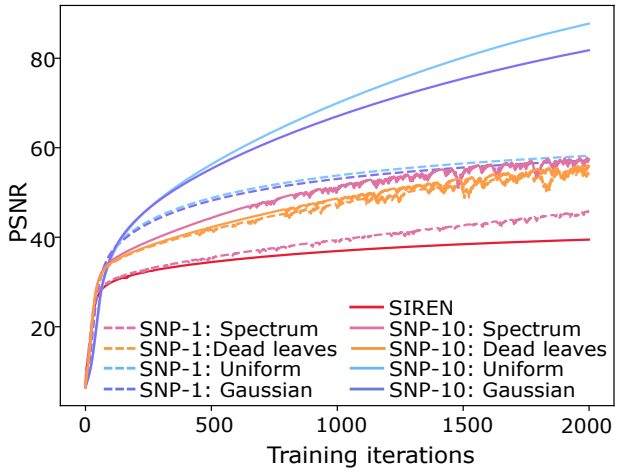

Figure 16. **Fitting performance v/s number of pretraining signals.** We show that initialization learned by SNP pretrained on multiple pretraining images, such as $N = 10$, learns a richer and powerful initialization achieving significantly better reconstruction quality than SNP pretrained on $N = 1$ signals. (SNP-k denotes INRs pretrained on k images).

10. Extending SNP to other activations (FINER)

We extend SNP to FINER [29] and demonstrate its signal fitting performance on CelebA-HQ dataset. We swap our Sine activations from SNP with the flexible FINER activations proposed by Liu et.al [29]. We pretrain SNP (FINER) on Spectrum and Uniform noise: specifically, we initialize the first-layer biases from $\mathcal{U}\left(-1/\sqrt{2}, 1/\sqrt{2}\right)$ and employ 10 noise images, otherwise adhering to our original hyperparameter setup. Table 8 reports a baseline FINER INR along with SNP (FINER) pretrained on spectrum and Uniform noises. We find SNP (FINER) pretrained on Uniform noise achieves a staggering PSNR of $100db+$ on CelebA-HQ faces. SNP trained on both, Uniform and Spectrum

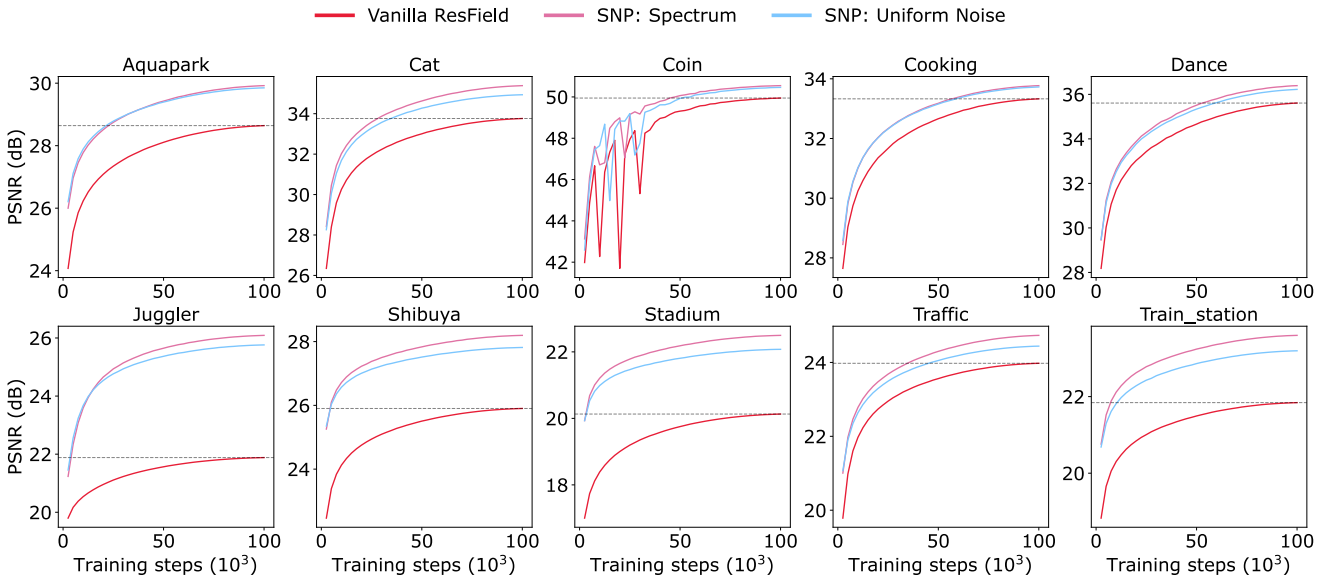

Figure 15. The SNP pretrained model provides a superior initialization, resulting in substantial starting gains and a consistently faster convergence rate during test-time fine-tuning across all videos.

Table 5. PSNR (dB) comparison across scenes for video representation and video denoising.

Task	Method	Cat	Coin	Cooking	Dance	Aquapark	Stadium	Train	Shibuya	Juggler	Traffic	All
Fitting	Vanilla Resfield	34.1	50.0	33.5	35.5	28.6	20.1	21.8	25.9	21.9	24.0	29.5
	ResField+SNP (Uniform)	35.1	50.5	33.9	36.1	29.8	22.1	23.2	27.8	25.7	24.4	30.9
	ResField+SNP (Spectrum)	35.6	50.5	33.9	36.3	29.9	22.5	23.6	28.2	26.1	24.7	31.1
Denoising	Vanilla Resfield	32.2	37.5	31.6	29.7	27.1	19.9	21.6	25.4	21.5	23.5	27.0
	ResField+SNP (Uniform)	32.8	34.3	31.6	28.5	27.9	21.8	22.8	27.2	24.8	23.9	27.6
	ResField+SNP (Spectrum)	33.3	35.4	31.8	29.1	28.0	22.2	23.2	27.5	25.1	24.1	28.0

Table 6. **Image fitting results on AFHQ.** We present mean PSNR(in db), SSIM and LPIPS with 1 std.dev. for all test methods at $T = 2000$ iterations, from vanilla randomly initialized (SIREN), to data driven methods such as Metalearned [41], STRAINER [44], and SNP variants trained on structured and unstructured noises.

Method	AFHQ		
	PSNR↑	SSIM↑	LPIPS↓
SIREN [39]	45.1 ± 3.13	0.991 ± 0.005	0.0025 ± 0.004
STRAINER [44]	58 ± 3.75	0.999 ± 0.001	0.00003 ± 0.06
Meta-learned [41]	53.3 ± 2.52	0.996 ± 0.044	0.003 ± 8.0e-5
SNP: Dead leaves oriented	55.4 ± 2.93	0.999 ± 0.0006	0.0008 ± 0.0011
SNP: Dead leaves texture	55.3 ± 3.5	0.999 ± 0.0007	0.0007 ± 0.001
SNP: Dead leaves mixed	55 ± 3.1	0.999 ± 0.0007	0.0008 ± 0.0011
SNP: Dead leaves squares	55 ± 3.3	0.999 ± 0.0008	0.0009 ± 0.0015
SNP: Spectrum	56.2 ± 3.3	0.999 ± 0.00045	0.0004 ± 0.0013
SNP: Spectrum + Color	51.1 ± 2.9	0.998 ± 0.0014	0.003 ± 0.0035
SNP: Spectrum + WMM	54.7 ± 3.5	0.999 ± 0.00088	0.0009 ± 0.0015
SNP: Spectrum + WMM + Color	55.9 ± 3.3	0.999 ± 0.0006	0.0005 ± 0.00087
SNP: Gaussian	81.8 ± 13.1	0.999 ± 4.70e-5	3.96e-6 ± 1.06e-5
SNP: Uniform	**85.91 ± 14.4**	**0.999 ± 3.30e-5**	**2.60e-6 ± 7.70e-6**

noises, outperform FINER [29] further demonstrating the effectiveness of noise pretraining and SNP's ability to readily extend to other activation functions.

Table 7. **Image fitting results on OASIS-MRI.** We present mean PSNR(in db), SSIM and LPIPS with 1 std.dev. for all test methods at $T = 2000$ iterations, from vanilla randomly initialized (SIREN), to data driven methods such as Metalearned [41], STRAINER [44], and SNP variants trained on structured and unstructured noises.

Method	OASIS-MRI		
	PSNR↑	SSIM↑	LPIPS↓
SIREN [39]	53.03 ± 1.72	0.999 ± 0.0002	0.0009 ± 0.0002
STRAINER [44]	62.8 ± 3.17	0.999 ± 0.0003	0.00001 ± 2.0e-5
Meta-learned [41]	67 ± 2.27	0.999 ± 1.0e-5	0.000005 ± 2.5e-6
SNP: Dead leaves oriented	59 ± 3.5	0.999 ± 0.0001	0.0003 ± 0.0003
SNP: Dead leaves texture	58.9 ± 3.38	0.999 ± 0.0002	0.0003 ± 0.0003
SNP: Dead leaves mixed	59.5 ± 2.76	0.999 ± 9.7e-5	0.0002 ± 0.00019
SNP: Dead leaves squares	59.1 ± 3.3	0.999 ± 0.0001	0.0002 ± 0.00026
SNP: Spectrum	60 ± 3.9	0.999 ± 0.0002	0.00017 ± 0.0004
SNP: Spectrum + Color	56.2 ± 2.69	0.999 ± 0.0002	0.00098 ± 0.0003
SNP: Spectrum + WMM	59 ± 3.3	0.999 ± 1.0e-4	0.0002 ± 0.0002
SNP: Spectrum + WMM + Color	59.8 ± 3.3	0.999 ± 9.5e-5	0.00018 ± 0.0002
SNP: Gaussian	95.2 ± 5.2	0.999 ± 1.40e-7	9.7e-8 ± 4.14e-8
SNP: Uniform	**101.8 ± 6.03**	**0.999 ± 6.30e-8**	**6.90e-8 ± 1.9e-8**

Table 8. **SNP readily extends to FINER activations [29].** We extend SNP framework to FINER [29] activations and evaluate image fitting performance on CelebA-HQ.

Method	PSNR	SSIM	LPIPS
FINER [29]	51.2 ± 3.1	0.998 ± 0.0012	0.001 ± 0.002
SNP Finer: Spectrum	52.3 ± 2.4	0.998 ± 0.001	0.003 ± 0.002
SNP Finer: Uniform	**117.2 ± 14**	**0.999 ± 2.1E−6**	**1.8E−7 ± 2.1E−6**

11. Additional baselines and deeper networks

We pretrain STRAINER using 1000 images ($100\times$ than SNP and STRAINER), each randomly selected from a different class to ensure spectral coverage, for 10^5 epochs using the best model for image fitting. Table 9 shows SNP: Spectrum outperforming STRAINER: ImageNet, indicating the crucial role of $1/f$ prior directly encoded through spectrum noise. Table 9 shows SNP also generalizes to deeper 10-layer INRs.

Table 9. **Fitting results** on additional baselines and deeper INRs.

Model	CelebA-HQ	AFHQ	OASIS-MRI
SIREN	44.9	45.1	53.03
STRAINER: ImageNet	50.87	44.11	53.37
STRAINER (dataset-specific)	57.8	58.0	62.80
SNP: **Spectrum**	56.4 ↑	56.2 ↑	60.0 ↑
SNP: **Uniform**	85.7 ↑	79.9 ↑	79.3 ↑
SIREN (10 layers)	51.3	52.6	57.3
SNP: **Spectrum (10 layers)**	64.1 ↑	64.7 ↑	66.2 ↑
SNP: **Uniform (10 layers)**	100+ ↑	100+ ↑	100+ ↑

12. Visualizing SNP initial and learned geometry

Following our local complexity (LC) analysis presented in Sec. 3 (of main text), we observe similar trends in the learned geometry of INR layers on multiple samples from the CelebA-HQ dataset as shown in Fig. 17. Similar to Fig 7 (of main text), we find that initial layers of the INR tend to reflect the features of their respective pretraining data. SIREN, which is solely fit to a given signal, exhibits LC strongly aligned with the signal morphology. We find STRAINER, which is pretrained on CelebA-HQ faces, to exhibit LC more similar to an average of all pretraining images. Furthermore, SNP spectrum , pretrained smooth noise images with $1/f^{\alpha}$ structure, reveals LC of initial layers to be smooth while later layers are seen to be quickly adapting to the signal structure. Finally, SNP Uniform, showcases a seemingly random LC in initial layers, while the deeper layer reflects extremely high non-linearity compared to other models spatially co-located with the high frequency details present in the signal.

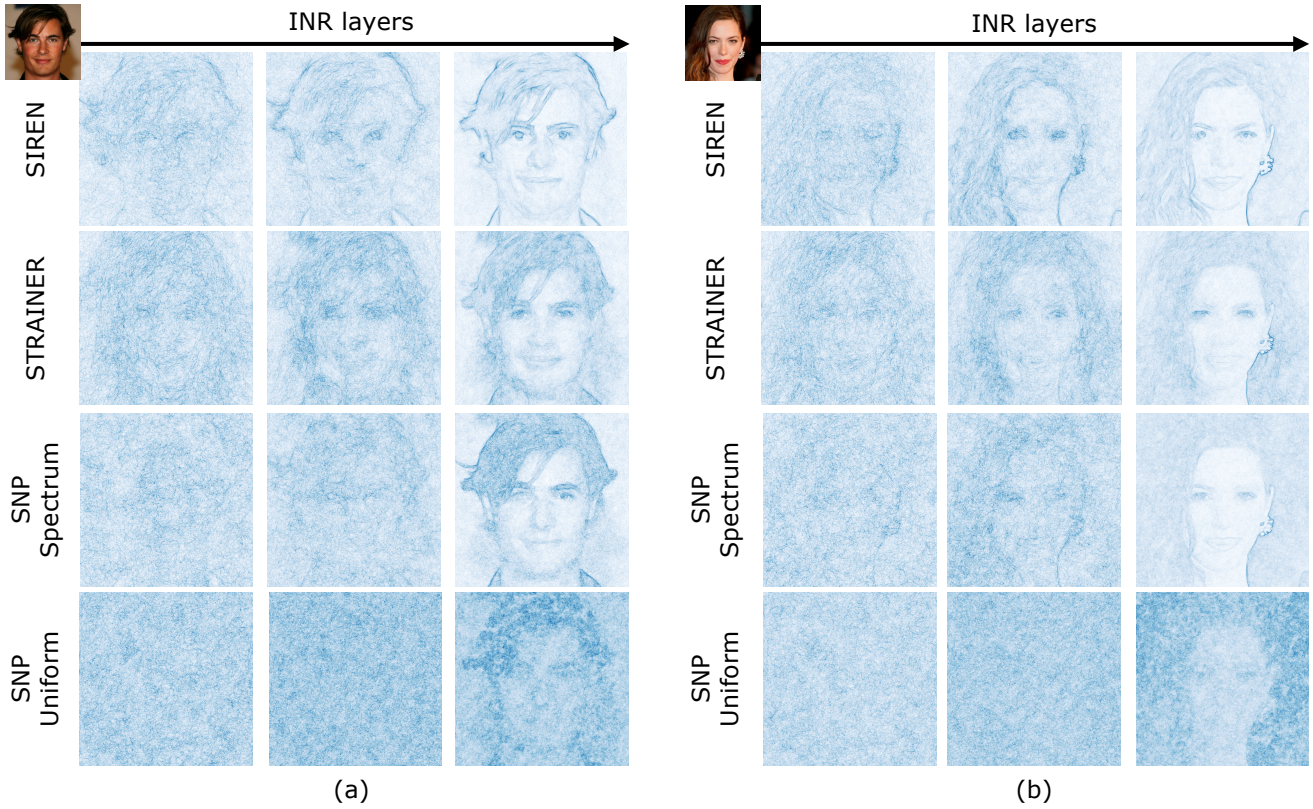

Figure 17. **More examples on local complexity of INR layers for SNP and baselines.** Using local complexity measure [18], we visualize , in (a) and (b), the local complexity of SIREN, STRAINER and SNP (uniform and spectrum) fit to images from CelebA-HQ. We observe that SIREN's internal layers quickly adapt to the underlying signal morphology. STRAINER and SNP (spectrum) exhibit internal geometry akin to their pretraining signals. Finally, we observe that SNP Uniform exhibit seemingly random geometry in the initial layers, much due to its pretraining on uniform noise.