

# SINR: Sparsity Driven Compressed Implicit Neural Representations

## Supplementary Material

### S1. Overview

In this supplementary material, we provide the pseudo-code for **SINR**, along with a detailed explanation of the activation function selection process for both the image representation and occupancy fields. Additionally, we present qualitative results for each configuration mentioned in the main paper, allowing for a clearer understanding of the model’s performance across different scenarios. Lastly, we include guidelines for selecting the value of the hyperparameter  $s$ , which plays a key role in optimizing the model’s performance. These materials are intended to complement the main text, offering further insights into the flexibility and effectiveness of **SINR**.

### S2. Pseudocode of SINR

The following algorithm provides the pseudocode for **SINR**

---

**Algorithm 1** PseudoCode of **SINR**

---

- 1: **Input:** INR, Sparsity level, Dictionary size, Random seed
  - 2: **Output:** Compressed model parameters
  - 3: *Initialization:*
  - 4:  $\mathbf{A} \leftarrow$  Sample a sensing matrix from  $\mathcal{N}(\mathbf{0}, \mathbf{I})$  using the Random seed
  - 5: **for**  $layer \in \{2, \dots, l\}$  **do**
  - 6:     **for** node in layer **do**
  - 7:          $\mathbf{w} \leftarrow$  node weights
  - 8:         minimize  $\|\mathbf{x}\|_1$  subject to  $\mathbf{w} = \mathbf{A} \mathbf{x}$
  - 9:         indices, values = **find**( $\mathbf{x} \neq 0$ )
  - 10:     **end for**
  - 11:     Store nodes’ non-zero indices and values
  - 12: **end for**
- 

### S3. Selecting the activation function

The choice of activation function plays a key role in the performance of an INR, as it is a critical factor in determining its effectiveness. Most of the literature on INR-based compression methods utilizes sinusoidal activations for signal compression. However, this may not always be the most effective activation function for all data modalities. Therefore, in this study, we examined which activation function works best for each data modality.

#### S3.1. For image representation

We randomly selected four images from the Kodak dataset to evaluate the image representation capacity of each INR.

Fig. S1 presents the results alongside the ground truth data. For this evaluation, each INR was configured with 300 hidden neurons. As shown in the results, SIREN[32] consistently outperforms all other INRs. Thus, we selected SIREN as the activation function for all image compression tasks.

#### S3.2. For occupancy fields

For the occupancy volumes presented in the main paper, specifically the Stanford Lucy and Thai statue, each occupancy field was trained using different activation functions prior to applying the compression mechanism described in **SINR**. The performance of each activation was recorded and is summarized in Tab. S1. These experiments were conducted with a hidden neuron count of 128. As shown by the results, Gaussian activation significantly outperforms sinusoidal activation for occupancy volumes. Therefore, it was selected as the default activation function for compressing occupancy volumes.

Table S1. Occupancy Field Performance Comparison

Occupancy Field	SIREN	GAUSS	WIRE
Thai Statue	0.962	0.975	0.944
Lucy	0.968	0.979	0.965

### S4. Choosing the value of $s$

In this section, we analyze the effect of  $s$  on the performance of the INR across various hidden sizes. We vary the hidden neuron size as 32, 64, 96, 128, 192 and 256. Hidden neuron size of 32 investigates the effect of  $s$  for a ‘tiny INR’ (Sec. 3.3.2). From Fig. S2 (a), we observe that for  $s > 350$ , the performance of the compressed INR closely matches that of the uncompressed INR (denoted by the dotted red line). A similar pattern emerges for other hidden sizes (Fig. S2 (b)-(f)), where performance shows only marginal improvement beyond a certain threshold of  $s$ . This marginal increase typically occurs at an optimal value of  $s$  where **SINR** achieves sufficient compression. Additionally, we also present a plot in Fig. S2 (g), which describes the variation of the optimal value of  $s$  with the number of hidden neurons. This plot enables the selection of the optimal  $s$  value for a given hidden size without needing to fit the INR across different values of  $s$ . Notably, the variation of the optimal  $s$  with hidden size is nearly linear.



Figure S1. Image representation performances for different activation functions

## S5. Neural radiance fields encoding

Neural Radiance Fields (NeRF) can be considered a novel view generator when it is trained with a sufficient number of training views, along with their corresponding positions and directions. Fundamentally, once trained, a NeRF is an INR. Therefore, the information encoded in its weights for generating novel views can be compressed into a dictionary. Figure S3 presents the results obtained with the proposed **SINR**. As shown, **SINR** achieves more than 50% compression while maintaining the same PSNR. These results further confirm the applicability of **SINR** for compressing INRs across different data modalities. We used the ReLU+PE activation for encoding NeRFs.

## S6. Additional qualitative and quantitative results

In this section, we present the decoded results of **SINR** and the baseline models for the network configurations  $C_1$ ,  $C_3$ , and  $C_4$ . Additionally, we provide details on the network depth and the number of hidden neurons for each configuration. Fig. S4 presents the results of image compression for INRs using two hidden layers with 32 neurons. Further, Figs. S5 and S6 showcase results for image compression with INRs using three hidden layers with 64 and 128 neurons, respectively. Finally, Figs. S7 and S8 display results for image compression with INRs using three hidden layers

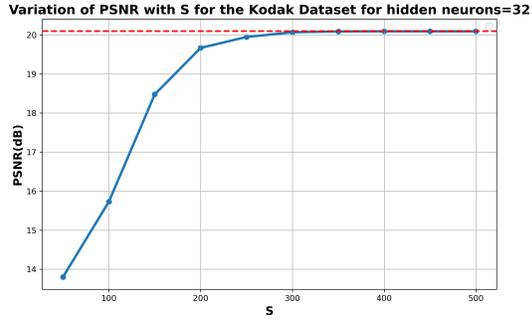
with 96 and 128 neurons, respectively.

## S7. Gigapixel image compression

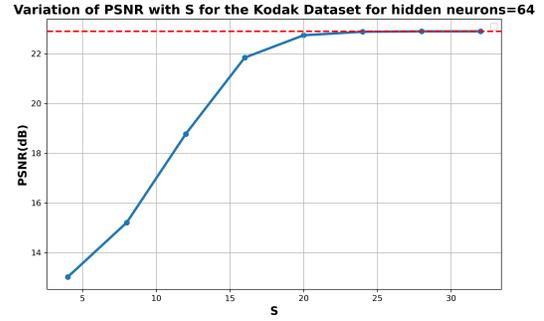
In this section, we demonstrate that SINR can be used to compress INRs for the task of gigapixel image compression. Gigapixel image compression focuses on efficiently encoding ultra-high-resolution images while preserving the visual quality. We incorporated SINR into the state-of-the-art gigapixel compression method, SHACIRA [11], and found that it achieves a compression ratio of 66% with only a 0.3 dB drop in PSNR. This experiment was run on the image in [40].

## S8. Comparison with low-rank factorization

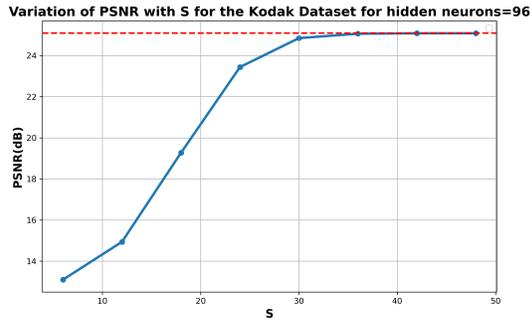
We performed low-rank factorization on  $W$  by using SVD. We then express  $W = AB$  where  $A = U\Sigma^k$  and  $B = V^T$  with  $\Sigma^k$  denoting the matrix formed by selecting the top  $k$  singular values from  $\Sigma$ . We utilized an INR with 64 hidden neurons. In order to have the same number of parameters after compression as SINR, we chose  $k = 24$  for the low-rank factorization. The obtained average results on the Kodak dataset is as follows. With low rank approximation: 11.62 dB, with SINR: 28.97 dB.



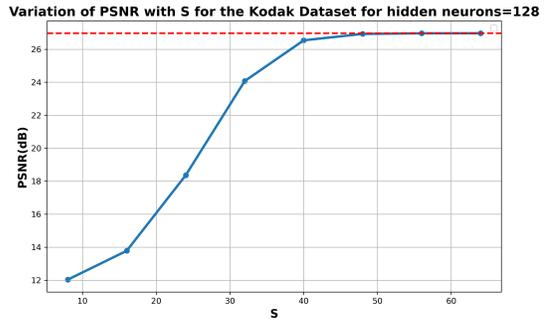
(a)



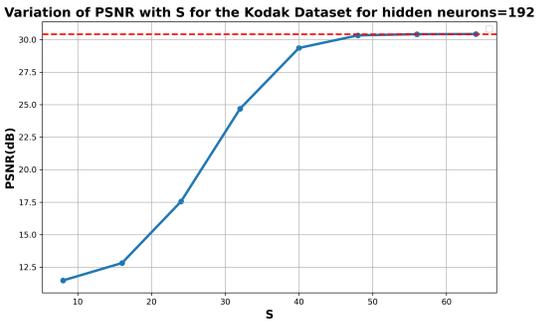
(b)



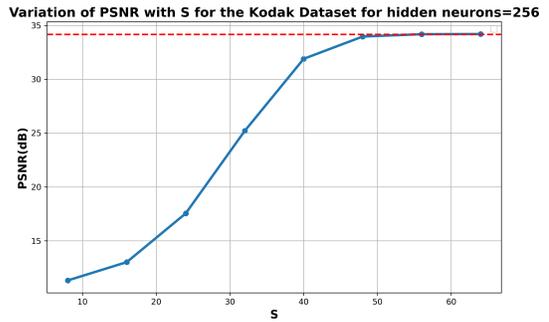
(c)



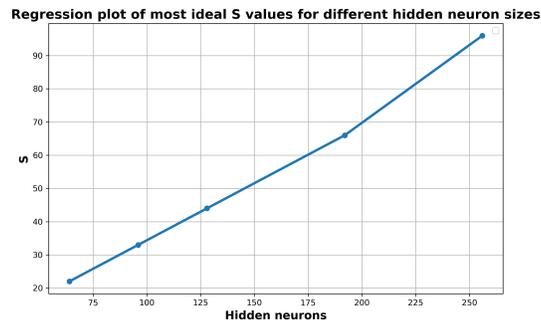
(d)



(e)



(f)



(g)

Figure S2. Variation of PSNR (dB) with the sparsity level for different number of hidden neurons. The red dotted line indicates uncompressed INR performance. The regression plot shows that the variation of the optimal value of  $s$  with hidden neuron size is nearly linear.

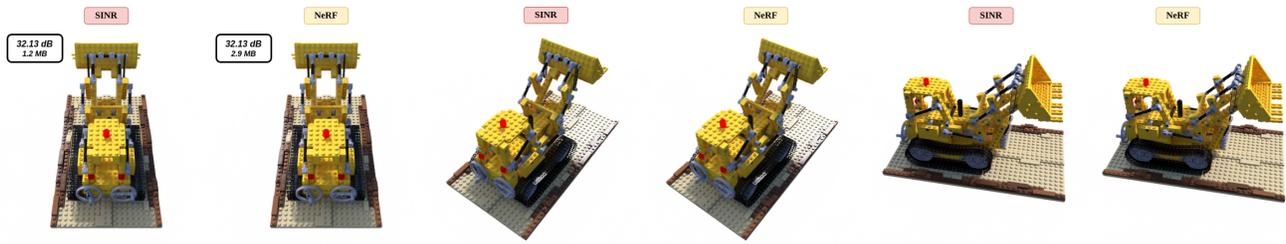


Figure S3. **Results for NeRF compression:** **SINR** compresses the radiance field without any loss in PSNR while significantly reducing storage requirements.

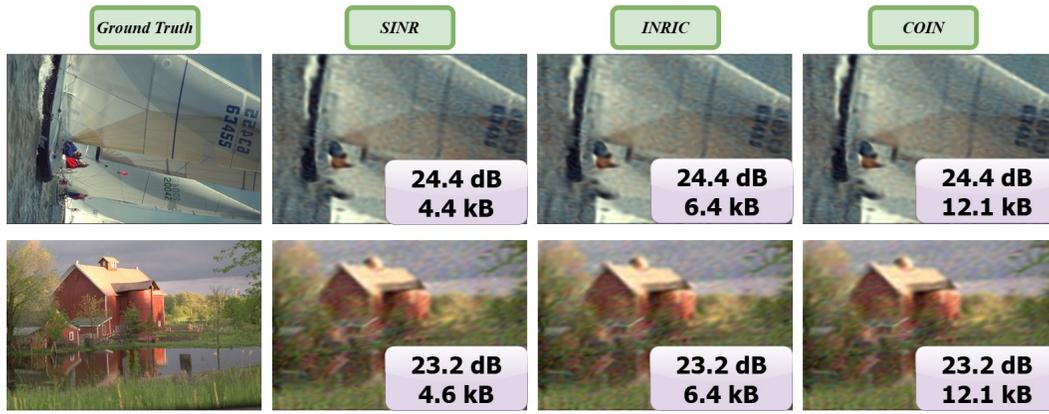


Figure S4. **Image compression performance for configuration  $C_1$**  when there are two hidden layers, and 32 neurons: As can be seen from the results, **SINR** obtains significant compression compared to competing methods.

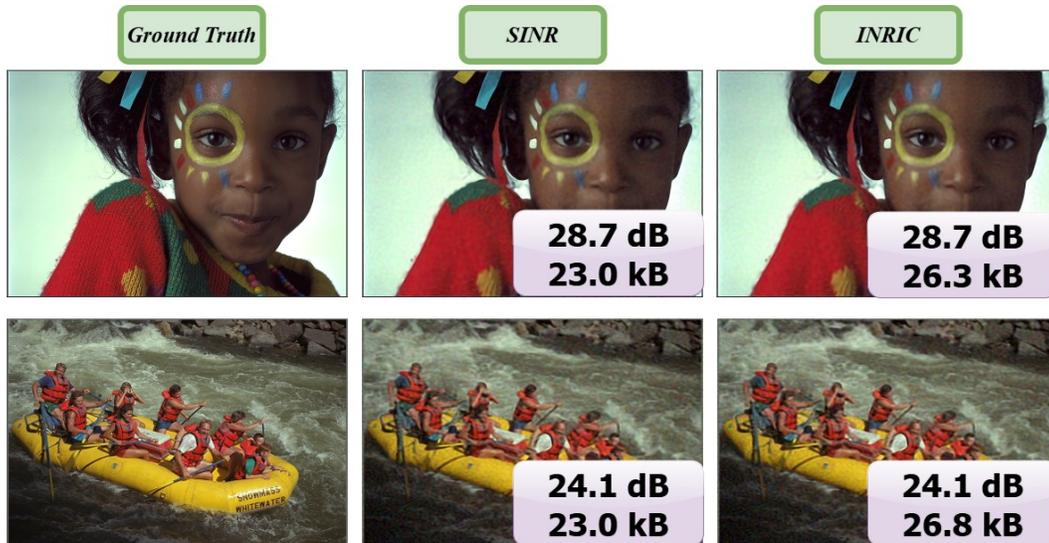


Figure S5. **Image compression performance for configuration  $C_3$**  when there are three hidden layers, and 64 neurons: As can be seen from the results, **SINR** obtains significant compression while preserving the same quantitative metrics compared to competing methods.



Figure S6. Image compression performance for configuration  $C_3$  when there are three hidden layers, and 128 neurons: The results indicate that **SINR** achieves notable compression without compromising the quantitative metrics when compared to the competing methods.

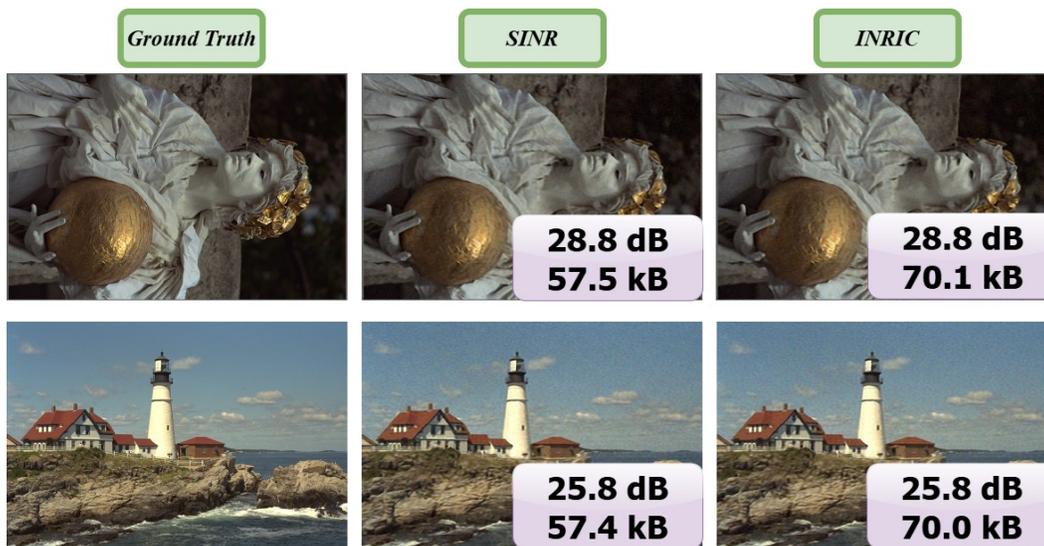


Figure S7. Image compression performance for configuration  $C_4$  when there are three hidden layers, and 96 neurons: The results show that **SINR** delivers significant compression without compromising quantitative measurements when compared to the competing methods.

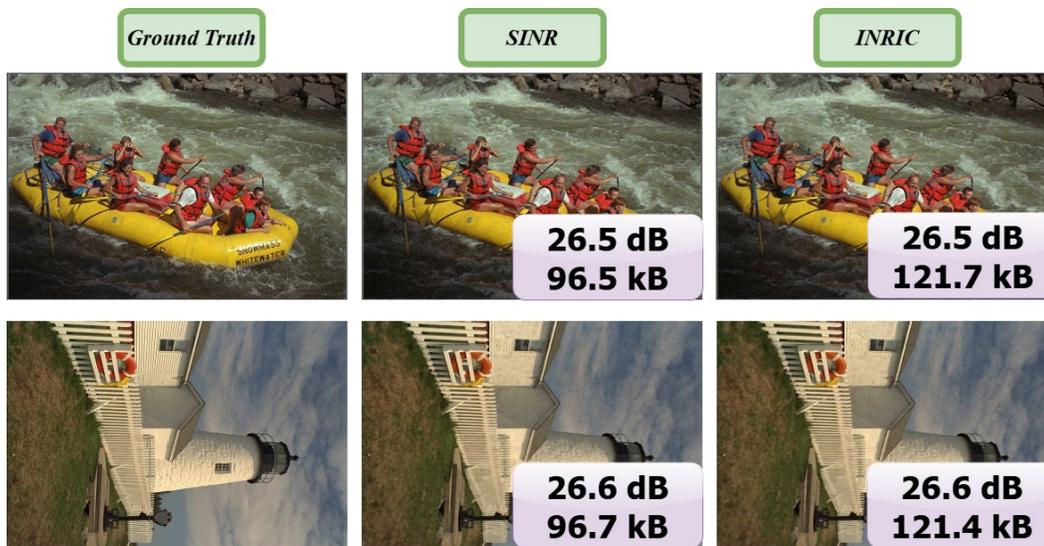


Figure S8. **Image compression performance for configuration  $C_4$  when there are three hidden layers, and 128 neurons:** The results reveal that **SINR** provides significant compression without compromising quantitative measurements when compared to the competing methods.