Towards Lossless Implicit Neural Representation via Bit Plane Decomposition

Supplementary Material

1. Theory

The specific theorems presented in [16] are as follows:

Theorem. *(The implicit ANN approximations with described error tolerance and explicit parameter bounds by Jentzen et al. [16], Proposition 4.3.8, Corollary 4.3.9) Let $d \in \mathbb{N}$, $L, a \in \mathbb{R}$, $b \in [a, \infty)$, $\epsilon \in (0, 1]$ and function f satisfy for $\forall x, y \in [a, b]^d$ that $|f(x) - f(y)| \leq L\|x - y\|_1$,*
Then there exist f_θ that satisfy :

1. *It holds the Upper Bound on Network Error:* $\sup ||f_\theta(\mathbf{x}) - f(\mathbf{x})||_1 \leq \epsilon$
2. *Upper Bound on Number of Layers:* $d(\log_2(\max\{\frac{3dL(b-a)}{2}, 1\}) + \log_2(\epsilon^{-1})) + 2$
3. *Upper Bound on Number of Channels of each layer:* $\epsilon^{-d} d(3\frac{(3dL(b-a))^d}{2^i} + 1)$
4. *Upper Bound on Network Parameters:* $\epsilon^{-2d} 9(3d \max\{L(b-a), 1\})^{2d} d^2$

In this section, we provide a brief proof of the theory. We also demonstrate $\mathfrak{C}$ in our setup. Note that Tab. 1 shows all notations for the main paper and Tab. 2 shows notations for supplementary material.

1.1. Proof of Sec. 1

According to Jentzen et al. [16], the proof of the theorem is derived as follows. The proof proceeds by designing a function that satisfies the Proposition 1, substituting the L1 distance and maximum value approximated by the ANN, and then generalizing a distance to an arbitrary number. For brevity, we provide a summarized outline of the proof. For a rigorous mathematical proof, please refer to the original document.

Proposition 1. *Let (E, δ) be a metric space and $L \in [0, \infty)$, $\varnothing \neq \mathcal{M} \subseteq E$, and $f : E \to \mathbb{R}$ that satisfy $\forall x \in E, y \in \mathcal{M}$ s.t. $|f(x) - f(y)| \leq L\delta(x, y)$. Let $F : E \to \mathbb{R} \cup \{\infty\}$ for all $x \in E$ that*

$$F(x) = \sup_{y \in \mathcal{M}} [f(y) - L\delta(x, y)].$$

Then, it holds $\forall x \in E$ that

$$|F(x) - f(x)| \leq 2L[\inf_{y \in \mathcal{M}} \delta(x, y)].$$

Let the notation $\mathbf{A}_{\mathcal{W},\mathbf{b}}$ indicates an affine transforms with weight ($\mathcal{W}$) and bias ($\mathbf{b}$), and $\mathbb{T}_{d,K}$ indicates an ANN that satisfies $\mathbb{T}_{d,K}(\mathbf{x}) = \underbrace{[\mathbf{x}^T, \mathbf{x}^T, \cdots \mathbf{x}^T]^T}_{K}$ with $d, K \in \mathbb{N}$. With ReLU activation, L1 distance is represented by 2-layer MLP as below:

Definition 1. *Let weights ($\mathcal{W}^{(1,2)}$) and bias ($\mathbf{b}^{(1,2)}$) of the affine transform be as follows:*

$$\mathcal{W}^{(1)} := \begin{bmatrix} 1 \\ -1 \end{bmatrix} \mathbf{b}^{(1)} := \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \mathcal{W}^{(2)} := \begin{bmatrix} 1 & 1 \end{bmatrix} \mathbf{b}^{(2)} := \begin{bmatrix} 0 \end{bmatrix}. \tag{2}$$

$\forall x \in \mathbb{R}$, the ANN $\mathbb{L}_1(x) = |x|$ is defined as below:

$$\mathbb{L}_1(\mathbf{x}) := \mathbf{A}_{\mathcal{W}^{(2)},\mathbf{b}^{(2)}}(\sigma(\mathbf{A}_{\mathcal{W}^{(1)},\mathbf{b}^{(1)}}(\mathbf{x}))), \tag{3}$$

$$\text{where, } \sigma(x) := \max(x, 0). \tag{4}$$

Then, $\forall \mathbf{x} \in \mathbb{R}^d$ and $d \in \mathbb{N}$, the ANN $\mathbb{L}_d(x) = ||\mathbf{x}||_1$ is defined as below:

$$\mathbb{L}_d(\mathbf{x}) := \mathbf{A}_{\mathcal{W}_d^{(2)},\mathbf{b}_d^{(2)}}(\sigma(\mathbf{A}_{\mathcal{W}_d^{(1)},\mathbf{b}_d^{(1)}}(\mathbf{x}))), \tag{5}$$

where $\mathcal{W}_d^{(1)} \in \mathbb{R}^{2d \times d}, \mathcal{W}_d^{(2)} \in \mathbb{R}^{1 \times 2d}, \mathbf{b}_d^{(1)}$, and $\mathbf{b}_d^{(2)}$ are as below:

$$\mathcal{W}_d^{(1)} := \mathbf{E}_d \otimes \mathcal{W}^{(1)} = \begin{bmatrix} \mathcal{W}^{(1)} & 0 & \cdots & 0 \\ 0 & \mathcal{W}^{(1)} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathcal{W}^{(1)} \end{bmatrix}, \tag{6}$$

$$\mathcal{W}_d^{(1)} := \underbrace{[1, 1, \cdots, 1]}_{2d} \quad \mathbf{b}_d^{(1)} = \vec{0} \in \mathbb{R}^{2d} \quad \mathbf{b}_d^{(2)} = [0] \tag{7}$$

where $\mathbf{E}$ is an identity matrix and $\otimes$ is Kronecker product.

We denote $\mathbf{P}_d(\cdot, \cdot, ...)$ as d-parallel of ANNs and $\bullet$ as sequential of ANNs. Likewise, the definition of the maximum value is defined as below:

Definition 2. *Let weights ($\mathcal{W}^{(1,2)}$) and bias ($\mathbf{b}^{(1,2)}$) of the affine transform be as follows:*

$$\mathcal{W}^{(1)} := \begin{bmatrix} 1 & -1 \\ 0 & 1 \\ 0 & -1 \end{bmatrix} \mathbf{b}^{(1)} := \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \tag{8}$$

$$\mathcal{W}^{(2)} := \begin{bmatrix} 1 & 1 & -1 \end{bmatrix} \mathbf{b}^{(2)} := \begin{bmatrix} 0 \end{bmatrix}. \tag{9}$$

$\forall \mathbf{x} = [x_1, x_2]^T \in \mathbb{R}^2$, the ANN $\mathbb{M}_2(x) = \max\{x_1, x_2\}$ is defined as below:

$$\mathbb{M}_2(\mathbf{x}) := \mathbf{A}_{\mathcal{W}^{(2)},\mathbf{b}^{(2)}}(\sigma(\mathbf{A}_{\mathcal{W}^{(1)},\mathbf{b}^{(1)}}(\mathbf{x}))), \tag{10}$$

Then, $\mathbb{M}_d$ for $d \in \mathbb{N} \cap [3, \infty]$ is defined as follow:

Symbol	Definition	Description	Example/Meaning/Note
d	$\in \mathbb{N}$	Dimension of Function (Signal) or Vector	$d = 2$ for an Image
n	$\in \mathbb{N}$	Bit Precision a Ground Truth Function	$n = 8$ for an 8-bit (uint 8) Image
k	$\in \text{div}^+(n)$	Bit Precision a Represented Function	Control Variable in Tab. 3
i	$\in \mathbb{N} \cap (0, \frac{n}{k}]$	Index of a Quantized Function	$i = \{0, 1, 2, 3\}$ in case of $n = 8$ and $k = 2$
H, W, C	$\in \mathbb{N}$	Height, Width, Channels of Function	
L	$\in \mathbb{R}$	Lipschitz Constant	Details in Sec. 1
ϵ	$\in \mathbb{R}$	Error (or distance)	Quantization Error in our paper
$\mathbf{x}$	$\in (\mathcal{X} \subseteq \mathbb{R}^d)$	Input Vector of a Function	
$\mathbf{I}$	$\in \mathbb{R}^{H \times W \times C}$	Analog C-channel Image	
$\mathbf{I}_n, \mathbf{Q}_k$	$\in Q_n^{H \times W \times C}, Q_k^{H \times W \times C}$	Digital n-bit (or k-bit) C-channel Image	$n = 8, 16$ for images
$\mathbf{B}$	$\in \{0, 1\}^{H \times W \times C}$	Bit-plane of an image	In case of $\mathbf{Q}_{k=1}$
$\mathcal{W}$	$\in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$	Weight Matrix (Trainable Parameters)	`nn.Linear.weight` (Pytorch)
$\mathbf{b}$	$\in \mathbb{R}^{d_{\text{out}}}$	Bias Vector (Trainable Parameters)	`nn.Linear.bias` (Pytorch)
θ	$\in \{\bigtimes_{i=0}^{M-1} (\mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}, \mathbb{R}^{d_{\text{out}}})\} (:= \Theta)$	Trainable Parameters of M-Layer MLP and its Set	
β	$:= \frac{1}{d_{\text{in}} d_{\text{out}}} \|\|\mathcal{W}\|\|_1$	Scaling Factor for a Weight Quantization	
γ	$:= \|\|x\|\|_\infty$	Scaling Factor for an Activation Quantization	
$\mathfrak{C}$	$:= 9 \cdot (3d \max\{L(b-a), 1\})^{2d} \cdot d^2$	Coefficient of $\mathcal{U}_d(n)$	Details in Sec. 1
$[a, b]$	$:= \{x \in \mathbb{R} \| a \leq x \leq b\}$	Domain of a function	a, b satisfy $(a, b \in \mathbb{R}, a \leq b)$
Q_n	$\subset \mathbb{Q} \cap [0, 1]$	Codomain (or Range) of a Digital Function	$Q_n = \{0, \frac{1}{3}, \frac{2}{3}, 1\}$ in case of $n = 2$
$[\mathbf{Q}_k^{(i)}]_{i=0}^j$	$\bigtimes_i^j Q_k^{H \times W \times C}$	Sequence a Quantized Function	LSBs to MSBs as i increased
$[\mathbf{B}_k^{(i)}]_{i=0}^j$	$\bigtimes_i^j \{0, 1\}_k^{H \times W \times C}$	Sequence a Bit-Plane	LSB to MSB as i increased
f, h	$\mathbb{R}^d \to \mathbb{R}^k$	Analog function	$k = 1$ in Sec. 3.1 of main paper.
f_n, h_n	$\mathbb{R}^d \to Q_n^k$	Digital function with n-bit precision	
f_θ, h_θ	$\mathbb{R}^d \to \mathbb{R}^k$	Function that parameterized with θ	Implicit Neural Representation (INR)
$\mathcal{Q}_n(\cdot)$	$\mathbb{R}^d \to Q_n^d, \quad \hat{x} \mapsto \arg\min_{x \in Q_n} \|\|x - \hat{x}\|\|_1$	n-bit Quantization	Element-wise operation for vector inputs
$\mathcal{P}(\cdot)$	$\Theta \to \mathbb{N}$	Number of Parameters of a Neural Network	
$\epsilon(\cdot)$	$\mathbb{N} \to \mathbb{R}, \quad n \mapsto \frac{1}{2(2^n - 1)}$	Upper bound of a quantization error with given n	
$\mathcal{U}_d(\cdot)$	$\mathbb{N} \to \mathbb{N} \quad n \mapsto \mathfrak{C}(2^{n+1} - 2)^{2d}$	Upper bound of a $\mathcal{P}$ with given n and d	
$\widehat{(\cdot)}$	$\simeq (\cdot)$	Prediction to $(\cdot)$	Applied to elements or functions

Table 1. Notation table for the main paper (Elements, Sets, and Functions (calculations), respectively)

$$\mathbb{M}_d = \begin{cases} \mathbb{M}_k \bullet \mathbf{P}_k(\mathbb{M}_2, \mathbb{M}_2, \cdots, \mathbb{M}_2) & d = 2k \\ \mathbb{M}_k \bullet \mathbf{P}_k(\mathbb{M}_2, \mathbb{M}_2, \cdots, \mathbb{M}_2, \mathbf{E}_1) & d = 2k - 1 \end{cases} \tag{11}$$

$\mathbb{M}_d$ satisfy $\forall \mathbf{x} = [x_1, x_2 \cdots, x_d]^T \in \mathbb{R}^d$, $\mathbb{M}_d(\mathbf{x}) = \max\{x_1, x_2, \cdots, x_d\}$

Then maximum convolution is represented with an ANN $\mathbf{\Phi}(\cdot)$ as follow:

Proposition 2. *Let $d, K \in \mathbb{N}$, $L \in [0, \infty)$, $\mathbf{x}_k \in \mathbb{R}^d$, and $\mathbf{y} = [y_1, y_2, \cdots y_K] \in \mathbb{R}^K$. Then the ANN $\mathbf{\Phi}$ defined as below:*

$$\mathbf{\Phi} = \mathbb{M}_K \bullet \mathbf{A}_{-L \cdot \mathbf{E}_K, \mathbf{y}} \bullet \mathbf{P}_K(\mathbb{L}_d \bullet \mathbf{A}_{\mathbf{E}_d, -\mathbf{x}_1}, \mathbb{L}_d \bullet \mathbf{A}_{\mathbf{E}_d, -\mathbf{x}_2}, \cdots \mathbb{L}_d \bullet \mathbf{A}_{\mathbf{E}_d, -\mathbf{x}_K}) \bullet \mathbb{T}_{d,K}.$$

$\mathbf{\Phi}$ holds $\forall \mathbf{x} \in \mathbb{R}^d$,

$$\mathbf{\Phi}(\mathbf{x}) = \max_{k \in \{1, 2, \cdots K\}} (y_k - L||\mathbf{x} - \mathbf{x}_k||_1). \tag{12}$$

Then with Propositions 1 and 2, the ANN approximation follows:

Proposition 3. *Let $d, K \in \mathbb{N}$, $L \in [0, \infty)$, $\mathbf{x}_k \in E \subseteq \mathbb{R}^d$. Let $f : E \to \mathbb{R}$ satisfies $\forall \mathbf{x}_{1,2} \in E$, $|f(\mathbf{x}_1) - f(\mathbf{x})| \leq L||\mathbf{x}_1 - \mathbf{x}_2||_1$. Let $\mathbf{y} = [f(\mathbf{x}_1), f(\mathbf{x}_2), \cdots, f(\mathbf{x}_K)]^T$ and $\mathbf{\Phi}$ is defined as Proposition 2. Then,*

$$\sup_{\mathbf{x} \in E} |\mathbf{\Phi}(\mathbf{x}) - f(\mathbf{x})| \leq 2L[\sup_{\mathbf{x} \in E} (\min_k ||\mathbf{x} - \mathbf{x}_k||_1)] \tag{13}$$

The proof of the Sec. 1.1 accomplished by substitution of Proposition 1 to Proposition 2. Generalizing Eq. (13) complete the proof. Let $\mathcal{C}^{(E,\delta),r}$ is r-covering number of (E, δ). Then,

$$\mathcal{C}^{([a,b]^d, ||\cdot||_p), r} \leq (\lceil \frac{d^{1/p}(b-a)}{2r} \rceil)^d \leq \begin{cases} (\frac{d(b-a)}{r})^d & (r < \frac{d(b-a)}{2}) \\ 1 & (r \geq \frac{d(b-a)}{2}) \end{cases} \tag{14}$$

Lemma 1. *Let $d, K \in \mathbb{N}$, $L \in [0, \infty)$, $a \in \mathbb{R}, b \in (a, \infty)$, $f : [a, b]^d \to \mathbb{R}$ satisfies $\forall \mathbf{x}_{1,2} \in [a, b]^d$, $|f(\mathbf{x}_1) - f(\mathbf{x})| \leq L||\mathbf{x}_1 - \mathbf{x}_2||_1$. And let $\mathbf{F} = \mathbf{A}_{0, f([(a+b)/2]^d)}$ Then,*

$$\sup_{\mathbf{x} \in [a,b]^d} |\mathbf{F}(\mathbf{x}) - f(\mathbf{x})| \leq \frac{dL(b-a)}{2}. \tag{15}$$

The inequality is derived by substituting $\mathbf{x}_1 = [(a+b)/2, (a+b)/2, \cdots, (a+b)/2]^T$ in $|f(\mathbf{x}_1) - f(\mathbf{x})| \leq L||\mathbf{x}_1 - \mathbf{x}_2||_1$.

Symbol	Definition	Description	Example/Meaning/Note
m	$\in \mathbb{Q}_{24}$	Mantissa	$x = m \times 2^e$
e	$\in \mathbb{Q}_8$	Exponent	
$\mathbf{m}$	$\in \mathbb{Q}_{24}^L$	Mantissa Tensor with L length	
$\mathbf{e}$	$\in \mathbb{Q}_8^L$	Exponent Tensor with L length	
$\mathbf{O}$	$\in \mathbb{R}^L$	Floating point audio signal with L length	$\mathbf{O} = \mathbf{m} \times 2^{\mathbf{e}}$
$\mathbf{E}_d$	$\in \mathbb{R}^{d \times d}$	Identity matrix	$\mathbf{E}_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$
$\mathcal{W}_d$	$\mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$	Weights for ANNs	Used for Definition 1 and Definition 2
$\mathbf{b}_d$	$\mathbb{R}^{d_{\text{out}}}$	Bias for ANNs	Used for Definition 1 and Definition 2
$\mathcal{C}^{(E,\delta),r}$	$\min(\{n \in \mathbb{N}_0 : [\exists A \subset E : ((\lvert A\rvert \leq n) \wedge (\forall x \in E : \exists a \in A : \delta(a,x) \leq r))]\} \cup \{\infty\})$	Covering numbers	r-convering number of (E, δ)
(E, δ)	-	Metric Space	Set E and its metric δ
$\mathcal{M}$	$\subseteq E$	Subset of E	
$\mathbf{N}$	$\forall \mathbf{\Phi}$	A set of ANNs	
δ	$E \times E \to [0, \infty)$	Metric on E	Satisfy positive definiteness, symmetry, and triangle inequality
$\mathbf{A}_{\mathcal{W},\mathbf{b}}$	$\mathbb{R}^{d_{\text{in}}} \to \mathbb{R}^{d_{\text{out}}}, \mathbf{x} \mapsto \mathcal{W}\mathbf{x} + \mathbf{b}$	Affine transform	
$\sigma(\cdot)$	$\mathbb{R}^d \to \mathbb{R}^d, \quad \mathbf{x} \mapsto \max\{\mathbf{x}, 0\}$	ReLU Activation function	Applied for each elements of $\mathbf{x}$
$\mathbf{\Phi}$	$\mathbb{R}^{\text{in}} \to \mathbb{R}^{\text{out}}$	Artificial Neural Networks	In Sec. 1, range is constrained to $\mathbb{R}$
$\mathbb{L}_d$	$\mathbb{R}^d \to \mathbb{R}$	$\lvert\lvert\mathbf{x}\rvert\rvert_1$ representation with ANN	(Definition 1) 2-layer
$\mathbb{M}_d$	$\mathbb{R}^d \to \mathbb{R}$	$\max\{\mathbf{x}\}$ representation with ANN	(Definition 2) An unique
$\mathbb{T}_{d,K}$	$\mathbb{R}^d \to \mathbb{R}^{Kd}, \mathbf{x} \mapsto [\mathbf{x}^T, \mathbf{x}^T, \cdots \mathbf{x}^T]^T$	K-times repetition of $\mathbf{x}$ with ANN	
$\mathbf{P}(\cdot, ..., \cdot)$	$\bigtimes_{i=0}^{K-1} \mathbf{N} \to \mathbf{N}$	Parallel of K ANNs	
$(\cdot) \bullet (\cdot)$	$\mathbf{N} \times \mathbf{N} \to \mathbf{N}$	Sequence of ANNs	$\mathbf{\Phi}_2 \bullet \mathbf{\Phi}_1(\mathbf{x}) = \mathbf{\Phi}_2(\mathbf{\Phi}_1(\mathbf{x}))$
$(\cdot)_{n=k}$	-	Functions with k-bit precision	$f_{\theta,n=2}$ indicates an INR with 2-bit precision
$(\cdot)^*$		INRs satisfy Eq. (5)	$f^*_{\theta,n=4}$ indicates lossless INR with 4-bit precision

Table 2. Notation table for the supplement material (Elements, Sets, and Functions (calculations), respectively)

Proposition 4. *Let $d \in \mathbb{N}$, $L \in [0, \infty)$, $a \in \mathbb{R}, b \in (a, \infty), r \in (0, d/4)$, $f : [a, b]^d \to \mathbb{R}$ satisfies $\forall \mathbf{x}_{1,2} \in [a, b]^d$, $|f(\mathbf{x}_1) - f(\mathbf{x})| \leq L||\mathbf{x}_1 - \mathbf{x}_2||_1$. Let $\mathbf{x}_k \in \mathbb{R}^d$, and $\mathbf{y} = [y_1, y_2, \cdots y_K] \in \mathbb{R}^K$ and let K satisfy $K = \mathcal{C}^{([a,b],||\cdot||_1),(b-a)r}, \sup_{\mathbf{x}}[\min_k ||\mathbf{x} - \mathbf{x}_k||_1] \leq (b-a)r$ and $\mathbf{y} = [f(\mathbf{x}_1), f(\mathbf{x}_2), \cdots, f(\mathbf{x}_K)]^T$ and $\mathbf{\Phi}$ is defined as Eq. (12). Then it holds*

$$\sup_{\mathbf{x}} |\mathbf{\Phi}(\mathbf{x}) - f(\mathbf{x})| \leq 2L(b-a)r \tag{16}$$

This is derived by Eq. (13) and assumption.

Then generalizing the function with the proposition as follows:

Proposition 5. *Let $d \in \mathbb{N}$, $L \in [0, \infty)$, $a \in \mathbb{R}, b \in (a, \infty), r \in (0, \infty)$, $f : [a, b]^d \to \mathbb{R}$ satisfies $\forall \mathbf{x}_{1,2} \in [a, b]^d$, $|f(\mathbf{x}_1) - f(\mathbf{x})| \leq L||\mathbf{x}_1 - \mathbf{x}_2||_1$. Then there exists an ANN $\mathbf{\Phi}$ s.t.*

$$\sup_{\mathbf{x}} |\mathbf{\Phi}(\mathbf{x}) - f(\mathbf{x})| \leq 2L(b-a)r \tag{17}$$

The definition of covering number and $K = \mathcal{C}^{([a,b],||\cdot||_1),(b-a)r} < \infty$ ensure that there exist $\mathbf{x}_k \in [a, b]^d$ s.t.

$$\sup_{\mathbf{x}}[\min_k ||\mathbf{x} - \mathbf{x}_k||_1] \leq (b-a)r \tag{18}$$

Without loss of generality, $L(b-a) \neq 0$ the main theorem is thus complete by Proposition 5 by adjusting r. In conclusion, reducing the bit-precision of a digital signal is equivalent to increasing r, i.e., reducing $\mathcal{C}$. The number of layers and parameters is then derived by calculating the number of parameters in Definitions 1 and 2.

1.2. Hyperparameter

The most important factor that determines $\mathfrak{C}$ is the Lipschitz constant L. The constant L represents how 'smooth' the signal is in a discrete setting. In discrete spaces, computing L is known to be an NP-hard problem. However, it can be estimated under various assumptions. Specifically, since L satisfies the inequality below, where 1) $x, y \in [a, b]^d$ are fixed-size discrete domains, and 2) $f(x) \in [0, 1]$, it is possible to estimate its upper bound.

Therefore, the term $\mathfrak{C}$ in the main text is given by

$$\mathfrak{C} = 9 \cdot (3d \max\{L(b-a), 1\})^{2d} \cdot d^2, \tag{19}$$

where a and b are the same as in previous studies, i.e., -1 and 1, d varies depending on the shape of the signal (Audio, Image or Video, etc.). The Lipschitz constant L changes according to the domain size and the signal derivative. For the 256×256 images used in the experiments, with a range of [0,1], the Lipschitz constant must satisfy $256 \leq L$ for all arbitrary signals.

2. Quantized Representation

Details for hypothesis Validation In this section, we provide a detailed schematic diagram of **Validation** of the experiment section to avoid confusion and provide additional

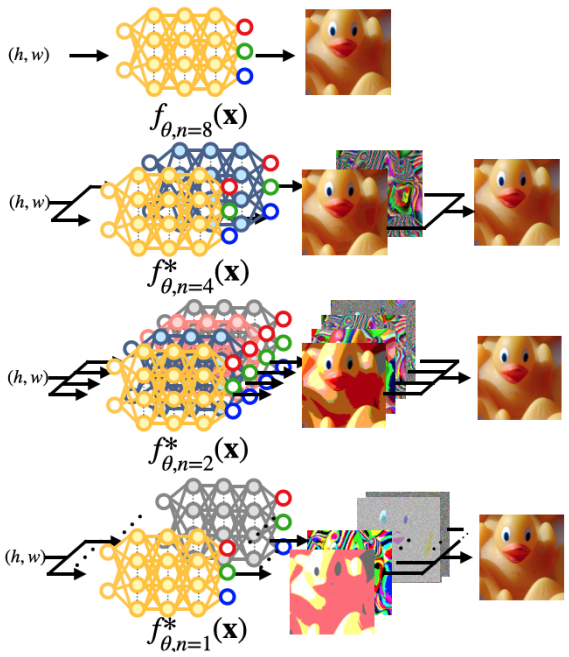


Figure 1. Schematic diagram of the parallel model used for the validation in the experiment section. $f_{\theta,n=k}$ indicates INRs that require k-bit precision with a given parameter θ.

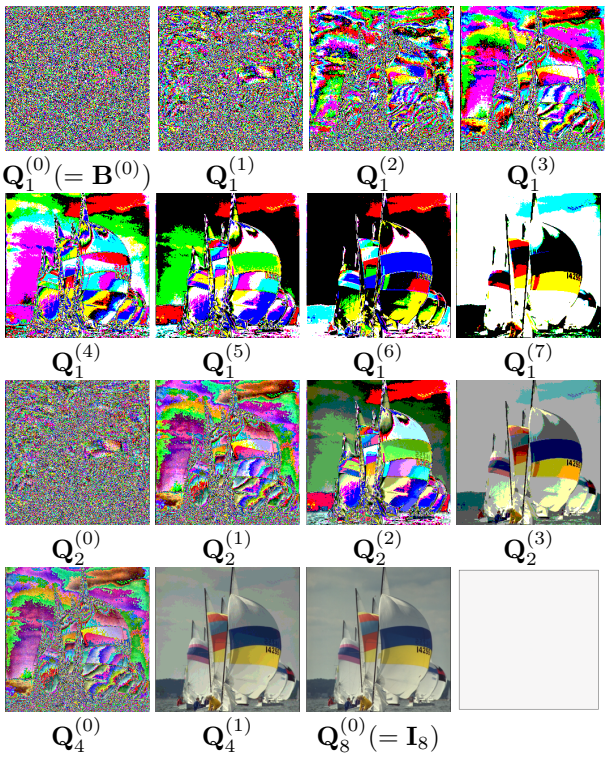


Figure 2. Quantized representations depending on k-bit precision.

analysis. The quantized representation is a generalized form of our main paper's bit-plane decomposition. We use a model without a bit axis for a fair comparison with SIREN.

Let n-bit images be $\mathbf{I}_n = \mathbf{Q}_n$, where $\mathbf{Q}_k^{(i)} \in Q_k^{H\times W\times 3}$. Images are represented as $f_{n=8} : \mathbb{R}^2 \rightarrow \mathbf{I}_8$ or $f_{n=16} : \mathbb{R}^2 \rightarrow \mathbf{I}_{16}$ for 8-bit and 16-bit, respectively. A bit-plane decomposition method reduces d to its divisor k, i.e., $k \in \text{div}^+(n)$, thereby reducing $\mathcal{U}_d(n)$. Instead of n-bit images, we parameterize a quantized set of images. Then quantized images $([\mathbf{Q}_k^{(i)}]_{i=0}^{\frac{n}{k}-1} := [\mathbf{Q}_k^{(0)}, \cdots, \mathbf{Q}_k^{(\frac{n}{k}-1)}])$ is a sequence that satisfies:

$$\mathbf{I}_n = \frac{1}{2^n-1}\sum_{i=0}^{\frac{n}{k}-1}(2^k)^i\mathbf{Q}_k^{(i)}. \tag{20}$$

Specifically, when $\mathbf{Q}_{k=1}^{(i)}$, it is bit-plane $\mathbf{B}$ and it is the method of our main paper. We present example images of $\mathbf{Q}, \mathbf{B}$, and $\mathbf{I}$ in Fig. 2.

The validation experiment for our hypothesis is representing an n-bit signal is employing $\frac{n}{k}$ parallel sequence of INRs i.e. :

$$[\mathbf{Q}_k^{(0)}, \cdots, \mathbf{Q}_k^{(\frac{n}{k}-1)}] \simeq [f_{\theta,k}^{(0)}, \cdots, f_{\theta,k}^{(\frac{n}{k}-1)}] \tag{21}$$

$$\mathbf{Q}_k^{(i)}(\mathbf{x}) \simeq f_{\theta,k}^{(i)}(\mathbf{x}) \tag{22}$$

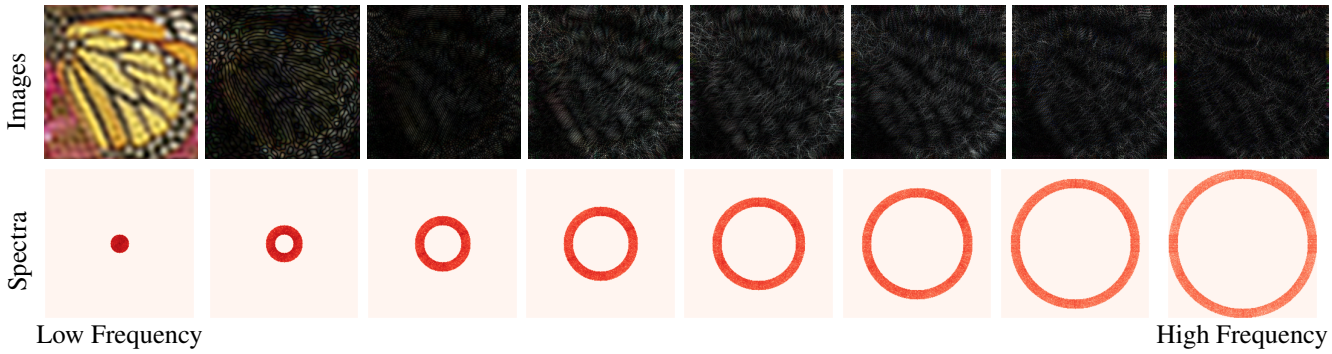
We denote each INR as $f_{n=k}$, meaning the INR with k-bit precision. Further, f_k^* indicates an INR that satisfies the required k-bit precision. Note that $f_{\theta,n=8}$ indicates the baseline SIREN model. Since we set all parameters to have identical numbers, increasing the threshold of error ($\epsilon(n)$) is identical to bringing closer to the upper bound $\mathcal{U}_d(n)$. For example, the second row of Fig. 1 indicates two models that require 4-bit precision and predicting 4-MSBs and 4-LSB ($[\mathbf{Q}_4^{(0)}, \mathbf{Q}_4^{(1)}]$). All evaluation follows the equation below:

$$\mathbf{I}_n(\mathbf{x};\theta|k) = \frac{1}{2^n-1}\sum_{i=0}^{\frac{n}{k}-1}(2^k)^i\mathcal{Q}_k(\hat{f}_{\theta,n=k}^{(i)}(\mathbf{x})). \tag{23}$$

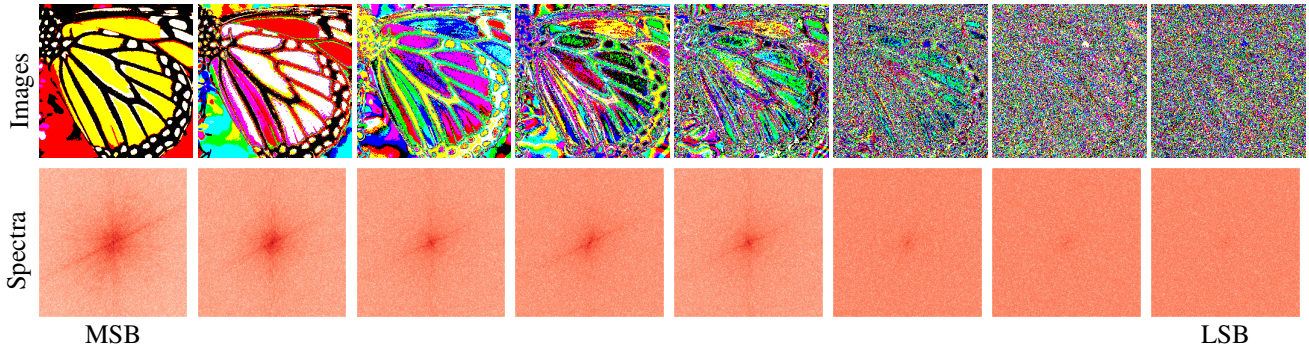
Eq. (23) is a generalized form of the equation of the main paper. We provide pseudocode, Algorithm 1 and Algorithm 2 for each method bit-plane decomposition and quantized representation, respectively.

3. Bit Bias & Spectral Bias

Our observation, *Bit Bias*, is highly correlated to the *Spectral Bias*; however, it is not identical. Fig. 3 provides visual information about the difference between *Bit-Bias* and *Spectral-Bias*. In Fig. 3a, we perform a Fourier transform on a single image and divide the frequencies into 8 bins, i.e., masking. After applying masking, we perform an inverse transform to obtain the resulting images. It shows the distribution of high-frequency components in the spatial domain, showing



(a) Frequency-based image decomposition and its spectrum



(b) Bit-plane image decomposition and its spectrum

Figure 3. Frequency-based decomposition (Fig. 3a) and bit-plane-based decomposition (Fig. 3b).

(a) 16bit bit-wise bias test image (b) Underfitted Image of Fig.4a (c) Waveform in the bit-axis

Figure 4. The sample image for the **Bit-wise bias** experiment (Fig. 4a). The image includes different bit-frequency. The Fig. 4c indicates waveforms of signals with different bit-frequency.

Algorithm 1 Bit-Plane Decomposition Algorithm

Input: `image` (tensor), `bits` (integer)
Output: `bit_planes` (list)
function BIT_DECOMPOSITION(image, bits)
    `bit_planes` ← [ ] ▷ Initialize an empty list
    for $i = 0$ to `bits` - 1 **do**
        `bit_planes.append(image % 2)`
        `image` ← `image` // 2 ▷ Integer division by 2
    end for
    return `bit_planes` ▷ Return bit-planes
end function

that high frequencies are concentrated in the wing's patterns.

In Fig. 3b, we present bit-planes. For example, determining LSB possesses high-frequency components. This corresponds to a problem of determining whether each pixel value is even or odd, which is equivalent to a Bernoulli distribution with a probability of 0.5. High-frequency components are indeed present. However, these high-frequency components do not always exist in the LSB alone. The spectrum in Fig. 3b shows that high-frequency components are also significantly present in the MSBs.

4. Details for Bit-Spectral bias experiment

Fig. 4 includes the image and its under-fitted prediction for the bit-spectral bias experiment. Extracting and comparing values with different bit frequencies from natural images is unsuitable because there are many variables, such as the spatial frequency of the image or surrounding pix-

Algorithm 2 Quantized Representation Algorithm

Input: `bit_planes` (list), `bits` (integer)
Output: Quantized Representations
function PARTIAL_COMPOSITION(bit_planes, bits)
    `basis` $\leftarrow 2^{\texttt{torch.arange(0,bits)}}$ ▷ Calculate basis
    `n` $\leftarrow 2^{(\texttt{bits})} - 1$ ▷ Normalize term
    `iters` $\leftarrow$ `len(bit_planes)//bits - 1`
    `res` $\leftarrow$ [ ] ▷ Initialize an empty list
    for $i = 0$ to `iters` **do**
        `part` $\leftarrow$ `bit_planes[i:i+bits]`
        `part` $\leftarrow$ `part * basis` ▷ Multiply Bit Weight
        `part` $\leftarrow$ `part / n` ▷ Normalize to [0,1]
        `res.append(part)`
    end for
    return `res` ▷ Return computed value
end function

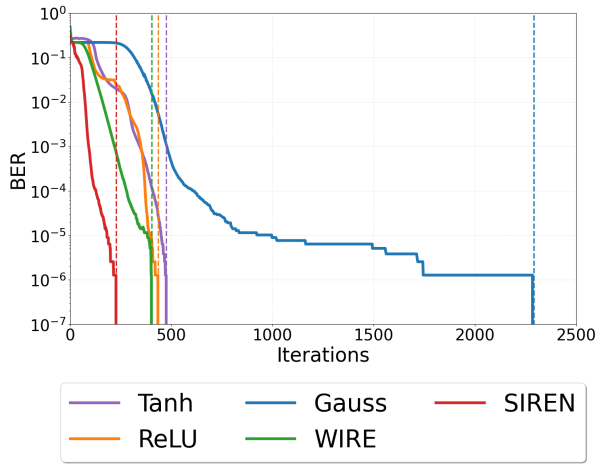


Figure 5. Quantitative ablation study on activation function. Vertical dashed lines indicate the iterations when each model achieves lossless.

els. Therefore, we control variables using the Fig. 4a and experiment with different bit frequencies for each part as an experiment variable. Fig. 4c illustrates the waveform and binary representation of each pixel value based on bit-spectral frequency. Fig. 4b shows the qualitative result of bit-spectral bias that high-frequency values, such as 43,690 (=1010101010101010_2) or 21,845 (=0101010101010101_2), are hard to fit.

5. Activation

We conduct ablation studies for the activation and loss function of the proposed method, as shown in Fig. 5. We utilized a 16-bit sample image in the TESTIMAGE dataset [2] and reduced the parameter count to observe convergence speed. We adopt periodic activation function [37] for fast conversion. To support this, we set the activation function as the controlled variable in Fig. 5. In Fig. 5, the Gauss activation [32] converged slower than other baselines. Fig. 5 indicates that our method achieved lossless implicit representation with a sufficient number of iterations and parameters, regardless of the activation function used.

6. Application Implementation Details

6.1. Ternary Implicit Neural Representation

We detach bias terms in each affine linear layer for a lighter INR. Representing lossless complex images with networks consisting only of sums and differences is challenging. Unlike the image representation with full-precision parameters, the parallel network has been implemented. We train each 1.58-bit INR from scratch per bit plane. The upper bound $\mathcal{U}_d(n)$ remains identical for 16-bit images. However, networks with limited precision have challenges due to the 3-dimensional complexity. The network to represent each bit-plane includes a 5-layer with 256 hidden channels. We replace sinusoidal activations with Gaussian Error Linear Unit (GELU) activation [15] following the prior works [23, 45]. The total number of iterations is 200K, with a learning rate scheduler decayed by a factor of 0.01 every 20K steps.

6.2. Lossless Compression

We conduct experiments using the MNIST and Fashion MNIST datasets. We selected 1,000 images for training and 100 images for testing. The network architecture follows RECOMBINER, with two main differences: it uses 3D coordinates as input and outputs the result using BCE Loss. The network consists of 3-layer MLP with 64 channels and sine activations.

7. Floating Point Representation

Our method is concentrated on presenting signals with a fixed bit-precision. However, following the standard format of the floating point representation, we expand our method to represent floating point (FP) representation. The straightforward approach is converting the numbers into binary numbers directly and aligning them to the longest bit length. We utilize the definition of floating-point representation. The floating number is formulated as below:

$$x = m \times 2^e, \tag{24}$$

where $m \in Q_{24}$ indicates a mantissa and $e \in Q_8$ indicates an exponent including a sign. Note that the range of m depends on the normalization method. The audio fitting experiment further supports the robustness of our approach. Audio has lower spatial complexity than an image but demands more bits. We serialize information, including signs for estimation and recombine them as below:

$$\mathbf{O}^*_\theta(\mathbf{x}) = \sum_{i=8}^{31} \mathcal{Q}_1(\mathbf{m}_{\theta,n=1}(\mathbf{x}, i)) \times 2^{\sum_{i=0}^{7} \mathcal{Q}_1(\mathbf{e}_{\theta,n=1}(\mathbf{x},i))}, \tag{25}$$

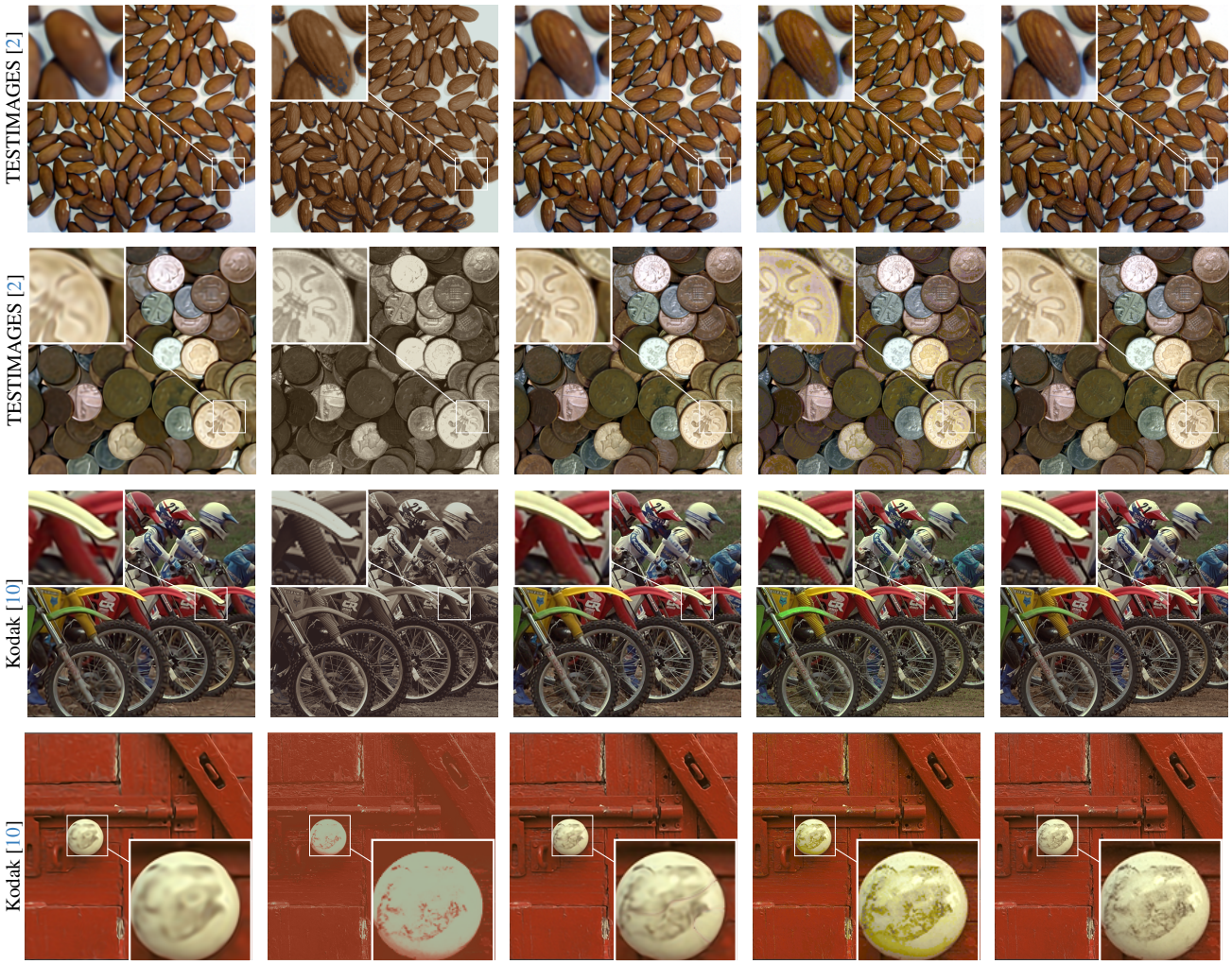


Figure 6. Qualitative comparison of under-fitted images (# of Iterations : 200) in **512**×**512** images.

where $\mathbf{m}_{\theta,n=1}$ and $\mathbf{e}_{\theta,n=1}$ are predicted mantissa and exponent, respectively. $\mathbf{O}$ indicates a FP32 audio signal. We estimate each part using a single network; however, separate notations are needed to avoid confusion, i.e., $f_\theta(\mathbf{x}, i) = [\mathbf{m}_\theta(\mathbf{x}, i); \mathbf{e}_\theta(\mathbf{x}, i)]$.

8. Additional Results

Extended Model In the main paper, we employed sinusoidal activations for generality; however, INRs with enhanced expressiveness perform more efficient results. Inspired by recent methods, we present a more efficient approach for accelerating the convergence of our INRs than using sinusoidal activations alone, as in the main paper. We utilize a hash-table [46] and modified sinusoidal activations [22]. We notate the method as '**Ours**$^{++}$'. This offers the following advantages: 1) faster convergence, 2) increased capacity for representable samples.

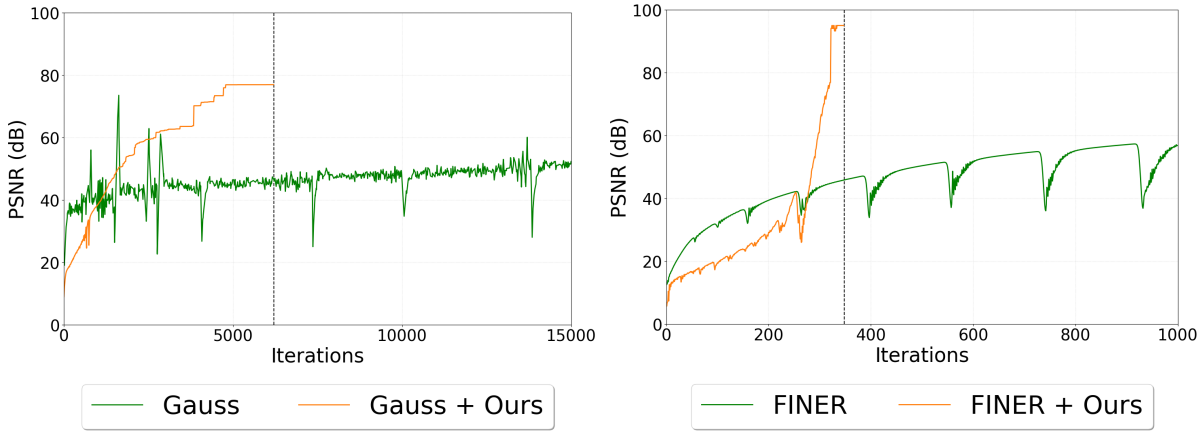
We conduct 2D image fitting experiments on 512×512 images which is a larger resolution than our main paper. Fig. 6 show that our method converges faster than other approaches while preserving details. Tab. 3 demonstrates that the applied method converges much faster while achieving lossless representation. Additionally, Tab. 4 shows that

Method	SIREN [37]	DINER [46]	FINER [22]	**Ours**
Iterations (↓)	400			**193**($\pm$72)
TESTIMAGES [2]	36.51	30.98	38.71	∞
Kodak [10]	32.91	32.85	34.31	∞

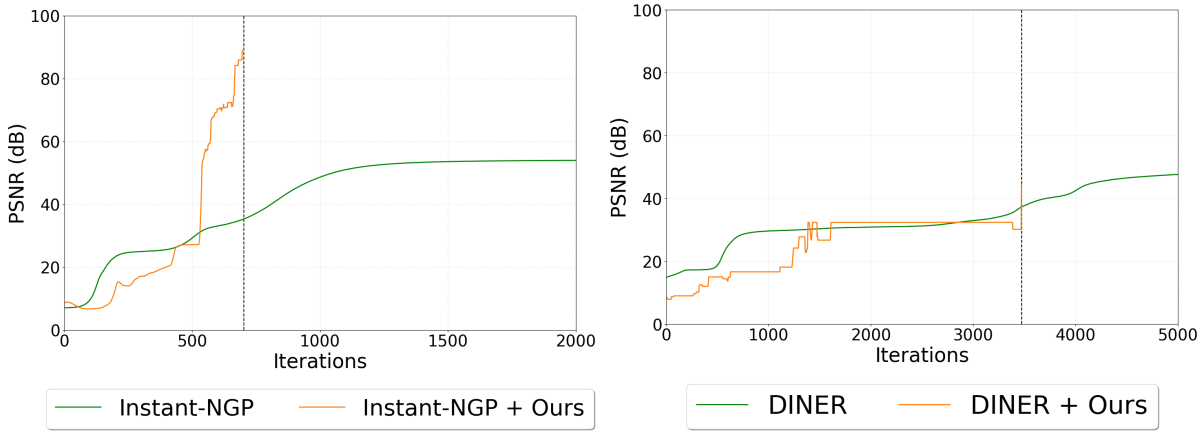
Table 3. **Quantitative comparison** on 512 × 512 image fitting with existing INR methods. The iteration number of our methods indicates '$mean(\pm std)$' for the total dataset.

	Kodak [10]		TESTIMAGES [2]	
	#Iter.(↓)	PSNR (↑)	#Iter.(↓)	PSNR (↑)
SIREN + Ours	790	∞	3450	∞
Ours^{++}	**180**		**214**	

Table 4. Quantitative comparison results of **Ours**$^{++}$ with our method in the main paper. The experiment has been conduct on 256 × 256 resolutions



(a) Activations



(b) High-dimensional input

Figure 7. Comparison of training curve. Vertical dashed lines indicate the iterations when our models achieve lossless.

Bits Per sub-Pixel (bpsp)(↓)	TESTIMAGES[2]
TIFF [29]	16.0017
JPEG2000 [42]	12.4021
PNG [33]	14.0001
Ours^{++}	10.4411

Table 5. Quantitative Comparison for lossless compression on 16-bit images.

	Experiment Group			SIREN [37]	Ours
Bit-precision (n)	1	2	4	8	1
#Params. (M)	**1.311**	1.322	1.318	1.316	1.316
Mem. (MB)	14.17	14.17	14.16	**14.15**	14.18
FLOPs (M)	**1.303**	1.316	1.314	1.313	10.51
Time (ms)	3.116	1.823	0.922	0.771	**0.761**

Table 6. Comparison of computational resource usage (parameters, memory, FLOPs, and time) among SIREN [37] and our method for bit-precision settings.

	Time(ms)
RECOMBINER [14]	0.455
RECOMBINER + **Ours**	0.876

Table 7. Decoding time for a single image used in the compression.

'Ours^{++}' converges faster than the method in the main paper.

Training Curve In Fig. 7, we provide additional training curves that could not be included in the main text due to space constraints. These curves illustrate trends when combined with each model: high-dimensional inputs (Fig. 7b) and activations (Fig. 7a).

Lossless Compression We observed that the hash table generated by our method (Ours^{++}) has low entropy, making it highly suitable for compression. Tab. 5 present a quantitative result on compressing 16-bit images. We applied quantization to the hash table, followed by entropy coding. Despite the challenges of compressing 16-bit images, this approach outperforms traditional codecs, demonstrating superior performance.

Computational Complexity In Tab. 6, we present a comparison of the computational resources including a SIREN [37] and our experiment group in Tab. 3 of the main paper. FLOPs and time are reported for all models based on the computation of a single pixel. Our method requires FLOPs proportional to the bit depth linearly, but memory usage and computation time remain nearly unchanged. Due to the parallel processing nature of GPUs, the computation time shows marginal differences. In Tab. 7, We show a decoding time for a single image used in the compression, demonstrating marginal differences.