Improved Positional Encoding for Implicit Neural Representation based Compact Data Representation

Bharath Bhushan Damodaran, Francois Schnitzler, Anne Lambert, Pierre Hellier
InterDigital, Inc.,
Rennes, France
bharath.damodaran@interdigital.fr

Abstract

Positional encodings are employed to capture the high frequency information of the encoded signals in implicit neural representation (INR). In this paper, we propose a novel positional encoding method which improves the reconstruction quality of the INR. The proposed embedding method is more advantageous for the compact data representation because it has a greater number of frequency basis than the existing methods. Our experiments shows that the proposed method achieves significant gain in the rate-distortion performance without introducing any additional complexity in the compression task and higher reconstruction quality in novel view synthesis.

1. Introduction

Implicit neural representation (INR) or neural radiance field (NRF) has gained popularity recently, due to its capability to represents different kinds of multi-dimensional signals [11]. INR represents a signal by over-fitting a continuous function (neural network) which takes as input the coordinates of the signal and outputs the pixel color values [16, 15], in the case of signal regression. By over-fitting, INR learns to compactly represent the signals in the lower-dimensional space by discarding irrelevant information when the number of parameters is lower than the length of the underlying signal. The representation capacity of the INR depends on the number of parameters used to approximate the signals. Thus, the approximation quality can be adjusted by changing the architecture of the INR network. INR can be used for compact data representation [9, 3], image and video compression, since storing an image/video amounts to storing the weights for the neural network [8, 12, 6, 7, 13]. Reconstructing the image or signal then amounts to extracting the weights and evaluating the neural network for all coordinates.

Directly using a multi-layer perceptron (MLP) results in an overly smooth reconstruction due to the spectral bias of the regular MLP. Tancik *et al*. [16] showed MLP with ReLU activation functions are not suitable to encode signals with high frequency content. To overcome the spectral bias, Sitzmann *et al*. [15] replaced ReLU with sine activation function whereas Tancik *et al*. [16] used positional encoding with random Fourier features followed by MLP with ReLU activation function.

Positional encoding using random Fourier features maps the input coordinates to a high dimensional embedding as $\gamma(\mathbf{x}) : \mathcal{R}^d \rightarrow \mathcal{R}^{2D}$ using D dimensional random basis. The embedding dimension as well as the number of random Fourier basis has a direct impact on the reconstruction quality. However, in Tancik *et al*. [16], the number of random Fourier basis used is only half the size (D) of the embedding dimension ($2D$). Thus, the existing Fourier embedding used in the INR is limited in covering the frequency spectrum and also in its reconstruction quality, especially when the embedding dimension is small.

In this paper, we propose an alternative positional embedding method for INR to improve reconstruction quality. Our proposed embedding method contains the same number of basis vectors as the embedding dimension. Therefore, it covers the frequency spectrum better. Additionally, our proposed method does not increase the encoding or decoding complexity of the signals nor the size of the bit-stream in compression tasks. We evaluated our proposed method on image reconstruction task, image compression and novel view synthesis and the results showed that our proposed method has significant gains without any additional increase in bit-stream size and complexity.

The rest of the paper is organized as follows: Section 2 introduces the implicit neural representation and existing Fourier feature mapping, section 3 presents our proposed method, section 4 reports on our experimental results, and conclusion is derived in section 5.

2. Implicit Neural Representation

Let $\mathbf{I} \in \mathbb{R}^{W \times H \times 3}$ be a color image, $x, y \in \mathbb{R}$ be the pixel coordinates in the normalized range $[-1, 1]$, $I(x, y)$ denotes the pixel values (RGB) at the coordinates x, y and $\gamma(x, y)$ a positional encoding of the coordinates. The INR is a neural network $f_{\boldsymbol{\theta}}$, parameterized by the weights $\boldsymbol{\theta}$ such that it maps the given coordinates to the pixel intensity values (RGB). In other words, $\forall x, y, f_{\boldsymbol{\theta}}(\gamma(x, y)) \sim I(x, y)$. Without loss of generality, it can be extended to any multi-dimensional signals.

The weights $\boldsymbol{\theta}$ of the INR are estimated by over-fitting (minimizing) the following loss function

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} L(x, y, \boldsymbol{\theta}) = \frac{1}{N} \sum_{x,y} d(I(x, y), f_{\boldsymbol{\theta}}(\gamma(x, y))), \tag{1}$$

where the sum is over all the pixels in the image ($N = W \times H$), W, H is width and height of the image, d is any distortion metric which measures the discrepancy between the predicted (reconstructed) pixels by $f_{\boldsymbol{\theta}}$ and the actual pixel values of the image I. The metric d is preferably a differentiable distortion measure, such as mean squared error or perceptual metric such as LPIPS. In this paper, mean squared error (MSE) is used as the distortion metric. Once the equation (1) is optimized, at the inference image can be reconstructed by evaluating $f_{\boldsymbol{\theta}^*}$ over all the pixel coordinates.

Data representation: The optimized $\boldsymbol{\theta}^*$ can be used as the data representation for the down-stream tasks. Compressing an image $\mathbf{I}$ is equivalent to encoding the values of the weights $\boldsymbol{\theta}^*$ in the bit-stream. For compression it is not possible to choose large neural network for better reconstruction quality as it would increase bit-length. Thus, the number of weights is constrained at the expense of the distortion. For each image I, there is one specific INR f_θ which is overfitted to the given image I. This is different from the end-to-end compression method [14, 1, 2]. To generate bit-streams of different sizes, INR is trained with different number of hidden layers and nodes. Figure 1 illustrates an implicit neural network (INR) based image compression system.

Positional Encoding with Fourier features: The Fourier feature mapping is based on the Bochner's theorem to approximate a shift-invariant kernels. Tancik *et al.* [16] used random Fourier feature (RFF) mapping $\gamma : \mathcal{R}^d \rightarrow \mathcal{R}^{2D}$ to the input coordinates before feeding them to MLP with ReLU activation functions. The Fourier feature mapping of the coordinate $\boldsymbol{v} = (x, y)$ is defined as follows:

$$\gamma(\boldsymbol{v}) \in \mathcal{R}^{2D} = [\cos\left(2\pi \boldsymbol{w}_1^T \boldsymbol{v}\right), \sin\left(2\pi \boldsymbol{w}_1^T \boldsymbol{v}\right), \ \ldots, \cos\left(2\pi \boldsymbol{w}_D^T \boldsymbol{v}\right), \sin\left(2\pi \boldsymbol{w}_D^T \boldsymbol{v}\right)]^T, \tag{2}$$

where the coefficients $\boldsymbol{w}_i$ are the Fourier basis frequencies

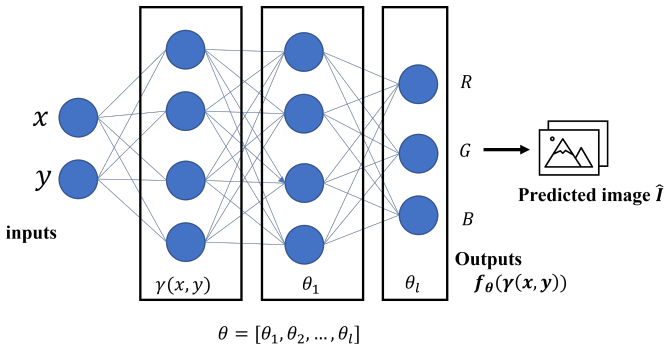

Figure 1. Implicit neural representation (INR) with Fourier feature mapping. $x, y \in \mathbb{R}$ is the normalized input coordinates, $\gamma(x, y)$ is the Fourier feature mapping, $\boldsymbol{\theta}'s$ are weights of the MLP. For the compression, $\boldsymbol{\theta}$'s are encoded in the bit-stream and transmitted to the decoder side.

when the mapping is seen as a Fourier approximation of a shift-invariant kernel function. The basis vectors $\boldsymbol{w}_i$'s are randomly sampled from the Gaussian distribution with appropriate band-width σ, i.e $\boldsymbol{w}_i \sim N(0, \sigma), i = 1 \ldots D$. This positional encoding is used in the existing INR and NeRF literature.

For the mapping dimension of $2D$, only D number of Fourier basis are sampled. One could note that the number of sampled frequencies is half the number of the mapping size. This could be an issue when the mapping size is small, which is the case in compression tasks.

3. Proposed method

To increase the number of random Fourier basis in the Fourier feature mapping, here we propose an alternative positional encoding which is also based on the Bochner's theorem, and we label our proposed method as *RFF-cosine mapping*. Let $\phi : \mathcal{R}^d \rightarrow \mathcal{R}^{2D}$ be the mapping of the our proposed RFF-cosine mapping, which maps the input coordinates to the RFF-cosine feature mapping as

$$\phi(\boldsymbol{v}) = [\sqrt{2}\cos\left(2\pi \boldsymbol{w}_1^T \boldsymbol{v} + b_1\right), \sqrt{2}\cos\left(2\pi \boldsymbol{w}_2^T \boldsymbol{v} + b_2\right), \ldots, \sqrt{2}\cos\left(2\pi \boldsymbol{w}_{2D}^T \boldsymbol{v} + b_{2D}\right)]^T, \tag{3}$$

where $\boldsymbol{w}_i' s, i = 1, \ldots, 2D$ are randomly sampled from the Gaussian distribution with the bandwidth parameter (σ), and bias vectors $b_i' s$ are randomly sampled from the uniform distribution in $[0, 2\pi]$. The advantage of using our proposed method is directly evident by comparing eqn (3) and (2), where our proposed RFF-cosine feature mapping has more (twice) number Fourier basis frequencies with the same mapping size as compared to eqn (2). This allows to encode high frequency information in the signals better than the existing positional encoding method. By replacing our proposed RFF-cosine mapping in eqn (1), the loss function

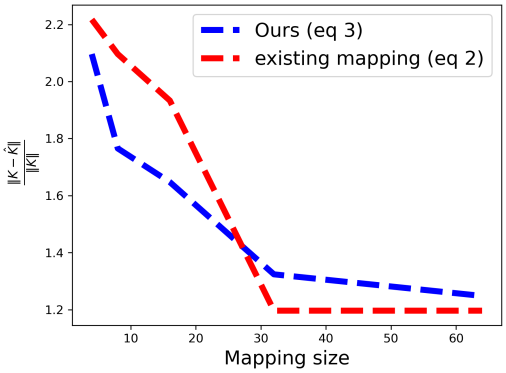

Figure 2. Kernel approximation error $\frac{\|\boldsymbol{K}-\hat{\boldsymbol{K}}\|}{\|\boldsymbol{K}\|}$ of the $1D$ sine wave function $(3sin(t) + U(0,1), t \in [0, 8\pi])$ using our proposed mapping and existing mapping. $\boldsymbol{K}$ is the Gaussian RBF kernel matrix (with std dev. σ computed using 5^{th} percentile of pairwise distances), and $\hat{\boldsymbol{K}}$ is the approximated kernel matrix using either our mapping (3) or existing mapping (2). Lower approximation error is better.

to be minimized is as follows

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} L(x, y, \boldsymbol{\theta}) = \frac{1}{N} \sum_{x,y} d(I(x, y), f_{\boldsymbol{\theta}}(\phi(x, y))). \tag{4}$$

For the signal compression, as the Fourier embeddings are generated from randomly sampled frequency basis, it is not necessary to encode the frequency basis in the bitstream, we only need to write the random seed used to sample frequency basis in the bitstream. Our proposed method needs only to write one additional random seed used for the sampling from the uniform distribution, if different seeds are used. Thus, our proposed positional encoding increases neither the size of the bitstream nor the complexity of encoding or decoding the signals.

Illustration: The benefit of our proposed feature mapping can be analyzed through the lens of neural tangent kernel (NTK) and kernel approximation. Due to the greater number of frequency basis, the eigenvalue decay of the NTK of our proposed feature mapping might be slower than the existing Fourier feature mapping in (2), thus our method can encode more frequency content of the signal in the low mapping size. However, in the higher mapping size the eqn (2) might be able to cover the entire frequency spectrum of the signal, thus performs equally similar to our proposed mapping or better. This can be validated with the toy experiment to approximate the Gaussian kernel using both the feature mapping. Figure 2 reports the kernel approximation error of the $1D$ sine wave function (with 1000 samples), and show that our proposed feature mapping has smaller approximation error in the lower the mapping size, and as the mapping size increases the existing mapping has better approximation error. Thus, validating our proposal to use our proposed feature mapping in the tasks where the large mapping size or large neural networks cannot be used.

4. Experimental Results

We evaluated our proposed RFF-cosine mapping method on different sets of tasks: image compression, and novel view synthesis. Further, we evaluated the impact of the mapping size, and showed the superiority of our proposed method in low mapping size settings.

Image compression: We choose a subset (images 5 to 14) of the Kodak dataset [10]. The RGB PSNR and bits per pixel (BPP) are used as the distortion and rate measure. We use the MLP architecture with varying number of hidden layers and hidden nodes $\{(5, 20), (5, 30), (10, 30), (10, 40)\}$, for different bit-rates as in the [8]. The loss function is minimized with Adam optimizer using learning rate $2e-4$ for $50K$ iterations. To choose bandwidth parameter σ for the Fourier mapping, we performed experiments using few images with $\{1, 5, 10\}$ and we observed that $\sigma = 1$ gives optimal performance for the compression task. For the compression, we encode the weights and bias in the half-precision as in [8].

The rate-distortion performance of our proposed embedding method and the existing one is reported in Table 1 with different network architectures and mapping size. The results reveal that our proposed mapping outperforms the existing one across different bit-rates and with different mapping size. More specifically, our method has a significantly better PSNR than the existing method in the low mapping size (more than 2dB PSNR in average) though the mapping size is the same, which proves the advantage of having more frequency basis.

Further, to quantify the bitrate gain in $\%$ we computed Bjontegard BD rate gain [5] and it is presented in figure 3. It demonstrates that our proposed embedding has about 98% BD-rate gain at the low mapping size and about 10% BD-rate gain at the higher mapping size. These gains are very significant in the compression literature, and they are achieved without any additional (encoding and decoding) complexity and increase in the size of bit-stream.

Novel view synthesis: Finally, we evaluate our proposed embedding method on the novel view synthesis. For this, we applied our proposed method on the recent NeRF method named *Nope-NeRF* [4], and compared with the existing Fourier features. We conducted experiments on the *Ignatius* sequence from *tanks* dataset, and after training we used 8 scenes to evaluate novel view synthesis quality. We evaluated the quality of the synthesis using PSNR, Structural Similarity Index Measure (SSIM) and Learned Perceptual Image Patch Similarity (LPIPS). We followed a similar experimental protocol to [4]. For the Fourier feature mapping, we conducted experiments with bandwidth parameter

Table 1. Rate distortion (RD) performance of our proposed embedding method and the existing one for different bit-rates with different mapping sizes evaluated on the subset (images 5 to 14) of Kodak dataset . Q1-Q4 are different MLP architectures. Rates are measured in bits per pixel (bpp), and distortion in PSNR. The best results are in **bold**.

Mapping size	Method		Q1	Q2	Q3	Q4
8		BPP	0.0782	0.1661	0.3111	0.6202
	PSNR	Existing method	18.26	18.70	19.04	19.38
		Ours	**20.33**	**20.83**	**21.37**	**22.01**
16		BPP	0.0848	0.1759	0.3202	0.633
	PSNR	Existing method	21.67	22.48	22.69	**23.51**
		Ours	**22.15**	**22.79**	**22.85**	23.50
32		BPP	0.0977	0.1954	0.3385	0.6593
	PSNR	Existing method	22.42	22.94	23.16	23.82
		Ours	**22.56**	**23.11**	**23.38**	**23.95**
64		BPP	0.1238	0.2345	0.3750	0.7114
	PSNR	Existing method	22.96	23.40	23.75	24.19
		Ours	**23.14**	**23.45**	**23.79**	**24.33**

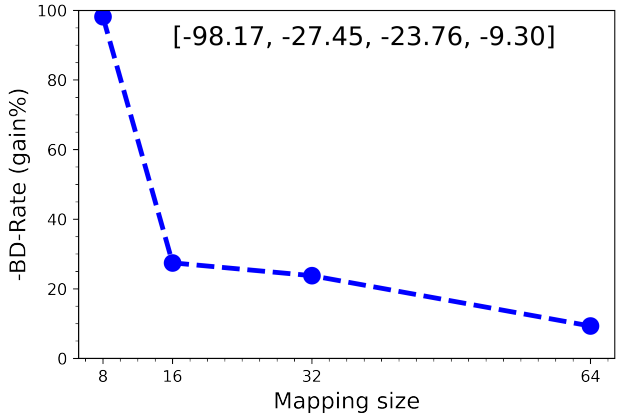

Figure 3. BD rate gain (in -%) of our proposed method with respect to the existing method over various mapping size. The exact gains are displayed in text.

$\{1, 5, 10\}$ with mapping size of 60 and we observed that $\sigma = 1$ gave the best results.

Table 2 reports the reconstruction quality of the Nope-Nerf with our proposed positional encoding and existing encoding method, and showed that with our proposed method Nope-Nerf achieved best results.

5. Conclusion

In this paper, we proposed an improved positional encoding for implicit neural representation. The proposed embedding method has a greater number of Fourier frequency basis than the existing Fourier feature mapping used in the INR literature. The superiority of our method is evaluated on image compression and novel view synthesis tasks, and showed that it offered significant BD rate gain in compression, and better reconstruction quality in view synthesis. In the future work, we will explore the quantization aware training [7] and entropic coding to further improve the compression efficiency.

Table 2. Reconstruction quality of the Nope-Nerf method for novel view synthesis with our proposed positional encoding and existing encoding. The best results are in **bold**.

Method	PSNR (↑)	SSIM (↑)	LPIPS (↓)
Nope-NeRF+(2)	23.79	0.60	0.50
Nope-NeRF+Ours	**23.91**	**0.61**	**0.49**

References

[1] Muhammet Balcilar, Bharath Damodaran, and Pierre Hellier. Reducing the amortization gap of entropy bottleneck in end-to-end image compression. In *2022 Picture Coding Symposium (PCS)*, pages 115–119. IEEE, 2022. 2

[2] Muhammet Balcilar, Bharath Bhushan Damodaran, Karam Naser, Franck Galpin, and Pierre Hellier. Latent-shift: Gradient of entropy helps neural codecs, 2023. 2

[3] Matthias Bauer, Emilien Dupont, Andy Brock, Dan Rosenbaum, Jonathan Schwarz, and Hyunjik Kim. Spatial functa: Scaling functa to imagenet classification and generation. *arXiv preprint arXiv:2302.03130*, 2023. 1

[4] Wenjing Bian, Zirui Wang, Kejie Li, Jia-Wang Bian, and Victor Adrian Prisacariu. Nope-nerf: Optimising neural radiance field with no pose prior. In *Proceedings of*

the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 4160–4169, 2023. 3

[5] Gisle Bjontegaard. Calculation of average psnr differences between rd-curves. *VCEG-M33*, 2001. 3

[6] Hao Chen, Bo He, Hanyu Wang, Yixuan Ren, Ser Nam Lim, and Abhinav Shrivastava. Nerv: Neural representations for videos. *Advances in Neural Information Processing Systems*, 34:21557–21568, 2021. 1

[7] Bharath Bhushan Damodaran, Muhammet Balcilar, Franck Galpin, and Pierre Hellier. Rqat-inr: Improved implicit neural image compression. In *2023 Data Compression Conference (DCC)*, pages 208–217, 2023. 1, 4

[8] Emilien Dupont, Adam Goliński, Milad Alizadeh, Yee Whye Teh, and Arnaud Doucet. Coin: Compression with implicit neural representations. *arXiv preprint arXiv:2103.03123*, 2021. 1, 3

[9] Emilien Dupont, Hyunjik Kim, S. M. Ali Eslami, Danilo Jimenez Rezende, and Dan Rosenbaum. From data to functa: Your data point is a function and you can treat it like one. In *Proceedings of the 39th International Conference on Machine Learning*, pages 5694–5725. PMLR, 17–23 Jul 2022. 1

[10] Eastman Kodak. Kodak Lossless True Color Image Suite (PhotoCD PCD0992). 3

[11] 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. In *ECCV*, 2020. 1

[12] Jonathan Schwarz and Yee Whye Teh. Meta-learning sparse compression networks. *Transactions on Machine Learning Research*, 2022. 1

[13] Jonathan Richard Schwarz, Jihoon Tack, Yee Whye Teh, Jaeho Lee, and Jinwoo Shin. Modality-agnostic variational compression of implicit neural representations. *arXiv preprint arXiv:2301.09479*, 2023. 1

[14] Mustafa Shukor, Bharath Bhushan Damodaran, Xu Yao, and Pierre Hellier. Video coding using learned latent gan compression. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 2239–2248, 2022. 2

[15] Vincent Sitzmann, Julien Martel, Alexander Bergman, David Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. *Advances in Neural Information Processing Systems*, 33:7462–7473, 2020. 1

[16] Matthew Tancik, Pratul Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *Advances in Neural Information Processing Systems*, 33:7537–7547, 2020. 1, 2