Positional Encodings as Group Representations: A Unified Framework

Derek Lim [* 1] **Hannah Lawrence** [* 1] **Ningyuan (Teresa) Huang** [2] **Erik H. Thiede** [3]

Abstract

Positional encodings are ubiquitous as an input featurization tool in language modeling, computer vision, and graph representation learning, enabling neural networks to capture important geometric structure of the input. Traditionally, positional encodings have been defined anew for each data domain. In this work, we reinterpret positional encodings for disparate data types — including sequences, grids, graphs, and manifolds — in the unifying framework of group representations. We show how to express existing positional encodings as group representations, and conversely, propose new positional encodings by choosing suitable groups and representations. We validate our framework with experiments on implicit neural representations of images and vector fields, highlighting the practical utility of such positional encodings for encouraging approximate equivariance and capturing geometric structure.

1. Introduction

Positional encodings are important ingredients for machine learning models in numerous domains, including implicit neural representations (Tancik et al., 2020), graph neural networks (Lim et al., 2023), and Transformers for text (Vaswani et al., 2017), images (Dosovitskiy et al., 2021), or various types of data that can be viewed as arrays (Jaegle et al., 2021). As such, many works have studied existing positional encodings or proposed new positional encodings. Our current work contributes to both of these directions.

First, we unify many existing positional encodings by interpreting them as group representations of a symmetry group of the data domain. We include concepts that are not traditionally viewed as positional encodings, such as the use of spherical harmonics to encode directions in Euclidean group equivariant networks (Thomas et al., 2018).

Grounded in our unified framework, popular positional encodings can be theoretically motivated as injecting equivariance priors and other geometric information into neural networks, without strict enforcement (Section 3). We give intuitive explanations for the approximate equivariance induced by group representation-based positional encodings (Section 2.2). Using this perspective, we propose new $SO(2)$ positional encodings for various data modalities (Section 4). Experiments with these encodings for implicit neural representations demonstrate that group representations are a promising space to explore for designing positional encodings (Section 5).

2. Pos. Encodings as Group Representations

2.1. Preliminaries

By a positional encoding, we mean a function $\gamma : \mathcal{X} \to \mathbb{R}^d$ that maps input $x \in \mathcal{X}$ to positional encodings $\gamma(x) \in \mathbb{R}^d$, where $\mathcal{X}$ denotes the space of positions. For example, $\mathcal{X} = \{1, \ldots, n\}$ for sequences with n tokens or graphs with n nodes. We will only consider absolute positional encodings here, and defer the discussions of relative positional encodings in Appendix A.1. It is often useful to identify $\mathcal{X}$ with G, either directly (if the two are equal) or indirectly (via G's action). In the very special case of regular grids (e.g. sequences or images), this identification is natural, as for instance the index $(x, y) \in \mathbb{Z}^2$ can be identified with the translation $(a, b) \mapsto (a + x, b + y)$. In general, this is also natural when $\mathcal{X}$ is a homogeneous space for G, meaning that G acts transitively on $\mathcal{X}$. In this case, we can map x to G by first fixing an origin $x_0 \in \mathcal{X}$, and then "lifting" x to the set x^G of group elements $g \in G$ such that $gx_0 = x$. For any space $\mathcal{X}$ acted on by G, define the orbit of $x \in \mathcal{X}$ as $\{gx : g \in G\}$.

A *group representation* is a vector space V together with a function $\rho : G \to GL(V)$ from group elements g to invertible linear maps $\rho(g)$ that respects group compositions, meaning $\rho(gh) = \rho(g)\rho(h)$ for $g, h \in G$. An *irreducible representation* (or irrep) is a group representation ρ such that there is no proper, nontrivial subspace $W \subset \mathbb{R}^n$ satisfying $\rho(g)W \subseteq W$ for all $g \in G$. Irreps are fundamental, as *any*

[*] Equal contribution [1] MIT CSAIL [2] Johns Hopkins University [3] Flatiron Institute. Correspondence to: Derek Lim <dereklim@mit.edu>, Hannah Lawrence <hanlaw@mit.edu>.

Presented at the 2^{nd} Annual Workshop on Topology, Algebra, and Geometry in Machine Learning (TAG-ML) at the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. 2023. Copyright 2023 by the author(s).

Table 1. Examples of positional encodings, interpreted as group representations. Y_ℓ^m denotes spherical harmonics, v_i the i-th eigenvector of the graph Laplacian, $J(r)$ a radial function, and $R^{2\times 2}(\theta)$ is the 2×2 rotation matrix by θ.

Data Type	Group	Encoding	Ref.
Text	T	$(x) \mapsto \{(\cos(\alpha x), \sin(\alpha x))\}_\alpha$	Vaswani et al. (2017)
Image	$T \times T$	$(x, y) \mapsto \{(\cos(\alpha_1 x + \alpha_2 y), \sin(\alpha_1 x + \alpha_2 y))\}_{\alpha_1,\alpha_2}$	Dosovitskiy et al. (2021)
Molecule	$SO(3)$	$(r, \theta, \phi) \mapsto \{Y_\ell^m(\theta, \phi)J(r)\}_{\ell,m}$	Thomas et al. (2018)
Graph	$S_{\lvert\mathcal{X}\rvert}$	$(x) \mapsto \{v_i(x)\}_i$	Lim et al. (2023)
Any (learned embedding)	$S_{\lvert\mathcal{X}\rvert}$	$x \mapsto \texttt{one\_hot}(x)$	Gehring et al. (2017)
Text (spherical embedding)	$SO(2)^{n/2}$	$(m) \mapsto \{\bigoplus R^{2\times 2}(m\alpha)\}_\alpha$	Su et al. (2021)
$\mathcal{X}$, homogeneous space	G	$x \mapsto \{\rho_\lambda(x^G)\}_\lambda$	Ours

group representation ρ can be written as a direct product of irreps.

Finally, we say a function $f : \mathcal{X} \to \mathcal{Y}$ is *equivariant* with respect to G if for all $g \in G, x \in X$, $f(gx) = gf(x)$. Similarly, a function f is *approximately equivariant* with respect to G if for all $g \in G, x \in \mathcal{X}$, $f(gx) \approx gf(x)$.

2.2. Positional Encodings as Irreps

We first show that many previously proposed positional encodings $\gamma(x)$ take the form $\gamma(x) = \mathrm{vec}(\rho(x))$ for some (often irreducible) group representation ρ, where vec flattens a matrix into a vector. More generally, positional encodings can be seen as residing in the equivariant vector space of some group representation (which subsumes the previous case). For ease of exposition, we focus on positional encodings as irreps, but elaborate on the subtle distinctions in Appendix A.2. See Table 1 for a list of examples. We expand on a few key instances now and leave further details to Appendix A.

Text and images. Sinusoidal positional encoding were popularized in (Vaswani et al., 2017) for encoding positions in 1D sequences, where we can identify the data domain with the translation group T. A similar identification can be made for images as signals supported on $T \times T$ (Dosovitskiy et al., 2021). The group irreps are $e^{i\alpha x}$ and $e^{i(\alpha_1 x + \alpha_2 y)}$, respectively; traditional positional encodings simply take the real and imaginary parts at certain frequencies.

Manifolds and graphs. In shape analysis and geometric machine learning, it is standard to generate positional encodings for a manifold or graph using the first eigenfunctions (i.e. those with smallest eigenvalues) of its Laplacian operator (Rustamov, 2007; Lim et al., 2023). Such functions are well-understood as the "smoothest" functions on the manifold (Rustamov, 2007). For a manifold and a closed subgroup G of the isometry group, the eigenspaces of the Laplacian are group representations of G (see Appendix A.3).

3. Why Group Representations?

3.1. Biasing Towards Equivariance

Group representations satisfy an equivariance property, in that when an input $h \in \mathcal{X} \equiv G$ is transformed by a group element $g \in G$, the corresponding group representation is transformed by g, since $\rho(gh) = \rho(g)\rho(h)$. Intuitively, models that process group representation positional encodings then have a (possibly weak) inductive bias towards equivariance. Thus, such models are in some sense endowed with equivariance priors, and may be more capable of learning (approximately) equivariant functions. We specifically elaborate on equivariance bias in both Transformers and multi-layer perceptrons (MLPs) that process group representation positional encodings as input.

Equivariance bias in transformers. Consider a layer of self-attention in a Transformer encoder (Vaswani et al., 2017), where the input solely consists of group representation positional encodings $\gamma(x) = \mathrm{vec}(\rho(x))$ of inputs $\mathbf{x} = [x_1, \ldots, x_n] \in \mathbb{R}^{d_x \times n}$, for some orthogonal representation ρ. Define $\gamma(\mathbf{x}) = [\gamma(x_1), \ldots, \gamma(x_n)] \in \mathbb{R}^{d \times n}$. Then the self-attention layer takes the form

$$f(\mathbf{x}) = W_V\, \gamma(\mathbf{x})\, \mathrm{softmax}\left(\gamma(\mathbf{x})^\top W_K^\top W_Q \gamma(\mathbf{x})\right), \quad (1)$$

for linear maps W_V, W_K, and W_Q. Suppose that W_V, W_K, W_Q are all scalar multiples of the identity. In Appendix A.4, we show that this self-attention layer is equivariant, in the sense that $f(g\mathbf{x}) = (I \otimes \rho(g))f(\mathbf{x})$.

Equivariance bias in MLPs. The neural tangent kernel (NTK) (Jacot et al., 2018) provides a theoretical model for the behavior of overparameterized MLPs during training. In Appendix A.5, we show that the NTK of an MLP with input positional encodings $\gamma(x) = \mathrm{vec}(\rho(x))$ for an orthogonal group representation ρ is group invariant. Thus, if the supervised prediction task is equivariant, then the MLP predictions are approximately equivariant.

3.2. Useful Structure in Group Representations

Besides encouraging equivariance, group representations have other desirable properties for a design space of features.

In most cases of interest (e.g. compact groups), every group representation can be decomposed into irreducible representations. Moreover, the matrix entries of irreducible representations form an orthonormal basis for square-integrable functions over the group by the Peter-Weyl theorem. Thus, irreducible representations are akin to a universal bank of features (Chughtai et al., 2023), which we can search over when designing positional encodings.

In addition, irreducible representations tend to hierarchically capture different levels of function resolution. For T, Fourier series of low frequencies capture low-resolution features, while high frequencies capture fine-grained changes. For $SO(3)$ acting on the sphere, the ℓ^{th} spherical harmonic is a polynomial of degree at most ℓ in x, y, and z. For a manifold, Laplacian eigenvectors are an orthonormal basis of functions over the manifold with varying smoothness, quantified in terms of the Dirichlet energy. By selecting irreps that correspond to multiple scales of resolution (e.g. exponentially varying α for T in Vaswani et al. (2017)), positional encodings enable downstream networks to process both fine- and coarse-grained changes (e.g. both background and sharp edges, for images) (Tancik et al., 2020).

Finally, group representations can motivate the design of positional encoding for *out-of-distribution* (OOD) generalization, where the distributions of the training set and test set are different (for example, short phrases during training time and long sentences during test time). In language modeling, Ruoss et al. (2023) improved OOD generalization by randomly subsampling an ordered set of positions (for the training data) from a much larger range of positions. From our group representation framework, this amounts to using representations from a larger group (based on the test sequence length) during training time.

4. Group Representations for Pos. Encodings

Suppose we have a new geometric data type or (possibly approximate) symmetry. What does our group representation framework prescribe as positional encodings? In particular, suppose G acts on the position space $\mathcal{X}$ in a meaningful way for the application of interest, and G has irreps ρ_λ. For $\mathcal{X}$ a homogeneous space of G, we define positional encodings by $\mathrm{vec}(\rho_\lambda(x^G))$, where x^G is the lift defined in Section 2.2.[1]

For general non-homogeneous spaces $\mathcal{X}$, we proceed by decomposing $\mathcal{X}$ into orbits, encoding an orbit's identity in the quotient space $\mathcal{X}/G$ and a point's position within the orbit separately. For example, the group $SO(2)$ consists of $2D$ rotations; however, $\mathbb{R}^2$ is not a homogeneous space of $SO(2)$. In this case, we embed $(x, y) \in \mathbb{R}^2$ via its orbit (indexed by the radius r in polar coordinates) and the position of (x, y) within its orbit (indexed by the angle θ in polar coordinates), separately[2]: $(r, \theta) \mapsto \{J_m(cr)e^{ik\theta}\}_{m,c,k}$, where J_m is the m-th Bessel function. In our experiments, we set $m = 0$, and sample both c and k; see Appendix D.2.

5. Experiments

5.1. 2D Implicit Neural Representations of Images

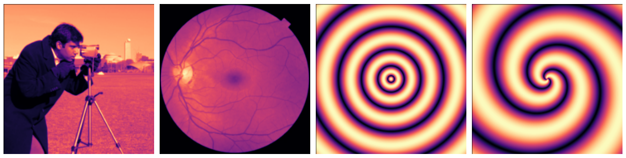

Figure 1. Images used in experiments of Section 5.1.

In this section, we explore learning implicit neural representations, where past work has shown that proper positional encodings are crucial (Tancik et al., 2020). We fix a grayscale image and positional encoding p, and then learn an MLP f_θ such that $f_\theta(\gamma(x, y))$ approximates the grayscale intensity at pixel (x, y). We consider two natural images (Cameraman, Retina) and two synthetic images (Radial, Spiral) — see Figure 1. The Retina and Spiral images are approximately invariant to the action of $SO(2)$, while the Radial image is exactly invariant to this action — that is, all pixels (x, y) of the same distance to the origin have the same value. We compare no positional encodings ($\gamma(x, y) = (x, y)$), $T \times T$ positional encodings as in Tancik et al. (2020), and our proposed $SO(2)$ positional encodings.

Table 2 (left) shows that our proposed $SO(2)$ positional encodings outperform the other positional encodings in the invariant and approximately invariant tasks. On the Cameraman image, $T \times T$ encodings perform slightly better; this is expected, as the variation in the image is better captured in Cartesian than polar coordinates. Figure 2 shows that $SO(2)$ encodings qualitatively provide rotationally symmetric biases, even at initialization and early in training.

5.2. 2D Implicit Neural Representations of Vector Fields

Section 5.1 showed the strength of $SO(2)$ positional encodings in tasks with some degree of $SO(2)$ invariance. Now, we consider tasks with $SO(2)$ equivariance. The setup is

[1] What makes a group "meaningful" for an application can range from strict symmetry, to simply a prior that the variation of the ground-truth function values is best captured along orbits (essentially, approximate symmetry).

[2] This procedure works for any non-homogeneous space acted on by G: if $x \in \mathcal{X}$, the lifting procedure in Section 2.2 equivariantly encodes x's position within its orbit, while any encoding of x's orbit itself is invariant by definition. These two encoding components can be combined in many ways. In the case of $SO(2)$, the invariant orbit embedding is a real scalar and the equivariant embedding is a complex unit vector, so it is natural to losslessly multiply the two. This is analogous to $SO(3)$ in Table 1.

Table 2. Test MSE of MLP with no positional encoding, $T \times T$ positional encodings, and $SO(2)$ positional encodings for implicit neural representation of images and vector fields. Lower is better. Mean and standard deviation are reported over 10 independent runs.

	Images				Vector Fields	
MSE ↓	Cameraman	Retina	Radial	Spiral	Radial	Spiral
None	.0249±.0017	.0040±.0004	.0572±.0144	.0071±.0033	.01099±.00431	.00075±.00018
$T \times T$	**.0209**±.0013	.0044±.0008	.0041±.0011	.0021±.0007	.00072±.00047	.00055±.00048
$SO(2)$	.0238±.0021	**.0024**±.0003	**.0024**±.0014	**.0015**±.0004	**.00050**±.00019	**.00027**±.00010

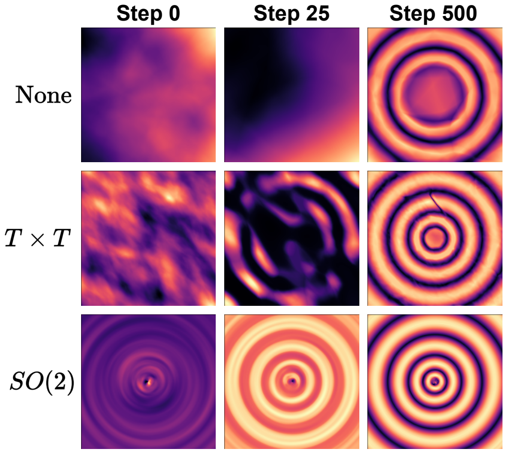

Figure 2. Learned representations for Radial image after 0, 25, and 500 training steps. Even early in training, $SO(2)$ positional encodings show the correct bias towards radially symmetric functions.

similar to Section 5.1, except we wish to predict a 2D vector field rather than an image. The neural network $f_\theta(\gamma(x, y))$ now seeks to approximate the 2D vector at location (x, y). We consider two vector fields: the Radial vector field is exactly $SO(2)$ equivariant, while the Spiral vector field is approximately $SO(2)$ equivariant. Table 2 (right) shows that $SO(2)$ positional encodings outperform the other choices on these tasks, validating the utility of our framework.

6. Conclusion

In this work, we gave a unified framework for several popular positional encodings as irreducible group representations. We motivated why irreps are a particularly useful building block for encodings, and using this intuition, proposed $SO(2)$-positional encodings as a proof of concept. In the future, we plan to extend this framework to other groups, such as $SE(2)$, and to further explore irrep positional encodings as a mechanism for incorporating approximate equivariance and geometric structure.

Acknowledgements

We would like to thank Behrooz Tahmasebi for discussions about kernels and Laplacians. DL is funded by an NSF Graduate Fellowship. HL is supported by the Fannie and John Hertz Foundation and the NSF Graduate Fellowship under Grant No. 1745302.

References

Chughtai, B., Chan, L., and Nanda, N. A toy model of universality: Reverse engineering how networks learn group operations. *arXiv preprint arXiv:2302.03025*, 2023.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.

Gehring, J., Auli, M., Grangier, D., Yarats, D., and Dauphin, Y. N. Convolutional sequence to sequence learning. In *ICML*, 2017.

Jacot, A., Gabriel, F., and Hongler, C. Neural tangent kernel: Convergence and generalization in neural networks. *NeurIPS*, 31, 2018.

Jaegle, A., Gimeno, F., Brock, A., Vinyals, O., Zisserman, A., and Carreira, J. Perceiver: General perception with iterative attention. In *ICML*, 2021.

Lim, D., Robinson, J. D., Zhao, L., Smidt, T., Sra, S., Maron, H., and Jegelka, S. Sign and basis invariant networks for spectral graph representation learning. In *ICLR*, 2023.

Ruoss, A., Delétang, G., Genewein, T., Grau-Moya, J., Csordás, R., Bennani, M., Legg, S., and Veness, J. Randomized positional encodings boost length generalization of transformers. *arXiv preprint arXiv:2305.16843*, 2023.

Rustamov, R. M. Laplace-beltrami eigenfunctions for deformation invariant shape representation. In *Symposium on geometry processing*, volume 257, 2007.

Su, J., Lu, Y., Pan, S., Wen, B., and Liu, Y. Roformer: Enhanced transformer with rotary position embedding. *CoRR*, abs/2104.09864, 2021.

Tancik, M., Srinivasan, P., Mildenhall, B., Fridovich-Keil, S., Raghavan, N., Singhal, U., Ramamoorthi, R., Barron, J., and Ng, R. Fourier features let networks learn high frequency functions in low dimensional domains. *NeurIPS*, 2020.

Thomas, N., Smidt, T., Kearnes, S., Yang, L., Li, L., Kohlhoff, K., and Riley, P. Tensor field networks: Rotation-and translation-equivariant neural networks for 3d point clouds. *arXiv preprint arXiv:1802.08219*, 2018.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. *NeurIPS*, 30, 2017.

A. More on Group Representations and Positional Encodings

A.1. Relative Positional Encodings

Many works also consider relative positional encodings $\eta : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^d$ that map pairs of points (g, h) to encodings $\eta(g, h)$ (Shaw et al., 2018). We note that relative positional encodings are often taken to be invariant to some relevant symmetry group. For instance, when encoding indices $i, j \in \mathbb{Z}$ in a 1D sequence, one choice of relative positional encoding is $\eta(i, j) = i - j$, which is invariant to translations: $\eta(i+t, j+t) = \eta(i, j)$. Therefore, one way to design relative positional encodings based on our framework is to take inner products of absolute positional encodings derived from an orthogonal group representation ρ, meaning $\gamma(x, y) = \mathrm{vec}(\rho(x))^\top \mathrm{vec}(\rho(y))$. This is because $\gamma(gx, gy) = \gamma(x, y)$, so the resulting relative positional encoding is group invariant. We capture this in the following elementary lemma, which we also use to demonstrate the equivariance bias for Transformers and MLPs with group representation positional encodings.

Lemma A.1. *If ρ is an orthogonal group representation, then* $\mathrm{vec}(\rho(gx))^\top \mathrm{vec}(\rho(gy)) = \mathrm{vec}(\rho(x))^\top \mathrm{vec}(\rho(y))$.

Proof. We can directly compute that

$$\begin{aligned}
\mathrm{vec}(\rho(gx))^\top \mathrm{vec}(\rho(gy)) &= \mathrm{trace}[\rho(gx)^\top \rho(gy)] && (2)\\
&= \mathrm{trace}[\rho(x)^\top \rho(g)^\top \rho(g) \rho(y)] && (3)\\
&= \mathrm{trace}[\rho(x)^\top \rho(y)] && (4)\\
&= \mathrm{vec}(\rho(x))^\top \mathrm{vec}(\rho(y)) && (5)
\end{aligned}$$

□

A.2. More Positional Encodings as Group Representations

In the main body of the paper, we explained how several positional encodings can be viewed through the lens of (often irreducible) group representations. A group representation refers to *both* a vector space V, and the group action of G on V via invertible linear maps, $\rho : G \to GL(V)$. Some existing positional encodings, such as those for text and images, can be readily viewed as irreps themselves: given some method of mapping a gridded position x (e.g. word position in a sentence, or pixel location in an image) to $x^G \in G$ (where G is T or $T \times T$), the standard positional encodings are understandable as $\rho_\lambda(x^G)$, with λ denoting the frequency of the irrep. The dimensionality of these irrep matrices owes to the fact that both T and $T \times T$ are abelian groups, so the complex vector space V (for irreps) is complex and one-dimensional, yielding the familiar positional encodings. For $SO(3)$ acting on the sphere S_2 in physical applications, the spherical harmonics are not precisely the irreps (which are the Wigner-D matrices), but they do appear in the Wigner-D matrices.

In contrast, however, consider the RoPe embedding (Su et al., 2021) for words. In this setting, we assume that each word is already assigned a semantic embedding vector v in $\mathbb{R}^d$. Implicitly, it is assumed that no two words' semantic embedding vectors have the same norm. Then, the *positional* information of a given word is encoded by applying a particular group representation of $SO(d)$ to v. In particular, if the word with semantic embedding vector v appears in the m^{th} position, it is embedded as:

$$\begin{bmatrix} R^{2\times 2}(m\alpha_1) & 0 & 0 & 0 \\ 0 & R^{2\times 2}(m\alpha_2) & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & R^{2\times 2}(m\alpha_{d/2}) \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_d \end{bmatrix},$$

where $R^{2\times 2}(m\alpha) = \begin{bmatrix} \cos(m\alpha) & -\sin(m\alpha) \\ \sin(m\alpha) & \cos(m\alpha) \end{bmatrix}$ is a 2 × 2 rotation matrix and the α_i are chosen as $\alpha_i = 10000^{\frac{2(i-1)}{d}}$. Thus, embedding of a positioned word is *equivariant*: when its position is translated, each pair of consecutive coordinates is rotated. Much like standard positional encodings for T, the frequencies of the rotations are chosen to exponentially vary.

An important takeaway is that this is distinct in character from previous examples, as the RoPe embeddings really are defined to be equivariant vectors, rather than linear maps. They still transform according to a group action, but are not themselves readily interpretable as matrices. Thus, embeddings such as RoPe (Su et al. (2021)), as well as Laplacian eigenvectors for points on manifolds (Rustamov (2007), described in the next subsection), are really equivariant maps from the input position space (e.g. the word location or the point on the manifold) into the vector space V of a group representation, rather than the actual matrices themselves.

Recall that a map $f : \mathcal{X} \to V$ is G-equivariant if $f(gx) = gf(x)$ for all $x \in \mathcal{X}, g \in G$, where we have implicitly assumed that g acts on V according to some representation. In the main text, we discussed two key advantages of using irreducible representation encodings: (1) equivariance and (2) capturing hierarchical resolution. Any map that is equivariant with respect to a unitary representation on V still satisfies (1)! Although (2) is not true for all equivariant maps, Laplacian eigenfunctions retain this property in the sense of minimizing the Dirichlet energy subject to orthogonality constraints (as noted in the main body). RoPe similarly retains this property to a degree, as a result of the particular representation of the special orthogonal group that is used, since it is very reminiscent of standard positional encodings for T.

We also note that group representations (in the sense of mappings between G and invertible matrices $GL(V)$), are a special case of equivariant maps. This is because the image of ρ itself is a vector space (consisting of matrices), and by definition, it is equivariant: a group representation ρ satisfies $\rho(gh) = \rho(g)\rho(h)$.

Learned embeddings / Embedding table. Another common type of positional encoding is fully learned embeddings (Gehring et al., 2017; Vaswani et al., 2017), which may also be referred to as the embedding table or lookup table approach. Here, each data point x is associated with a unique vector $\gamma(x) = w_x \in \mathbb{R}^d$ that is learned end-to-end with gradient descent, where we assume that the data domain $\mathcal{X}$ is finite, so that we can encounter each data point during training at least once. For instance, for encoding positions in text, one may store a vector $w_i \in \mathbb{R}^d$ for each $i \in \{1, \dots, 8192\}$, where 8192 is the largest sequence length that can be processed.

Fix an ordering $x_1, \dots, x_{|\mathcal{X}|}$ of the points in $\mathcal{X}$. By stacking the vectors w_i as columns of a matrix $W = [w_1, \dots, w_{|\mathcal{X}|}] \in \mathbb{R}^{d \times |\mathcal{X}|}$, we can view the embedding table as one-hot encoding each x_i to a vector in $\mathbf{1}_{x_i} \in \mathbb{R}^{|\mathcal{X}|}$, and then applying a learned linear transformation $\gamma(x_i) = w_{x_i} = W\mathbf{1}_{x_i}$. In other words, we can view the positional encoding as the one-hot encoding $\mathbf{1}_{x_i}$, and then we modify the model that processes the positional encoding by prepending a learned linear projection W to it. The one-hot encodings belong to a vector space of a group representation of the group of permutations $S_{|\mathcal{X}|}$, where $g \in \mathcal{X}$ acts naturally as $g \cdot \mathbf{1}_{x_i} = \mathbf{1}_{gx_i}$.

A.3. Laplacian Eigenspaces and Positional Encodings

Here, we follow the exposition of Tahmasebi & Jegelka (2023) to explain why the eigenspaces of the Laplace-Beltrami operator are vector spaces of group representations. We consider manifold data, but the discrete graph case is very similar. Consider a Riemannian manifold $\mathcal{M}$, and let G be any closed subgroup of its isometry group, i.e. the group of bijections $\tau : \mathcal{M} \to \mathcal{M}$ that preserve geodesic distances between pairs of points. Let $C^\infty(\mathcal{M})$ be the vector space of smooth real-valued functions from $\mathcal{M}$ to $\mathbb{R}$. The action of G on $f \in C^\infty(\mathcal{M})$ is given by $(g \cdot f)(x) = f(g^{-1}x)$. The Laplace-Beltrami operator is a linear operator $\Delta : C^\infty(\mathcal{M}) \to C^\infty(\mathcal{M})$. Let V_λ be an eigenspace of Δ, corresponding to eigenvalue $\lambda \in \mathbb{R}$. Then V_λ is a representation of G since Δ commutes with isometries, because for any eigenfunction ϕ of eigenvalue λ we have

$$\Delta(g \cdot \phi) = g \cdot \Delta(\phi) = \lambda(g \cdot \phi). \tag{6}$$

Now, suppose we have an orthonormal basis $\phi_1, \dots, \phi_k$ of an eigenspace V_λ. Define the positional encoding $\gamma(x) = [\phi_1(x), \dots, \phi_k(x)] \in \mathbb{R}^k$. We will show that this positional encoding is equivariant, i.e. $\gamma(gx) = \rho(g)\gamma(x)$ for some orthogonal $\rho(g) \in O(k)$. This is because $g \cdot \phi_1, \dots, g \cdot \phi_k$ is still an orthonormal basis of V_λ. Thus, there exists an orthogonal change of basis $\rho(g)$ such that $[g \cdot \phi_1, \dots, g \cdot \phi_k] = \rho(g)[\phi_1, \dots, \phi_k]$. Evaluating this at x, we have

$$\gamma(gx) = [g \cdot \phi_1(x), \dots, g \cdot \phi_k(x)] = \rho(g)[\phi_1(x), \dots, \phi_k(x)] = \rho(g)\gamma(x). \tag{7}$$

To see that ρ is a representation, note that for $g, h \in G$ we have $(gh) \cdot \phi = g \cdot (h \cdot \phi)$. Thus, it holds that

$$\begin{aligned} \rho(gh)[\phi_1, \dots, \phi_k] &= [(gh) \cdot \phi_1, \dots, (gh) \cdot \phi_k] && (8)\\ &= [g \cdot (h \cdot \phi_1), \dots, g \cdot (h \cdot \phi_k)] && (9)\\ &= \rho(g)[h \cdot \phi_1, \dots, h \cdot \phi_k] && (10)\\ &= \rho(g)\rho(h)[\phi_1, \dots, \phi_k]. && (11) \end{aligned}$$

Since the action on an orthonormal basis determines a linear map, we have $\rho(gh) = \rho(g)\rho(h)$.

Thus, Laplacian eigenfunction positional encodings also belong to a vector space of a group representation. The above argument can be extended to a positional encoding $\gamma(x) = [\phi_1(x), \dots, \phi_k(x)]$, where ϕ_i are orthonormal eigenvectors potentially belong to different eigenspaces. The only requirement is that if ϕ_i has eigenvalue λ, that there is an orthonormal basis for the eigenspace V_λ within $\phi_1, \dots, \phi_k$.

A.4. Transformers with Group Representation Positional Encodings

Consider again the form of the self-attention layer with positional encoding input, with $\gamma(\mathbf{x}) = [\gamma(x_1), \dots, \gamma(x_n)]$ and $\gamma(x_i) = \mathrm{vec}(\rho(x_i))$ for an orthogonal representation ρ.

$$f(\mathbf{x}) = W_V\, \gamma(\mathbf{x})\, \mathrm{softmax}\left(\gamma(\mathbf{x})^\top W_K^\top W_Q \gamma(\mathbf{x})\right), \tag{12}$$

and suppose that $W_V = b_1 I, W_K = b_2 I, W_Q = b_3 I$ for $b_i \in \mathbb{R}$ (in particular, the weight matrices are all square). We will show that the self-attention layer is equivariant, in the sense that $f(g\mathbf{x}) = (I \otimes \rho(g)) f(\mathbf{x})$.

First, note that the attention matrix is invariant, because

$$\begin{aligned}
\mathrm{softmax}\left(\gamma(g\mathbf{x})^\top W_K^\top W_Q \gamma(g\mathbf{x})\right) &= \mathrm{softmax}\left(b_2 b_3 \gamma(g\mathbf{x})^\top \gamma(g\mathbf{x})\right) && (13)\\
&= \mathrm{softmax}\left(b_2 b_3 \begin{bmatrix} \gamma(gx_1)^\top \gamma(gx_1) & \gamma(gx_1)^\top \gamma(gx_2) & \dots \\ \gamma(gx_2)^\top \gamma(gx_1) & \gamma(gx_2)^\top \gamma(gx_2) & \\ \vdots & \vdots & \ddots \end{bmatrix}\right) && (14)\\
&= \mathrm{softmax}\left(b_2 b_3 \begin{bmatrix} \gamma(x_1)^\top \gamma(x_1) & \gamma(x_1)^\top \gamma(x_2) & \dots \\ \gamma(x_2)^\top \gamma(x_1) & \gamma(x_2)^\top \gamma(x_2) & \\ \vdots & \vdots & \ddots \end{bmatrix}\right) && (15)\\
&= \mathrm{softmax}\left(\gamma(\mathbf{x})^\top W_K^\top W_Q \gamma(\mathbf{x})\right), && (16)
\end{aligned}$$

where in the second to last equality we used Lemma A.1 in each entry.

Next, we show that γ is equivariant using the Kronecker product trick:

$$\begin{aligned}
\gamma(g\mathbf{x}) &= [\mathrm{vec}(\rho(g)\rho(x_1)), \dots, \mathrm{vec}(\rho(g)\rho(x_n))] && (17)\\
&= [(I \otimes \rho(g))\mathrm{vec}(\rho(x_1)), \dots, (I \otimes \rho(g))\mathrm{vec}(\rho(x_n))] && (18)\\
&= (I \otimes \rho(g))\gamma(\mathbf{x}). && (19)
\end{aligned}$$

Finally, we can show equivariance of the self-attention layer f:

$$\begin{aligned}
f(g\mathbf{x}) &= W_V\, \gamma(g\mathbf{x})\, \mathrm{softmax}\left(\gamma(g\mathbf{x})^\top W_K^\top W_Q \gamma(g\mathbf{x})\right) && (20)\\
&= W_V\, \gamma(g\mathbf{x})\, \mathrm{softmax}\left(\gamma(\mathbf{x})^\top W_K^\top W_Q \gamma(\mathbf{x})\right) && (21)\\
&= b_1 I\, (I \otimes \rho(g))\gamma(\mathbf{x})\, \mathrm{softmax}\left(\gamma(\mathbf{x})^\top W_K^\top W_Q \gamma(\mathbf{x})\right) && (22)\\
&= (I \otimes \rho(g)) b_1 I \gamma(\mathbf{x})\, \mathrm{softmax}\left(\gamma(\mathbf{x})^\top W_K^\top W_Q \gamma(\mathbf{x})\right) && (23)\\
&= (I \otimes \rho(g)) f(\mathbf{x}). && (24)
\end{aligned}$$

A.5. Neural Tangent Kernel of MLPs with Group Representation Positional Encodings

Here, we demonstrate that the NTK of an MLP trained with positional encodings coming from an orthgonal group representation ρ is group invariant. As such, when trained on an equivariant task, the predictions on a test set are approximately equivariant in a sense that we elaborate on below. Our exposition is similar to that of Tancik et al. (2020).

Suppose we have data points $x_1, \dots, x_n$, with positional encodings $\gamma(x_i) = \mathrm{vec}(\rho(x_i))$ for some orthogonal group representation ρ, so the norm of $\gamma(x_i)$ is the same for each i. The NTK is then rotationally invariant, so it can be written as an inner product kernel, $K_{ij} = h(\gamma(x_i)^\top \gamma(x_j))$ for some $h : \mathbb{R} \to \mathbb{R}$. If we transform each x_i by $g \in G$, then the NTK K_{ij}^g for the dataset $gx_1, \dots, gx_n$ is given by

$$\begin{aligned}
K_{ij}^g &= h\left(\mathrm{vec}(\rho(gx_i))^\top \mathrm{vec}(\rho(gx_j))\right) && (25)\\
&= h\left(\mathrm{vec}(\rho(x_i))^\top \mathrm{vec}(\rho(x_j))\right) && (26)\\
&= K_{ij}, && (27)
\end{aligned}$$

where we used Lemma A.1 for the second equality. Thus, the NTK is rotationally invariant with respect to the training dataset. Now, suppose the labels $y_i \in \mathbb{R}^{d'}$ are G-equivariant functions of $x_i \in \mathbb{R}^{d_x}$, that is $y_i = f(x_i)$ for some f such that $f(gx) = \rho(g)f(x)$. Denote $\mathbf{x} = [x_1, \dots, x_n] \in \mathbb{R}^{d_x \times n}$, and let $f(\mathbf{x}) = [f(x_1), \dots, f(x_n)]$. Let $\mathbf{x}^{\text{test}} \in \mathbb{R}^{d_x \times n'}$ denote test points. Finally, let $K^{\text{test}} \in \mathbb{R}^{n \times n'}$ denote the kernel between train and test points, $K^{\text{test}}_{i,j} = \gamma(x_i)^\top \gamma(x^{\text{test}}_j)$. Under NTK evolution when trained on a mean-squared error loss, the outputs $f_{\theta_t}(\mathbf{x}^{\text{test}})$ at training time t can be approximated as follows (Lee et al., 2019):

$$f_{\theta_t}(\mathbf{x}^{\text{test}}) \approx f(\mathbf{x})(I - e^{-\eta K t})K^{-1}K^{\text{test}}. \tag{28}$$

Now, suppose we transform both the train and test data by g. Note that then $f(g\mathbf{x}) = \rho(g)f(\mathbf{x})$. Further, the train and test NTK matrices K and K^{test} are invariant. Thus, if θ^g_t denotes the parameters of the network at training time t on this new data, we have

$$f_{\theta^g_t}(g\mathbf{x}^{\text{test}}) \approx f(g\mathbf{x})(I - e^{-\eta K t})K^{-1}K^{\text{test}} = \rho(g)f(\mathbf{x})(I - e^{-\eta K t})K^{-1}K^{\text{test}} \approx \rho(g)f_{\theta_t}(\mathbf{x}^{\text{test}}). \tag{29}$$

Thus, we have the approximate equivariance $f_{\theta^g_t}(gx^{\text{test}}) \approx \rho(g)f_{\theta_t}(x^{\text{test}})$. Importantly, here we have to transform the training data of the MLP to get the equivariance property, which is slightly different than typical equivariance. This is because we are considering the NTK-approximated training dynamics of the MLP, as opposed to considering equivariance for any single function.

B. More Examples of Learned Implicit Neural Representations

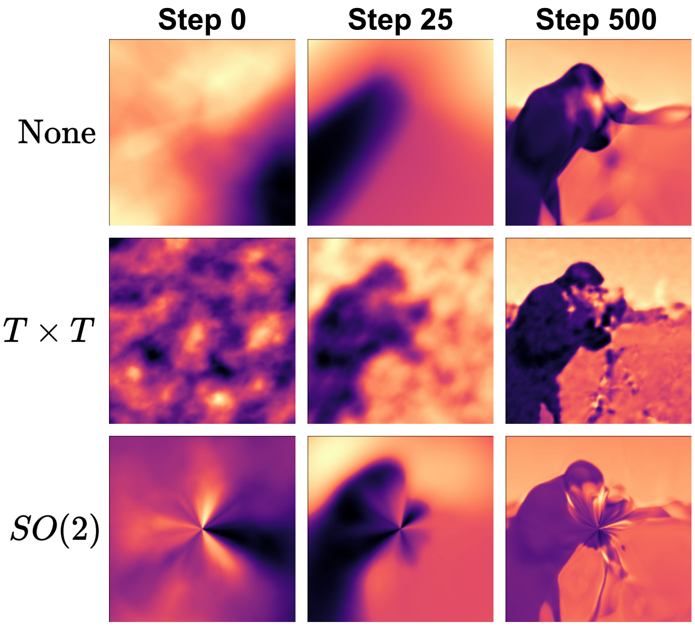

Figure 3. Learned implicit neural representations at different points in training for the Cameraman image. Each positional encoding uses its best hyperparameter settings from the sweeps for the experiments in Section 5.1 (which is why the model at initialization looks differently from that of Figure 2).

See Figure 3 for learned implicit neural representations at different points in training for the Cameraman image. We see that the equivariance bias of the $SO(2)$ is not helpful in this task. Nonetheless, it is able to ignore this equivariance bias during training, and learn a highly non-rotationally-invariant prediction.

Also, see Figure 4 for learned implicit neural representations on the vector field tasks. The equivariance bias is somewhat less visually evident here, but we can still see that early in training, the model with $SO(2)$ positional encodings learns a nearly rotationally-equivariant vector field for the Radial task.

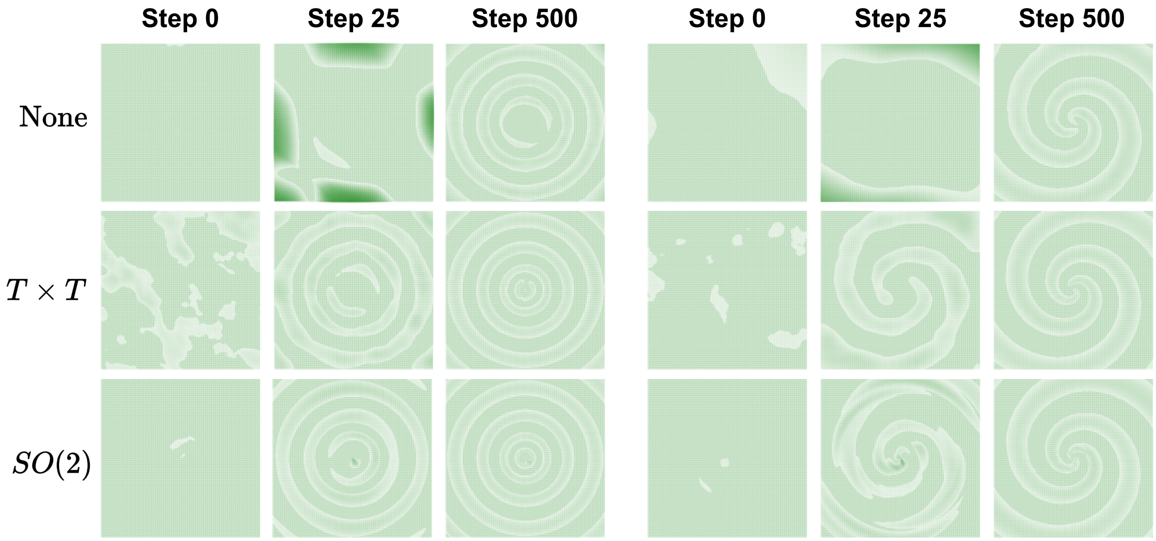

Figure 4. Learned implicit neural representations at different points in training for the (left) Radial and (right) Spiral vector fields.

C. Related work

Several works have proposed using Laplacian eigenvectors as positional encodings for general geometric domains (Koestler et al., 2022; Grattarola & Vandergheynst, 2022; Lim et al., 2023). Koestler et al. (2022) suggests using Laplace-Beltrami eigenfunctions to encode positions on manifolds, Grattarola & Vandergheynst (2022) propose using Laplacian eigenvectors of a suitable discrete graph associated to any non-Euclidean domain, and Lim et al. (2023) show that eigenvector-symmetry-invariant networks applied to Laplacian eigenvectors of graphs can approximate many previously proposed graph positional encodings. These works follow many prior works that use Laplacian eigenvectors for positional encodings or similar purposes, such as in geometry processing (Rustamov, 2007), graph neural networks / graph Transformers (Dwivedi & Bresson, 2020), and general point clouds with low-dimensional structure (Belkin & Niyogi, 2003).

D. Experimental Details

D.1. Data Details

Here, we give the exact definitions of the synthetic vector fields and images that we experiment on. We use polar coordinates (r, θ) instead of (x, y) for the definition, to give a more natural definition of $SO(2)$ equivariant or approximately $SO(2)$ equivariant vector fields. For the Radial vector field, the vector at location (r, θ) is:

$$v(r, \theta) = \sin(15\sqrt{r}) \begin{bmatrix} \cos(\theta) \\ \sin(\theta) \end{bmatrix}. \tag{30}$$

For the Spiral vector field, the vector at location (r, θ) is:

$$v(r, \theta) = \sin(30\sqrt{.1r} + \theta) \begin{bmatrix} \cos(\theta) \\ \sin(\theta) \end{bmatrix}. \tag{31}$$

The corresponding Radial image at pixel (x, y) has intensity value equal to the magnitude of the vector at (x, y) (so, $\sin(15\sqrt{r})$), and similarly for the Spiral image the value at (x, y) is $\sin(30\sqrt{.1r} + \theta)$.

D.2. Model and Positional Encoding Hyperparameters

$T \times T$ **encoding hyperparameters.** Similarly to Tancik et al. (2020), we sample the coefficients for the group representations from a Gaussian distribution with a chosen scale c. That is, our positional encodings are of the form $(x, y) \mapsto [\cos(\alpha_1^{(i)} x +$

$\alpha_2^{(i)} y), \ \sin(\alpha_1^{(i)} x + \alpha_2^{(i)} y)]_{i=1,\ldots,16}$, for $\alpha_j^{(i)} \sim \mathcal{N}(0, c^2)$. In experiments, we search for $c \in \{.1, 1, 3, 5, 10, 15, 20, 50\}$. We use a positional encoding dimension of 32.

$SO(2)$ **encoding hyperparameters.** Recall our form of the $SO(2)$ positional encodings: $(r, \theta) \mapsto \{J_m(cr)e^{ik\theta}\}_{m,c,k}$. In our experiments, we fix $m = 0$ (so, we only use the 0th Bessel function J_0). We sample the scale c uniformly within the range $[0, C]$ for some maximum value C, and uniformly sample integers k in the range $[1, K)$ for some maximum value K. In the experiments, we search over $C \in \{5, 25, 50\}$ and $K \in \{2, 4, 8\}$. Again, we use a positional encoding dimension of 32.

Model details. In all experiments, we use a multi-layer perceptron with hidden dimension 128, 3 hidden layers, and ReLU nonlinearities. The input dimension is 2 when not using positional encodings, and 32 when using positional encodings. The output dimension is 1 for the image task, and 2 for the vector field task.

Task setup and training details. For all experiments, the grid size is 256×256 (both for images and vector fields). We uniformly sample 5% of points on the grid as training points, then sample 40% for validation, and we test on the rest of the points. We train, validate, and test using the mean squared error reconstruction loss, meaning $L(\theta) = \frac{1}{|S|} \sum_{i \in S} (v(x_i, y_i) - f_\theta(\gamma(x_i, y_i)))^2$, where $v(x_i, y_i)$ is the value of the image or vector field at location (x_i, y_i), $\gamma(x_i, y_i)$ is the positional encoding, and S is either the training, validation, or test set. We train for 500 parameter update steps, where at each step we compute gradients with respect to all training points. We use the Adam optimizer with learning rate searched for in $\{.0001, .001, .01\}$; for the no positional encoding baseline we search over more learning rates: $\{.0001, .0005, .001, .005, .01, .05, .1\}$.

Appendix Citations

Belkin, M. and Niyogi, P. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural computation*, 15(6):1373–1396, 2003.

Dwivedi, V. P. and Bresson, X. A generalization of transformer networks to graphs. *arXiv preprint arXiv:2012.09699*, 2020.

Grattarola, D. and Vandergheynst, P. Generalised implicit neural representations. In Oh, A. H., Agarwal, A., Belgrave, D., and Cho, K. (eds.), *NeurIPS*, 2022.

Koestler, L., Grittner, D., Moeller, M., Cremers, D., and Lähner, Z. Intrinsic neural fields: Learning functions on manifolds. In *ECCV*, 2022.

Lee, J., Xiao, L., Schoenholz, S., Bahri, Y., Novak, R., Sohl-Dickstein, J., and Pennington, J. Wide neural networks of any depth evolve as linear models under gradient descent. *Advances in neural information processing systems*, 32, 2019.

Shaw, P., Uszkoreit, J., and Vaswani, A. Self-attention with relative position representations. *arXiv preprint arXiv:1803.02155*, 2018.

Tahmasebi, B. and Jegelka, S. The exact sample complexity gain from invariances for kernel regression on manifolds. *arXiv preprint arXiv:2303.14269*, 2023.