Parabolic Positional Encoding: A Vision-Centric, Principled, Extrapolatable, and General Approach

Christoffer Koo Øhrstrøm[1,2] Rafael I. Cabral Muchacho[3] Yifei Dong[3]
Filippos Moumtzidellis[1,2] Ronja Güldenring[1,2] Florian T. Pokorny[3] Lazaros Nalpantidis[1,2]

[1] Technical University of Denmark Denmark [2] Pioneer Centre for Artificial Intelligence Denmark
[3] KTH Royal Institute of Technology Sweden

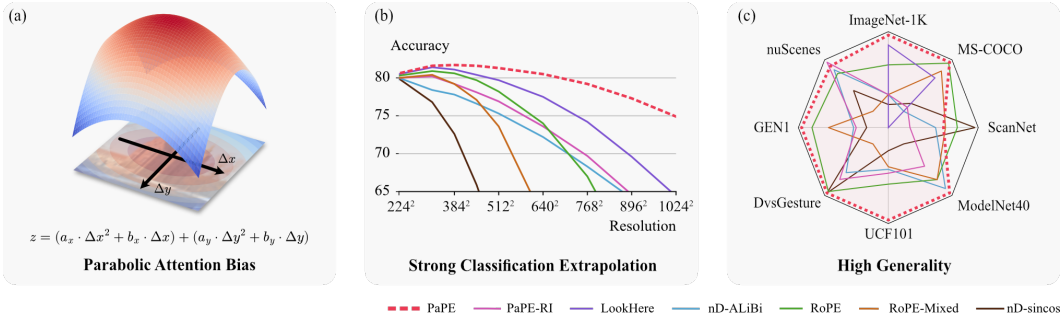

Figure 1: **Parabolic Positional Encoding (PaPE). (a)** PaPE encodes positions using an attention bias based on a sum of learnable parabolas with the relative position between tokens as the dependent variable. **(b)** PaPE has remarkable classification extrapolation, showing high robustness beyond the training resolution of 224^2. **(c)** PaPE is a general vision positional encoding—on 8 datasets that span 4 vision modalities, PaPE matches the best baseline on 5 datasets and exceeds all on 2 datasets.

Abstract

We propose Parabolic Positional Encoding (PaPE), a parabola-based positional encoding for vision modalities in attention-based architectures. Given a set of vision tokens—such as from videos, event camera streams, images, or point clouds—our objective is to encode their positions while accounting for the characteristics of vision modalities. Prior works have largely extended positional encodings from 1D-sequences in language to nD-structures in vision, but only with partial account of vision characteristics. We address this gap by designing PaPE from principles distilled from prior work: translation invariance, rotation invariance (PaPE-RI), distance decay, directionality, and context awareness. Extrapolation experiments on ImageNet-1K show how PaPE extrapolates remarkably well, improving in absolute terms by up to 10.5 over the next-best encoding. Generality experiments on 8 datasets across 4 modalities show that PaPE is a general vision positional encoding, as PaPE matches the best baseline on 5 datasets and exceeds all on 2 datasets. The code is available at https://github.com/DTU-PAS/parabolic-position-encoding and weights are available at [56].

1 Introduction & Related Works

In this work, we propose a positional encoding that is designed specifically for vision modalities.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

Transformers [43] are widely used for computer vision and robotics tasks. They have proven to be highly flexible, finding applications in several vision modalities such as videos [3, 6], event cameras [21, 37], images [18], and point clouds [51]. Despite their success in the vision domain, they tend to use positional encodings that are first designed for language [34, 41, 43] and later adapted for vision [19, 38, 48]. Although adaptations *can* account for vision-specific characteristics, we hypothesize that they are incomplete in their coverage of those, because some characteristics matter little in language. This leads us to identify relevant ideas in prior works and use these as guiding design principles for our proposed positional encoding.

Absolute positional encodings. Vaswani et al. [43] propose sinusoidal positional embeddings that are added directly to the input embeddings. These are extended from 1D to 2D by Wang and Liu [48] by using half of the position vector for one position dimension and the other half for the second position dimension. The extension to arbitrary dimensions follows naturally from this. While sinusoidal positional encoding handles multiple positional dimensions, its absolute nature lacks properties that RoPE-based and attention bias methods possess. However, it provides the exact location of each token, which can be relevant for some tasks, as we observe in Section 5.3.

RoPE-based methods. RoPE [41] stands out as a state-of-the-art positional encoding. It rotates queries and keys by a product of their position and a set of fixed frequencies, so the query-key dot product encodes the relative position, yielding *translation invariance*. One of the properties of RoPE is that attention strength diminishes as tokens move farther apart, a property we denote as *distance decay*. While vanilla RoPE operates purely in 1D, Axial RoPE [10] generalizes RoPE to arbitrary dimensionality by assigning disjoint subsets of the query and key vectors to each positional dimension and applying RoPE separately to each. However, because these dimensions are treated independently, Axial RoPE cannot naturally capture diagonal interactions. RoPE-Mixed [26], along with related approaches [33, 38, 53], introduce distinct learnable frequencies per positional dimension to imbue *directionality* into RoPE.

Extrapolation. More recently, *context-aware* variants of RoPE [44, 47] have shown that letting frequencies depend on each token improves the ability of language models to generalize to sequence lengths beyond what is seen during training. In vision, we also want models that train cheaply at low resolutions, yet seamlessly run inference at higher ones. Extrapolation reduces training costs and is an indicator of robustness. Moreover, strong extrapolation suggests that a model pretrained at low resolutions can be fine-tuned more easily to high-resolution target domains. Prior works [20, 26, 42] have shown that positional encoding greatly determines the extrapolation capabilities of a model.

Attention biases modify attention scores by adding a bias term directly into the attention matrix before the softmax. This bias depends on the relative positions of the query and key, reinforcing the importance of *translation invariance*. However, these bias terms often require materializing the full attention matrix, making them incompatible with efficient attention kernels [13, 14]. ALiBi [34], designed for autoregressive language models, subtracts a multiple of the relative distance between the tokens. This simple mechanism enforces both *translation invariance* and *distance decay* of attention scores. 2D-ALiBi [19] generalizes this idea to images by subtracting a multiple of the 2D Euclidean distance between positions, further endowing the encoding with *rotation invariance*. LookHere [20] extends 2D-ALiBi with *directionality* by restricting the field of view of each head to different directions. This adds *directionality*, but at the cost of losing *rotation invariance*.

From the prior works, we have identified the following set of guiding design principles that we consider to be important for positional encoding of vision modalities, and elaborate on in Section 3: *translation invariance*, *rotation invariance*, *distance decay*, *directionality*, and *context awareness*. To the best of our knowledge, no existing positional encoding is derived from all five principles. We use the principles to propose a novel positional encoding: **Pa**rabolic **P**ositional **E**ncoding (PaPE). PaPE treats the relative position between two tokens as the dependent variable in a sum of parabolas and uses this to encode positions through an attention bias. However, its formulation supports separate query-key transformations, which means PaPE does not have to materialize the full attention matrix. In addition, we propose a rotation invariant version—PaPE-RI. We summarize our contributions and claims as follows:

❶ **Parabolic Positional Encoding.** We propose PaPE and PaPE-RI: parabola-based positional encodings for vision modalities that are designed from the compiled principles.

❷ **Compatibility with Efficient Attention Kernels.** PaPE uses query-key transformations—similar to RoPE [41]—to be compatible with efficient attention kernels [13, 14].

❸ **Strong Classification Extrapolation.** PaPE is found to extrapolate remarkably well beyond its training resolution for classification, surpassing all baselines by at least 10.5 at resolution 1024^2.

❹ **High Generality.** We evaluate PaPE on 8 datasets across 4 vision modalities (videos, event cameras, images, and point clouds) and find that PaPE matches the best baseline in 5 datasets and exceeds all baselines in 2 datasets.

2 Preliminaries

Attention [43] for a collection of tokens $X = \{x_1, \cdots, x_n\}$ with $x_i \in \mathbb{R}^d$ is given as follows where queries, keys, and values are derived through learnable matrices $W_q, W_k, W_v \in \mathbb{R}^{h \times d}$, so that $q_i = W_q \cdot x_i$, $k_i = W_k \cdot x_i$, and $v_i = W_v \cdot x_i$:

$$A \cdot V = \text{softmax}\left(\frac{Q \cdot K^T}{\sqrt{h}}\right) \cdot V. \tag{1}$$

For clarity, we omit attention heads here and in the next section, noting that the extension to multi-head attention is straightforward. We define $S = Q \cdot K^T$ and obtain from Equation (1) that

$$S_{ij} = \langle q_i, k_j \rangle, \tag{2}$$

i.e., token similarity is measured as the dot products between all query-key pairs. Since attention is permutation invariant, we require positional information to encode information about the arrangement of tokens. To this end, we associate a position vector $r_i \in \mathbb{R}^p$ with each token, where p is the position dimensionality. For example, $p = 2$ for images (x, y) and $p = 3$ for point clouds (x, y, z). Our goal is to leverage these position vectors to inject positional information into the dot products.

3 Design Principles

We now describe and motivate the design principles that we identified in prior works. These principles are not a rigid checklist for the perfect positional encoding. They are a set of properties that we reason—and empirically evaluate in the experiments—are valuable in practice. We give mathematical definitions of the principles when applicable.[1]

Translation invariance matters because vision tasks usually depend on patterns defined by how parts relate to each other, not by where they are. A cat is a cat whether it appears in the top-left or bottom-right of an image. This calls for positional encodings that are invariant to global translations.

Definition 3.1. A function f is translation invariant if,

$$f(r_i + t, r_j + t) = f(r_i, r_j), \forall t, r_i, r_j \in \mathbb{R}^p. \tag{3}$$

Rotation invariance is valuable in settings where the orientation of an object should not affect the prediction. For instance, classifying a 3D object from a point cloud should not depend on how it is rotated in space. However, we do not treat rotation invariance as a universal requirement: orientation often carries key information about motion and action. We therefore consider rotation invariance as a task-dependent, special-case principle.

Definition 3.2. A function f is rotation invariant if,

$$f(Rr_i, Rr_j) = f(r_i, r_j), \forall r_i, r_j \in \mathbb{R}^p, \forall R \in \text{SO}(p). \tag{4}$$

Distance decay captures the intuition that nearby tokens should interact more strongly than distant ones. Specifically, attention between two tokens should decrease as their distance increases. This biases the positional encoding towards local interactions.

Directionality is the ability to modulate attention not just by how far tokens are from each other, but also by which direction. Unlike language, which is 1D and largely indifferent to geometric direction,

[1] We assume that token positions reside in a standard p-dimensional Euclidean space $\mathbb{R}^p$.

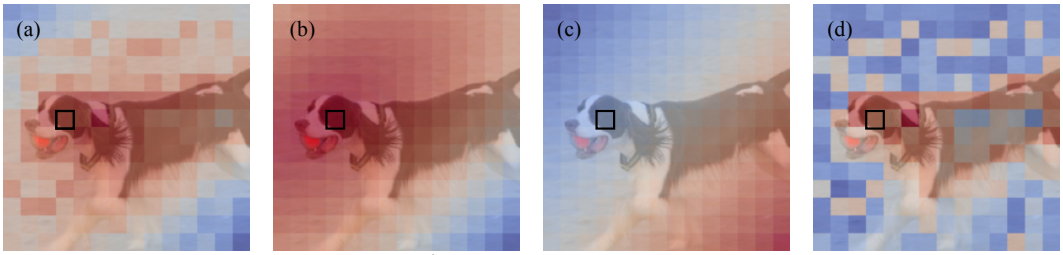

Figure 2: **Overview of Parabolic Positional Encoding (PaPE).** PaPE decomposes attention (a) into distance (b), direction (c), and semantics (d). Using the dog's eye as the query, PaPE learns to look in a bottom-right direction, while decaying attention with distance. The attention (a) is compatible with efficient attention kernels through separate query-key transformations. Colormap:

vision is inherently directional: above, below, left, right, and diagonal often carry distinct semantic roles. As we move to higher-dimensional settings, e.g. 3D and spatio-temporal data, the number of possible directions amplifies this effect. A good positional encoding for vision modalities should account for these directional cues.

Context awareness. Positional encodings must be capable of emphasizing local neighborhoods while also enabling long-range interactions. At first glance, these appear to be conflicting goals. Context awareness reconciles the two by letting the model adapt the decay and direction strength conditioned on the content of the token. Rather than enforcing a fixed decay and direction pattern, the model can decide when to emphasize local neighborhoods and when to focus on long-range interactions.

4 Parabolic Positional Encoding

We now introduce **Pa**rabolic **P**ositional **E**ncoding (PaPE) and its rotation invariant version, PaPE-RI. We begin by deriving the method and then illustrate how the design principles are incorporated into PaPE. Afterwards, we derive query-key transformations that ensure compatibility with efficient attention kernels.

A general form 1D parabola can be written as

$$y = ax^2 + bx + c. \tag{5}$$

Our goal is to reshape the attention score in Equation (2) into a sum of m such parabolas. To this end, we first define

$$\Delta r_{ij} = W_p \cdot (r_j - r_i), \tag{6}$$

where $W_p \in \mathbb{R}^{m\times p}$ is a learnable projection.[2] The components of Δr_{ij} serve as the dependent variable of the parabolas. We obtain the a and b coefficients directly from the token representation, x_i, using the learnable projections $W_a, W_b \in \mathbb{R}^{m\times d}$:

$$a_i = -\text{softplus}(W_a \cdot x_i) \tag{7}$$

$$b_i = W_b \cdot x_i, \tag{8}$$

where softplus followed by negation guarantees $a_i \in \mathbb{R}^m_{<0}$, making each parabola concave. We now present the main equation that describes PaPE, where $(\cdot)^{\odot 2}$ denotes the Hadamard square:

$$S_{ij} = \langle a_i, \Delta r_{ij}^{\odot 2}\rangle + \langle b_i, \Delta r_{ij}\rangle + \langle q_i, k_j\rangle. \tag{9}$$

Figure 2 visualizes this as a sum of three attention maps. Expanding the dot products makes the parabolic structure explicit: the attention score decomposes into a sum of m general form parabolas,

$$S_{ij} = \sum_{\ell=1}^{m} \left(\underbrace{a_{i\ell}\Delta r_{ij\ell}^2}_{ax^2} + \underbrace{b_{i\ell}\Delta r_{ij\ell}}_{bx} + \underbrace{\frac{\langle q_i, k_j\rangle}{m}}_{c} \right). \tag{10}$$

[2]While W_p is not strictly required to achieve our goal, our empirical results (see Section 5.6) show that it is beneficial, and we motivate it in Appendix E.

PaPE instantiates our design principles, as we will see next.

PaPE is translation invariant. This follows directly from modeling the relative position, Δr_{ij}, between pairs of tokens: $W_p((r_j + t) - (r_i + t)) = W_p(r_j - r_i), \forall t \in \mathbb{R}^p$.

PaPE-RI: a rotation invariant instantiation of PaPE. In the form given in Equation (9), PaPE is *not* rotation invariant. By imposing simple constraints on its parameters, we obtain a rotation invariant instantiation, which we call PaPE-RI. PaPE-RI is defined by setting all $b_i = 0$, choosing $a_{ik} = \alpha_i$ with $\alpha_i \in \mathbb{R}_{<0}$, and enforcing $W_p = w_p I_p \in \mathbb{R}^{p \times p}$ with $w_p \in \mathbb{R}$. Under these constraints, PaPE-RI becomes provably rotation invariant; we provide the formal proof in Appendix A.1. An implication of setting $b_i = 0$ is that PaPE-RI loses directionality.

PaPE has distance decay. Since a_i is constrained to be negative, it follows that the square terms penalize the distances as Δr_{ij} moves away from the origin.

PaPE is directional. Recall that the dot product between two vectors x, y is proportional to the cosine of the angle θ between them:

$$\langle x, y \rangle = \|x\| \|y\| \cos(\theta). \tag{11}$$

The dot product is maximized when $\theta = 0$ since $\cos(0) = 1$. Therefore, $\langle b_i, \Delta r_{ij} \rangle$ acts as a direction term and it is largest when Δr_{ij} is aligned with the direction defined by b_i.

PaPE has context awareness. The distance decay and direction are governed by a_i and b_i, respectively. Because both vectors are computed from the token content itself, Equations (7) and (8), PaPE naturally adapts to context instead of relying on a fixed pattern. This enables PaPE to support long-range interactions. By effectively turning off positional information—letting $a_i \to 0$ and setting $b_i = 0$—Equation (9) reduces to the standard attention score in Equation (2). In this regime, PaPE no longer depends on position, enabling long-range interactions.

PaPE decomposes attention. The decomposition in Equation (9) cleanly separates distinct contributions to the attention score: $\langle a_i, \Delta r_{ij}^{\odot 2} \rangle$ encodes *distance*, $\langle b_i, \Delta r_{ij} \rangle$ captures *direction*, and $\langle q_i, k_j \rangle$ is the familiar *semantic* term between tokens from Equation (2). Although this separation is not one of our design principles, it is a valuable property for model analysis, see Appendix F.

4.1 Compatibility with Efficient Attention Kernels

The formulation in Equation (9) is not directly compatible with efficient attention kernels such as FlashAttention [13, 14], because it explicitly depends on all pairwise relative positions. To recover compatibility, we draw inspiration from RoPE [41] and inject positional information directly into the dot product through separate position-aware transformations of the query and key. These transformations must ensure that the resulting dot product exactly recovers Equation (9). That is, to guarantee that $\langle f_q(q_i, r_i), f_k(k_j, r_j) \rangle =$

$$\langle a_i, \Delta r_{ij}^{\odot 2} \rangle + \langle b_i, \Delta r_{ij} \rangle + \langle q_i, k_j \rangle. \tag{12}$$

Crucially, this preserves the standard attention computation. Queries and keys are transformed independently using only absolute positions, and the attention kernel itself remains unchanged. As a result, PaPE becomes plug-and-play compatible with efficient attention kernels.

Let $\oplus$ denote vector concatenation, $\otimes$ outer product, $M_i = W_p^T \mathrm{diag}(a_i) W_p$, and $\mathrm{flatten}(X)$ be row-major flattening of a matrix into a vector. We can then transform the query as

$$f_q(q_i, r_i) = q_i \oplus r_i^T M_i r_i \oplus \mathrm{flatten}(M_i) \oplus -2 M_i r_i \oplus W_p^T b_i \oplus -\langle W_p^T b_i, r_i \rangle, \tag{13}$$

and the key as

$$f_k(k_j, r_j) = k_j \oplus 1 \oplus \mathrm{flatten}(r_j \otimes r_j) \oplus r_j \oplus r_j \oplus 1. \tag{14}$$

We prove in Appendix A.2 that these transformations exactly recover Equation (9).

This construction preserves compatibility with efficient attention kernels with only a small overhead: the dimensionality of queries and keys increases by $p^2 + 2p + 2$. This cost is moderate as p is typically small at $p = 2$ or $p = 3$, increasing head sizes by 10 and 17, respectively. We further measure and report efficiency in Appendix D.

Table 1: **PaPE extrapolates well beyond the training resolution.** Training resolution is 224^2 and evaluations are at resolutions up to 1024^2. No additional training or modifications are applied. **Best** and next-best. PaPE extrapolates better than the next-best encoding by up to 10.5.

Positional Encoding	224^2	320^2	384^2	448^2	512^2	640^2	768^2	896^2	1024^2
nD-sincos [48]	79.9	76.8	72.6	65.9	57.5	41.3	27.2	16.8	9.3
RoPE [41]	80.3	80.9	80.6	79.7	78.2	74.0	67.0	56.2	42.3
RoPE-Mixed [26]	80.0	80.4	79.2	77.1	73.6	61.5	43.4	27.1	17.1
nD-ALiBi [19]	80.0	78.4	77.8	76.6	75.3	72.2	68.3	64.1	59.1
LookHere [20]	80.5	81.4	81.1	80.4	79.7	77.5	74.2	69.6	64.4
PaPE (ours)	**80.6**	**81.6**	**81.7**	**81.6**	**81.3**	**80.5**	**79.2**	**77.3**	**74.9**
PaPE-RI (ours)	80.0	80.2	79.2	78.1	76.9	73.6	69.7	64.6	58.6

5 Experiments

We validate the main empirical claims by ❸ probing PaPE's ability to extrapolate in classification (Section 5.2), and ❹ testing its generality on 8 datasets across 4 vision modalities (Section 5.3). Beyond this, we strengthen the evaluation through parameter-matched experiments (Section 5.4), out-of-distribution generalization (Section 5.5), and an ablation study (Section 5.6).

5.1 Experimental Setup

Baselines. We compare to five representative baselines: nD-sincos [48] serves as a multi-dimensional generalization of the classic sinusoidal positional encoding of Vaswani et al. [43]. We use it to compare against an absolute positional encoding. RoPE [41] is a state-of-the-art and frequently used positional encoding, combining translation invariance with distance decay. We specifically adopt the axial RoPE variant [10] to handle multiple position dimensions. RoPE-Mixed [26] is included both because it is a common choice for adapting RoPE to vision and because it extends RoPE with directionality. nD-ALiBi [19] extends ALiBi [34] to multiple dimensions using Euclidean distances, thereby achieving translation invariance, rotation invariance, and distance decay. LookHere [20] imbues directionality to nD-ALiBi and loses rotation invariance in doing so. It delivers high classification accuracy and strong extrapolation. However, LookHere is only defined for images and does not admit a straightforward extension to other modalities, so we have to restrict it to image-based experiments. We note that Wang and Liu [48] in fact introduce a 2D variant of sinusoidal, and Fuller et al. [19] a 2D variant of ALiBi, but extending them to arbitrary dimensionality is straightforward.

Datasets. We evaluate PaPE on 8 datasets that span diverse vision modalities and tasks, showcasing its broad applicability. We consider three spatio-temporal datasets. UCF101 [40] contains videos for action recognition of 101 actions. DvsGesture [1] and GEN1 [15] are event camera datasets. Event streams are asynchronous, spatially sparse and temporally continuous, in contrast to synchronous, spatially dense, and temporally discrete videos. DvsGesture targets action recognition of 11 actions and has high spatial sparsity. GEN1 provides 39 hours of automotive driving for 2D object detection of cars and pedestrians and has medium spatial sparsity. Thus, we get a rich variety of spatio-temporal data types. ImageNet-1K [16, 36] offers 1M images for classification of 1K categories. COCO [31] provides more than 200K images for 2D object detection of 80 categories. Together, ImageNet-1K and COCO evaluate PaPE on large-scale image-based datasets. ScanNet [12] and ModelNet40 [52] are 3D point cloud datasets. ScanNet focuses on indoor scenes with widely distributed points and point-level semantic segmentation. ModelNet40 contains densely sampled object shapes and targets object-level classification. Finally, nuScenes [7] combines camera and LiDAR inputs for 3D object detection. We use nuScenes to test PaPE in a challenging multimodal real-world setting.

Models. The different modalities call for modality-specific Transformer variants. We choose to use standard models that stay close to the original Transformer architecture. For images, we adopt ViT [18] and equip it with a YOLOv10 [45] head for object detection. For point clouds, we rely on Point Transformer V3 [51], optimized for 3D reasoning. For videos, we use ViViT [3], a dedicated video vision transformer. For event cameras, we employ a ViT with spatio-temporal tokens obtained from Spiking Patches [57] and re-use the YOLOv10 head for object detection on GEN1. For nuScenes, we use UniTR [46], a unified multimodal transformer.

Table 2: **PaPE: a general vision positional encoding.** Comparison of PaPE and baselines across 8 diverse datasets. **Best** and next-best. mAP is measured at 0.5:0.95:0.05 IoU intervals. PaPE matches the best baseline on 5 of 8 datasets and exceeds all on 2 datasets (UCF101 and GEN1), demonstrating the broad usefulness of PaPE across multiple vision modalities and tasks.

Positional Encoding	Spatio-temporal			Image		Point cloud		Multimodal	Average
	UCF101 Acc.	DvsGesture Acc.	GEN1 mAP	ImageNet-1K Acc.	COCO mAP	ScanNet mIoU	ModelNet40 Acc.	nuScenes mAP	
nD-sincos [48]	38.7	**93.4**	27.3	79.9	34.7	**72.6**	92.2	67.4	63.3
RoPE [41]	43.9	93.1	33.6	80.3	38.8	71.7	93.0	68.3	65.3
RoPE-Mixed [26]	41.2	83.3	31.7	80.0	38.0	71.1	93.0	66.3	63.1
nD-ALiBi [19]	41.6	89.2	28.8	80.0	33.9	70.6	93.2	68.5	63.2
LookHere [20]	-	-	-	80.5	37.3	-	-	-	-
PaPE (ours)	**49.5**	**93.4**	**34.8**	**80.6**	**38.9**	71.0	**93.3**	68.7	**66.3**
PaPE-RI (ours)	42.2	90.6	28.5	80.0	34.3	69.3	92.7	**68.9**	63.3

Training. We implement all models in PyTorch [2] using 16-bit mixed precision and optimize them with AdamW [32]. All PaPE models are trained with $m = 50$. The choice of $m = 50$ was made in preliminary experiments by tuning it on the validation split of CIFAR-10 [28]. As such, m has not been specifically tuned to any of the datasets that we consider in the paper. Models are validated at every epoch, and we select the checkpoint with the highest validation score for final evaluation on the test set. All models are randomly initialized with uniform Kaiming initialization [24]. Crucially, for each dataset we fix the model size and all hyperparameters across positional encodings, so the only systematic difference between runs is the choice of positional encoding (up to training stochasticity). This setup yields a fair comparison of the positional encodings. See Appendix H for details.

5.2 How Well Does PaPE Extrapolate Beyond the Training Resolution? ❸

We assess PaPE's extrapolation ability (for classification) by evaluating how well it scales from its training resolution of 224^2 up to 1024^2 on ImageNet-1K. Importantly, we do not change the models in any way for evaluation at higher resolutions. Table 1 reports the raw results corresponding to the extrapolation plot shown in Figure 1b.

We find that PaPE extrapolates far better than any baseline, even outperforming LookHere, which is itself highly capable at extrapolating. PaPE increases accuracy by up to 1.1 for resolutions up to 512^2, and only falls below its training-resolution accuracy after 640^2. What is more, PaPE improves the accuracy by 10.5 over the next-best method (LookHere) at the highest resolution of 1024^2. Unlike the baselines, PaPE is context-aware. This aligns with the findings of Wang et al. [47] that adding a context-aware term significantly boosts RoPE's extrapolation in language models. Taken together, these results suggest that context awareness has a role to play in extrapolation.

In contrast, PaPE-RI does not share the strong extrapolation. This may be a consequence of the trade-off we made, sacrificing directionality to achieve rotation invariance. Similarly, LookHere can be seen as an nD-ALiBi variant, enhanced with directionality. In both cases, introducing directionality leads to better extrapolation, indicating that directionality is a key ingredient for extrapolation.

See additional details in Appendix C.2, including an experiment (Table 11) that confirms the superior extrapolation of PaPE, even when baselines are allowed to use position interpolation.

5.3 Is PaPE a General Vision Positional Encoding? ❹

Table 2 reports the results on the 8 datasets. PaPE achieves the highest average score of 66.3—surpassing the next-best method, RoPE, by a margin of 1 point (65.3 vs. 66.3). The score differences are small for some datasets; however, we still see that PaPE matches the best baseline on 5 datasets and clearly exceeds all baselines on 2 datasets, only underperforming on ScanNet. These results validate that PaPE is a general vision positional encoding. In contrast, PaPE-RI lags behind with a markedly lower average score of 63.3, highlighting the importance of directionality and the added flexibility in a_i and W_p. Breaking down the results reveals further insight:

PaPE excels on spatio-temporal data. On UCF101, PaPE delivers the largest absolute accuracy gain among all datasets: 49.5 vs. 43.9 for RoPE. We follow the official 3-fold cross-validation and report the mean accuracy (see Appendix C.5 for per-fold scores and significance test), highlighting

Table 3: **Parameter-matched.** Increasing the parameters of the baselines to match the additional parameter counts introduced by PaPE does not change the ranking reported in Table 2.

Dataset	nD-sincos	RoPE	RoPE-Mixed	nD-ALiBi	LookHere	PaPE (ours)	PaPE-RI (ours)
UCF101	39.5	<u>45.2</u>	41.3	41.9	-	**49.5**	41.8
DvsGesture	**93.4**	<u>93.1</u>	87.8	92.4	-	**93.4**	<u>93.1</u>
ImageNet-1K	79.6	80.2	80.1	80.0	<u>80.4</u>	**80.6**	79.8

the robustness of the result. A potential reason for PaPE's strong UCF101 performance is that many actions unfold very quickly, causing objects to move rapidly across frames—making directionality and context awareness matter greatly for understanding a fast moving object. PaPE also leads on event camera datasets, outperforming RoPE by 1.2 mAP on GEN1 and tying for top accuracy with sinusoidal encodings on DvsGesture. By contrast, RoPE-Mixed performs notably poorly in the spatio-temporal setting, especially on DvsGesture. These results underscore how challenging spatio-temporal reasoning is—and how non-trivial it is for models to capture these relations effectively.

PaPE is well-suited for images. On ImageNet-1K, PaPE has an accuracy of 80.6, matching that of LookHere at 80.5, and on COCO it likewise has a positive 0.1 difference in mAP compared to RoPE. While the differences are small, these results show PaPE's applicability to large-scale image datasets.

PaPE is capable of handling multiple modalities. PaPE-RI delivers the strongest performance on nuScenes with 68.9 mAP, trailed by PaPE at 68.7, with nD-ALiBi close behind at 68.5. This ranking is striking and points to an unexpectedly important role of rotation invariance in multimodal processing. Mirroring our findings on spatio-temporal data, RoPE-Mixed again lags behind in this setting, showing how challenging it is to learn multimodal positional structure.

When translation invariance matters. nD-sincos ranks as the lowest performing encoding on 4 of 8 datasets and trails as second-lowest on 2 of 8 datasets. In particular, it consistently lags behind on object detection tasks (COCO, GEN1, and nuScenes). While there are a few exceptions (ScanNet and DvsGesture), the pattern is clear: the results suggest that translation invariance is often crucial, as nD-sincos is the only encoding in our study that lacks it.

When absolute positional encoding matters. The ScanNet results reveal additional findings. It is the only dataset where PaPE trails the baselines. What is more, nD-sincos surpasses all others, including RoPE by 0.9 mIoU, suggesting that absolute positional information is important for ScanNet. A potential explanation is that everyday household objects have characteristic sizes. For example, a table typically sits about $1\,\mathrm{m}$ above the floor. Purely relative encodings may fail to capture such relationships when the context in Point Transformer V3 does not span sufficiently distant points. This explanation is supported by ModelNet40, where the translation invariant encodings surpass sinusoidal encodings by at least 0.8 accuracy. Here, each sample is a single object with no surrounding scene, so absolute position carries less semantic weight. In this setting, all methods achieve similar accuracy, yet PaPE remains the top performer, reinforcing it as a robust positional encoding.

Rotation invariance matters less than hypothesized. Our point cloud experiments show that rotation invariant positional encodings do not improve over rotation variant alternatives. This is evident from how neither rotation invariant encoding (nD-ALiBi and PaPE-RI) manages to outperform the rotation-variant encodings. One possible explanation is that the rotation invariant positional encodings trade strict rotation invariance for less representation flexibility, and that decreased flexibility matters more than what they gain from invariance [49].

5.4 Does PaPE's Performance Stem from Additional Parameters?

PaPE increases the parameter count through W_p, W_a, and W_b, raising the question of whether its performance simply reflects greater model capacity. We answer this using UCF101, DvsGesture, and ImageNet-1K by matching the baselines' parameter counts to PaPE's (see Appendix C.4 for details). Notably, this comparison favors the baselines, since whereas PaPE's extra parameters influence only positional terms, the matched baselines necessarily gain parameters that directly influence semantic representations as well. Despite this, Table 3 reports that matching parameters does not change the outcome observed in Table 2: PaPE remains the highest scoring on UCF101 and ImageNet-1K, and tied with nD-sincos on DvsGesture, supporting PaPE as a strong positional encoding. In addition, we report results on reducing non-PaPE parameters as well as extrapolation under parameter-matched

Table 4: **Out-of-distribution generalization.** PaPE does not degrade more than the baselines when evaluated on out-of-distribution versions of ImageNet.

Dataset	nD-sincos	RoPE	RoPE-Mixed	nD-ALiBi	LookHere	PaPE (ours)	PaPE-RI (ours)
ImageNetV2 [35]	68.3	**69.2**	68.2	68.1	<u>68.9</u>	**69.2**	68.8
ImageNet-Renditions [25]	27.0	<u>29.2</u>	28.2	27.2	28.6	**29.4**	26.8
ObjectNet [5]	28.9	<u>31.0</u>	28.5	29.2	29.8	**31.1**	29.4

Table 5: **Ablations on ImageNet-1K.** All components of PaPE contribute to its accuracy as removing any component results in lower accuracy at the training resolution of 224^2. W_p does not influence the extrapolation performance, but we observe that all other components affect it, with distance decay (see $\langle a_i, \Delta r_{ij}^{\odot 2} \rangle$ and a_i activation) particularly driving much of the high extrapolation.

Ablation	224^2	320^2	384^2	448^2	512^2	640^2	768^2	896^2	1024^2
PaPE	79.2	80.3	80.1	79.5	78.9	77.5	75.4	72.9	69.8
- $\langle a_i, \Delta r_{ij}^{\odot 2} \rangle$	77.2	77.4	75.5	72.2	67.9	57.7	47.0	37.7	30.1
- $\langle b_i, \Delta r_{ij} \rangle$	78.7	79.3	78.5	77.4	76.1	73.1	69.5	64.9	59.5
- W_a, W_b	78.4	78.9	78.4	77.6	76.6	74.2	71.4	68.1	64.0
- a_i activation	78.6	79.7	79.4	78.4	76.9	71.9	63.5	50.7	36.0
- W_p	78.9	79.9	79.8	79.4	78.7	77.2	75.4	72.7	69.5

settings in Appendix C.4. PaPE maintains its advantage in all cases, further supporting that its high performance is due to improved positional encoding.

5.5 Does PaPE Harm Out-of-distribution Generalization?

PaPE is context-aware through W_a and W_b, making the positional encoding dependent on the feature space, and so raising the question if PaPE is harmful on out-of-distribution samples. To assess this, we use the models trained on ImageNet-1K and evaluate them without additional training on ImageNetV2 [35], ImageNet-Renditions [25], and ObjectNet [5]. ImageNetV2 has 10K images that are collected a decade after ImageNet. ImageNet-Renditions has 30K image renditions (art, cartoons, sketches, etc.) of 200 classes from ImageNet-1K. ObjectNet has 50K images of objects with varying rotations, backgrounds, and viewpoints. ImageNetV2 seeks to stay close to the ImageNet domain whereas ImageNet-Renditions and ObjectNet focus on sampling from different input domains. Table 4 reports that PaPE is tied with RoPE as best performing on ImageNetV2 and it is the best performing on ImageNet-Renditions. The differences in accuracy are small, but they confirm that PaPE does not degrade more than the baselines on out-of-distribution datasets. We do not find PaPE to be harmful to out-of-distribution generalization. We further confirm this for extrapolation in Appendix C.3.

5.6 Ablation Study

We systematically ablate each component of PaPE to understand its contribution and to validate the instantiations of distance decay, directionality, and context awareness. To ablate context awareness,

Table 6: **Ablation efficiency.** Measured on an RTX 4090 with a batch size of 1. See Appendix D for detailed setup. Context awareness (W_a, W_b) and W_p are the main contributors of PaPE's moderate overhead. Together with the results in Table 5, this shows that we can remove W_p to achieve high extrapolation without the moderate overhead.

		Training			Inference		
Ablation	Params. (M)	VRAM (MB)	GFLOPS	Time (ms)	VRAM (MB)	GFLOPS	Time (ms)
PaPE	97.5	1901	119.7	14.6	433	37.6	1.6
- $\langle a_i, \Delta r_{ij}^{\odot 2} \rangle$	92.0	1791	112.8	13.7	388	35.3	1.5
- $\langle b_i, \Delta r_{ij} \rangle$	92.0	1791	112.8	13.8	388	37.4	1.5
- W_a, W_b	86.4	1673	106.6	12.9	363	35.3	1.4
- a_i activation	97.5	1901	119.7	14.6	433	37.6	1.6
- W_p	86.9	1679	106.7	13.1	365	35.4	1.5

Table 7: **Ablation of m on ImageNet-1K.** PaPE obtains the highest accuracy at $m = 50$. The differences are small and a low m may be preferable for increased efficiency.

m	2	4	8	16	32	50	64
Accuracy	78.7	78.4	79.0	79.1	78.9	79.2	79.0

we remove W_a and W_b and instead treat a_i and b_i as learned parameters, where the softplus activation is still applied to the a coefficients. They are learned distinctly for all layers and heads, but are shared between tokens such that $a_i = a_j$ and $b_i = b_j$ for any pair of i and j. This way, we maintain distance decay and directionality without context awareness. To keep the study computationally feasible, we conduct all ablations on ImageNet-1K using a ViT-S instead of a ViT-B model size, and train for 150 epochs rather than 300. Table 5 reports the ablation results.

Ablation of principle instantiations. We see that PaPE has an accuracy of 79.2 and that all ablations result in lower accuracy. Noticeably, removing any of the three instantiations of the design principles—distance decay ($\langle a_i, \Delta r_{ij}^{\odot 2} \rangle$), directionality ($\langle b_i, \Delta r_{ij} \rangle$), and context awareness (W_a, W_b)—causes an accuracy drop between 0.5 and 2.0 at the 224^2 training resolution and from 5.8 to 39.7 at 1024^2, validating the relevance of instantiating those principles in PaPE. The ablations show that distance decay is a significant reason behind PaPE's extrapolation performance, as the accuracy drops by 39.7 points without it. Distance decay is further validated by how the model struggles to use the squared distance term when removing the softplus activation, resulting in accuracy decreases of 0.6 at 224^2 and 33.8 at 1024^2. Directionality and context awareness also contribute to extrapolation as we observe respective drops in accuracy of 10.3 and 5.8 at 1024^2 without them.

Ablation of W_p and m. Although its impact is modest at a 0.3 difference in accuracy, adding W_p contributes positively to PaPE. W_p does not influence extrapolation since the accuracy drops at all resolutions range from 0 to 0.4 without W_p, aligned with the training drop of 0.3. We also report ablation efficiency in Table 6 and find that W_p is a significant contributor to PaPE's moderate overhead. This matters because we can remove W_p to reduce the overhead significantly while preserving the extrapolation performance. Table 7 further confirms the choice of $m = 50$ as it obtains the highest accuracy of 79.2. We note though that the difference is small and that decreasing m improves the efficiency. In practice, $m = 8$ provides a good trade-off between accuracy and efficiency, evident from the efficiency reported for various m in Appendix D. Additionally, Appendix C.6 shows that m has low influence on extrapolation, meaning the choice of m is of little importance in this regard.

6 Limitations & Future Work

PaPE's main limitation is its moderate overhead (Appendix D), dependent on m. Future work will focus on how to lower m in practice, or on removing W_p altogether and thereby keeping $m = p$, which is small for all modalities and tasks considered here. Additionally, we want to integrate PaPE directly into efficient attention kernels to further decrease the overhead. Finally, an interesting direction is to explore parameter-efficient fine-tuning (PEFT) using the PaPE parameters for position-based PEFT.

7 Conclusions

We introduce Parabolic Positional Encoding (PaPE), a principled, vision-centric positional encoding. PaPE, along with its rotation invariant version PaPE-RI, is derived from principles identified in prior research and is compatible with efficient attention kernels. The experiments demonstrate that PaPE exhibits exceptional classification extrapolation capabilities, substantially outperforming all baselines by at least 10.5 at resolution 1024^2. We evaluate PaPE on 8 datasets spanning videos, event cameras, images, point clouds, and multimodal settings. The results validate PaPE as a general vision positional encoding: it raises the average score by 1 over RoPE while matching the best baseline on 5 datasets and exceeding all baselines on 2 datasets. Thus, PaPE paves a new path for vision positional encoding, and we anticipate that its underlying principles will motivate future research.

Acknowledgments and Disclosure of Funding

This work has been supported by Innovation Fund Denmark through the project "Safety and Autonomy for Vehicles in Agriculture (SAVA)", 2105-00013A.

References

[1] A. Amir, B. Taba, D. Berg, T. Melano, J. McKinstry, C. Di Nolfo, T. Nayak, A. Andreopoulos, G. Garreau, M. Mendoza, J. Kusnitz, M. Debole, S. Esser, T. Delbruck, M. Flickner, and D. Modha. A Low Power, Fully Event-Based Gesture Recognition System. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7243–7252, 2017.

[2] J. Ansel, E. Yang, H. He, N. Gimelshein, A. Jain, M. Voznesensky, B. Bao, P. Bell, D. Berard, E. Burovski, G. Chauhan, A. Chourdia, W. Constable, A. Desmaison, Z. DeVito, E. Ellison, W. Feng, J. Gong, M. Gschwind, B. Hirsh, S. Huang, K. Kalambarkar, L. Kirsch, M. Lazos, M. Lezcano, Y. Liang, J. Liang, Y. Lu, CK. Luk, B. Maher, Y. Pan, C. Puhrsch, M. Reso, M. Saroufim, M. Y. Siraichi, H. Suk, M. Suo, P. Tillet, E. Wang, X. Wang, W. Wen, S. Zhang, X. Zhao, K. Zhou, R. Zou, A. Mathews, G. Chanan, P. Wu, and S. Chintala. PyTorch 2: Faster machine learning through dynamic python bytecode transformation and graph compilation. In *29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2 (ASPLOS '24)*. ACM, Apr. 2024. doi: 10.1145/3620665.3640366.

[3] A. Arnab, M. Dehghani, G. Heigold, C. Sun, M. Lučić, and C. Schmid. ViViT: A Video Vision Transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6836–6846, 2021.

[4] F. Barbero, A. Vitvitskyi, C. Perivolaropoulos, R. Pascanu, and P. Veličković. Round and Round We Go! What makes Rotary Positional Encodings useful? In *The Thirteenth International Conference on Learning Representations*, 2025.

[5] A. Barbu, D. Mayo, J. Alverio, W. Luo, C. Wang, D. Gutfreund, J. Tenenbaum, and B. Katz. ObjectNet: A large-scale bias-controlled dataset for pushing the limits of object recognition models. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL `https://proceedings.neurips.cc/paper_files/paper/2019/hash/97af07a14cacba681feacf3012730892-Abstract.html`.

[6] G. Bertasius, H. Wang, and L. Torresani. Is Space-Time Attention All You Need for Video Understanding? In *Proceedings of the 38th International Conference on Machine Learning*, volume 139, pages 813–824. PMLR, July 2021.

[7] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom. nuScenes: A Multimodal Dataset for Autonomous Driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11621–11631, 2020.

[8] S. Chen, S. Wong, L. Chen, and Y. Tian. Extending Context Window of Large Language Models via Positional Interpolation, June 2023.

[9] X. Chu, Z. Tian, B. Zhang, X. Wang, and C. Shen. Conditional Positional Encodings for Vision Transformers. In *The Eleventh International Conference on Learning Representations*, Sept. 2022. URL `https://openreview.net/forum?id=3KWnuT-R1bh`.

[10] X. Chu, J. Su, B. Zhang, and C. Shen. VisionLLaMA: A Unified LLaMA Backbone for Vision Tasks. In *European Conference on Computer Vision*, pages 1–18. Springer, 2024. ISBN 978-3-031-72848-8.

[11] E. D. Cubuk, B. Zoph, J. Shlens, and Q. V. Le. Randaugment: Practical Automated Data Augmentation With a Reduced Search Space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 702–703, 2020.

[12] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Niessner. ScanNet: Richly-Annotated 3D Reconstructions of Indoor Scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5828–5839, 2017.

[13] T. Dao. FlashAttention-2: Faster Attention with Better Parallelism and Work Partitioning. In *The Twelfth International Conference on Learning Representations*, 2024.

[14] T. Dao, D. Fu, S. Ermon, A. Rudra, and C. Ré. FlashAttention: Fast and memory-efficient exact attention with IO-awareness. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 16344–16359. Curran Associates, Inc., 2022.

[15] P. de Tournemire, D. Nitti, E. Perot, D. Migliore, and A. Sironi. A Large Scale Event-based Detection Dataset for Automotive, Jan. 2020.

[16] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255. IEEE, 2009.

[17] Y. Ding, L. L. Zhang, C. Zhang, Y. Xu, N. Shang, J. Xu, F. Yang, and M. Yang. LongRoPE: Extending LLM Context Window Beyond 2 Million Tokens. In *Forty-First International Conference on Machine Learning*, June 2024.

[18] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representations*, Oct. 2020.

[19] A. Fuller, K. Millard, and J. Green. CROMA: Remote Sensing Representations with Contrastive Radar-Optical Masked Autoencoders. *Advances in Neural Information Processing Systems*, 36: 5506–5538, Dec. 2023.

[20] A. Fuller, D. G. Kyrollos, Y. Yassin, and J. R. Green. LookHere: Vision Transformers with Directed Attention Generalize and Extrapolate. *Advances in Neural Information Processing Systems*, 37:19683–19739, Dec. 2024.

[21] M. Gehrig and D. Scaramuzza. Recurrent Vision Transformers for Object Detection with Event Cameras. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13884–13893, June 2023.

[22] G. Ghiasi, Y. Cui, A. Srinivas, R. Qian, T.-Y. Lin, E. D. Cubuk, Q. V. Le, and B. Zoph. Simple copy-paste is a strong data augmentation method for instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2918–2928, June 2021.

[23] F. Gu, W. Sng, X. Hu, and F. Yu. EventDrop: Data augmentation for event-based learning. In *30th International Joint Conference on Artificial Intelligence, IJCAI 2021*, June 2021.

[24] K. He, X. Zhang, S. Ren, and J. Sun. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1026–1034, 2015.

[25] D. Hendrycks, S. Basart, N. Mu, S. Kadavath, F. Wang, E. Dorundo, R. Desai, T. Zhu, S. Parajuli, M. Guo, D. Song, J. Steinhardt, and J. Gilmer. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8340–8349, October 2021.

[26] B. Heo, S. Park, D. Han, and S. Yun. Rotary Position Embedding for Vision Transformer. In *European Conference on Computer Vision*, pages 289–305. Springer Nature Switzerland, 2024. ISBN 978-3-031-72684-2. doi: 10.1007/978-3-031-72684-2_17.

[27] D. P. Kingma and J. L. Ba. Adam: A Method for Stochastic Optimization. In *3rd International Conference for Learning Representations*, San Diego, 2015.

[28] A. Krizhevsky. Learning multiple layers of features from tiny images. *Master's thesis, University of Toronto*, 2009.

[29] Y. Li, Y. Kim, H. Park, T. Geller, and P. Panda. Neuromorphic Data Augmentation for Training Spiking Neural Networks. In *European Conference on Computer Vision*, 2022.

[30] Y. Li, H. Mao, R. Girshick, and K. He. Exploring Plain Vision Transformer Backbones for Object Detection. In *European Conference on Computer Vision*, June 2022.

[31] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft COCO: Common Objects in Context. In *European Conference on Computer Vision*, pages 740–755. Springer International Publishing, 2014. ISBN 978-3-319-10602-1. doi: 10.1007/978-3-319-10602-1_48.

[32] I. Loshchilov and F. Hutter. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations*, 2019.

[33] S. Ostmeier, B. Axelrod, M. Varma, M. E. Moseley, A. Chaudhari, and C. Langlotz. LieRE: Lie Rotational Positional Encodings. In *Forty-Second International Conference on Machine Learning*, 2025.

[34] O. Press, N. Smith, and M. Lewis. Train Short, Test Long: Attention with Linear Biases Enables Input Length Extrapolation. In *International Conference on Learning Representations*, Oct. 2021.

[35] B. Recht, R. Roelofs, L. Schmidt, and V. Shankar. Do ImageNet classifiers generalize to ImageNet? In K. Chaudhuri and R. Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 5389–5400. PMLR, 09–15 Jun 2019.

[36] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115(3):211–252, Dec. 2015. ISSN 0920-5691, 1573-1405. doi: 10.1007/s11263-015-0816-y.

[37] A. Sabater, L. Montesano, and A. C. Murillo. Event Transformer. A Sparse-Aware Solution for Efficient Event Data Processing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2677–2686, 2022.

[38] C. Schenck, I. Reid, M. G. Jacob, A. Bewley, J. Ainslie, D. Rendleman, D. Jain, M. Sharma, K. A. Dubey, A. Wahid, S. Singh, R. Wagner, T. Ding, C. Fu, A. Byravan, J. Varley, A. A. Gritsenko, M. Minderer, D. Kalashnikov, J. Tompson, V. Sindhwani, and K. M. Choromanski. Learning the RoPEs: Better 2D and 3D Position Encodings with STRING. In *Forty-Second International Conference on Machine Learning*, June 2025.

[39] L. N. Smith and N. Topin. Super-convergence: Very fast training of neural networks using large learning rates. In *Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications*, volume 11006, pages 369–386. SPIE, May 2019. doi: 10.1117/12.2520589.

[40] K. Soomro, A. R. Zamir, and M. Shah. UCF101: A Dataset of 101 Human Actions Classes From Videos in The Wild, Dec. 2012.

[41] J. Su, M. Ahmed, Y. Lu, S. Pan, W. Bo, and Y. Liu. RoFormer: Enhanced transformer with Rotary Position Embedding. *Neurocomputing*, 568:127063, Feb. 2024. ISSN 0925-2312. doi: 10.1016/j.neucom.2023.127063.

[42] Y. Sun, L. Dong, B. Patra, S. Ma, S. Huang, A. Benhaim, V. Chaudhary, X. Song, and F. Wei. A Length-Extrapolatable Transformer. In A. Rogers, J. Boyd-Graber, and N. Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14590–14604, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.816.

[43] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is All you Need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[44] A. Veisi, D. Fartoot, and H. Amirzadeh. Context-aware Rotary Position Embedding, July 2025.

[45] A. Wang, H. Chen, L. Liu, K. Chen, Z. Lin, J. Han, and G. Ding. YOLOv10: Real-Time End-to-End Object Detection. *Advances in Neural Information Processing Systems*, 37:107984–108011, Dec. 2024.

[46] H. Wang, H. Tang, S. Shi, A. Li, Z. Li, B. Schiele, and L. Wang. UniTR: A Unified and Efficient Multi-Modal Transformer for Bird's-Eye-View Representation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6792–6802, 2023.

[47] Y. Wang, S. Shen, R. Munos, H. Zhan, and Y. Tian. Positional Encoding via Token-Aware Phase Attention, Sept. 2025.

[48] Z. Wang and J.-C. Liu. Translating math formula images to LaTeX sequences using deep neural networks with sequence-level training. *International Journal on Document Analysis and Recognition (IJDAR)*, 24(1):63–75, June 2021. ISSN 1433-2825. doi: 10.1007/s10032-020-00360-2.

[49] A. G. Wilson. Position: Deep learning is not so mysterious or different. In *Forty-second International Conference on Machine Learning Position Paper Track*, 2025.

[50] H. Wu, B. Xiao, N. Codella, M. Liu, X. Dai, L. Yuan, and L. Zhang. CvT: Introducing Convolutions to Vision Transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22–31, 2021. URL `https://openaccess.thecvf.com/content/ICCV2021/html/Wu_CvT_Introducing_Convolutions_to_Vision_Transformers_ICCV_2021_paper.html`.

[51] X. Wu, L. Jiang, P.-S. Wang, Z. Liu, X. Liu, Y. Qiao, W. Ouyang, T. He, and H. Zhao. Point Transformer V3: Simpler, Faster, Stronger. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4840–4851, June 2024.

[52] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao. 3D ShapeNets: A Deep Representation for Volumetric Shapes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1912–1920, 2015.

[53] H. Yu, T. Jiang, S. Jia, S. Yan, S. Liu, H. Qian, G. Li, S. Dong, and C. Yuan. ComRoPE: Scalable and Robust Rotary Position Embedding Parameterized by Trainable Commuting Angle Matrices. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 4508–4517, 2025.

[54] S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, and Y. Yoo. CutMix: Regularization Strategy to Train Strong Classifiers With Localizable Features. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6023–6032, 2019.

[55] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz. Mixup: Beyond Empirical Risk Minimization. In *International Conference on Learning Representations*, 2018.

[56] C. K. Øhrstrøm. Model weights for "Parabolic Positional Encoding: A Vision-Centric, Principled, Extrapolatable, and General Approach". *Technical University of Denmark. Online resource.*, 2026. doi: 10.11583/DTU.32076279. URL `https://data.dtu.dk/articles/online_resource/Model_weights_for_Where_to_Attend_A_Principled_Vision-Centric_Position_Encoding_with_Parabolas_/32076279`.

[57] C. K. Øhrstrøm, R. Güldenring, and L. Nalpantidis. Spiking Patches: Asynchronous, Sparse, and Efficient Tokens for Event Cameras. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2026.

A Proofs

A.1 PaPE-RI is Rotation Invariant

By the definition of PaPE-RI, assume $b_i = 0$, $a_{i\ell} = \alpha_i \in \mathbb{R}_{<0}$, and $W_p = w_p I_p$ with $w_p \in \mathbb{R}$, where I_p represents the identity matrix of shape $p \times p$.

Let $\Delta r_{ij} = W_p \cdot (r_j - r_i)$ and $\Delta r'_{ij} = W_p \cdot (Rr_j - Rr_i)$ for $R \in \mathrm{SO}(p)$.

In this proof, we show that restricting the model to the conditions assumed above guarantees that $A_{ij}{=}A'_{ij}$, where entries A_{ij} depend on Δr_{ij}, and entries A'_{ij} depend respectively on $\Delta r'_{ij}$. This satisfies Definition 3.2.

Inserting the conditions in the position-dependent terms leads to the simplified expression,

$$\left(\sum_{\ell=1}^{m} a_{i\ell}\Delta r_{ij\ell}^2 + b_{i\ell}\Delta r_{ij\ell}\right) \tag{15}$$

$$= \sum_{\ell=1}^{m} a_{i\ell}\Delta r_{ij\ell}^2 \quad \text{using} \quad b_{i\ell} = 0 \tag{16}$$

$$= \alpha_i \sum_{\ell=1}^{m} \Delta r_{ij\ell}^2 \quad \text{using} \quad a_{i\ell} = \alpha_i \tag{17}$$

$$= \alpha_i \Delta r_{ij}^T \Delta r_{ij}. \tag{18}$$

The equivalence of the position-dependent terms ($\Delta r_{ij}'^T \Delta r'_{ij} = \Delta r_{ij}^T \Delta r_{ij}$) proves the rotation invariance of PaPE-RI:

$$\Delta r_{ij}'^T \Delta r'_{ij} \tag{19}$$

$$= (W_p \cdot R(r_j - r_i))^T (W_p \cdot R(r_j - r_i)) \tag{20}$$

$$= w_p^2 (r_j - r_i)^T R^T R (r_j - r_i) \quad \text{using} \quad W_p = w_p I_p \tag{21}$$

$$= w_p^2 (r_j - r_i)^T (r_j - r_i) \quad \text{using} \quad R^T R = I_p, R \in \mathrm{SO}(p) \tag{22}$$

$$= \Delta r_{ij}^T \Delta r_{ij} \tag{23}$$

$$\implies A_{ij} = A'_{ij} \quad \square. \tag{24}$$

A.2 PaPE is Compatible with Efficient Attention Kernels

We claim in Section 4.1 that

$$\langle f_q(q_i, r_i), f_k(k_j, r_j) \rangle = \langle a_i, \Delta r_{ij}^{\odot 2} \rangle + \langle b_i, \Delta r_{ij} \rangle + \langle q_i, k_j \rangle. \tag{25}$$

Recall that $M_i = W_p^T \mathrm{diag}(a_i) W_p$ and that we define in Equations (13) and (14) f_q and f_k as

$$\begin{array}{rcccccccccccc} f_q(q_i, r_i) & = & q_i & \oplus & r_i^T M_i r_i & \oplus & \mathrm{flatten}(M_i) & \oplus & -2M_i r_i & \oplus & W_p^T b_i & \oplus & -\langle W_p^T b_i, r_i \rangle \\ f_k(k_j, r_j) & = & k_j & \oplus & 1 & \oplus & \mathrm{flatten}(r_j \otimes r_j) & \oplus & r_j & \oplus & r_j & \oplus & 1 \end{array}$$

Completing the dot product block by block shows that $\langle f_q(q_i, r_i), f_k(k_j, r_j) \rangle =$

$$\langle q_i, k_j \rangle + r_i^T M_i r_i + \langle \mathrm{flatten}(M_i), \mathrm{flatten}(r_j \otimes r_j) \rangle - \langle 2M_i r_i, r_j \rangle + \langle W_p^T b_i, r_j \rangle - \langle W_p^T b_i, r_i \rangle. \tag{26}$$

As such, we prove the claim by proving the following equations

$$\langle a_i, \Delta r_{ij}^{\odot 2} \rangle = r_i^T M_i r_i + \langle \mathrm{flatten}(M_i), \mathrm{flatten}(r_j \otimes r_j) \rangle - \langle 2M_i r_i, r_j \rangle \tag{27}$$

$$\langle b_i, \Delta r_{ij} \rangle = \langle W_p^T b_i, r_j \rangle - \langle W_p^T b_i, r_i \rangle \tag{28}$$

from which Equation (25) directly follows.

Proof of quadratic terms (Equation (27)).

First, we see that M is symmetric because

$$\left[W_p^T \mathrm{diag}(a_i) W_p\right]_{jk} = \sum_{\ell=1}^{m} a_\ell w_{\ell j} w_{\ell k}, \tag{29}$$

and $w_{\ell j} \cdot w_{\ell k} = w_{\ell k} \cdot w_{\ell j}$. Therefore, $M_{jk} = M_{kj}$, making M symmetric.

It also holds that

$$2 r_j^T M_i r_i = r_i^T M_i r_j + r_j^T M_i r_i \tag{30}$$

because M_i is symmetric.

We also note that

$$\langle \text{flatten}(M_i), \text{flatten}(r_j \otimes r_j) \rangle = r_j^T M_i r_j. \tag{31}$$

Together, we can write the right hand side of Equation (27) as

$$r_i^T M_i r_i + r_j^T M_i r_j - r_i^T M_i r_j - r_j^T M_i r_i, \tag{32}$$

which simplifies to

$$\begin{aligned} r_i^T M_i (r_i - r_j) + r_j^T M_i (r_j - r_i) &= -r_i^T M_i (r_j - r_i) + r_j^T M_i (r_j - r_i) && (33)\\ &= (r_j - r_i)^T M_i (r_j - r_i) && (34) \end{aligned}$$

Substituting with $M_i = W_p^T \text{diag}(a_i) W_p$ completes the proof:

$$\begin{aligned} (r_j - r_i)^T M_i (r_j - r_i) &= (r_j - r_i)^T W_p^T \text{diag}(a_i) W_p (r_j - r_i) && (35)\\ &= \left((r_j - r_i)^T W_p^T\right) \text{diag}(a_i) \left(W_p (r_j - r_i)\right) && (36)\\ &= \Delta r_{ij}^T \text{diag}(a_i) \Delta r_{ij} && (37)\\ &= \langle a_i, \Delta r_{ij}^{\odot 2} \rangle \quad \square. && (38) \end{aligned}$$

Proof of linear terms (Equation (28)).

We verify Equation (28) below:

$$\begin{aligned} \langle W_p^T b_i, r_j \rangle - \langle W_p^T b_i, r_i \rangle &= \langle W_p^T b_i, r_j - r_i \rangle && (39)\\ &= \left(W_p^T b_i\right)^T (r_j - r_i) && (40)\\ &= \left(b_i^T W_p\right) (r_j - r_i) && (41)\\ &= b_i^T \left(W_p (r_j - r_i)\right) && (42)\\ &= b_i^T \Delta r_{ij} && (43)\\ &= \langle b_i, \Delta r_{ij} \rangle \quad \square. && (44) \end{aligned}$$

B Generalizations of PaPE

We motivate and discuss generalizations of PaPE.

B.1 Polynomial Generalizations of PaPE

PaPE can be viewed as the first instance of a larger family of multivariate polynomials on Δr_{ij} with context-dependent coefficients. Although we leave this direction unexplored in the current work, we are eager to investigate other members of this function family in future work. We elaborate on this perspective here and describe polynomial generalizations of PaPE.

Let $x_i \in \mathbb{R}^d$ and $x_j \in \mathbb{R}^d$ be two tokens at positions $r_i \in \mathbb{R}^p$ and $r_j \in \mathbb{R}^p$. An expressive family of continuous, context-aware, and translation invariant functions are polynomials of the difference $(r_j - r_i) \in \mathbb{R}^p$,

$$g(x_i, r_i, x_j, r_j) = \sum_{n=0}^{N} a_n(x_i, x_j)(r_j - r_i)^{\odot n}, \tag{45}$$

with coefficients $a_n \in \mathbb{R}^{1 \times p}$ and where $(\cdot)^{\odot n}$ represents the element-wise exponentiation.

For compatibility with efficient attention kernels, the function g needs to be representable as a dot product of finite feature vectors that each only depend on the position and content of either of the two tokens. Since $(r_j - r_i)^n$ can be expanded to sums of monomials $r_i^l r_j^{n-l}$, the expression is separable into feature vectors, as long as the coefficients a_n are of the type $a_n = h_q(x_i) h_k(x_j)$ for functions

Table 8: **Additional baselines.** PaPE confirms its high generality when compared to context-aware baselines and learned positional embeddings from ViT.

Dataset	CPVT	CvT	ViT-PE	PaPE (ours)
UCF101	-	-	38.3 ± 3.1	**49.5** ± 1.6
ImageNet-1K	79.2	78.9	79.2	**80.6**
COCO	32.1	34.1	14.7	**38.9**

h_q, h_k. This results in terms of type $\langle h_q(x_i) r_i^l, r_j^{n-l} h_k(x_j) \rangle$. Note that in PaPE we set $h_k(x_j) = 1$, leading to coefficients of type $a_n(x_i, x_j) = h_q(x_i)$.

A further generalization of PaPE is into the multivariate polynomials. To simplify notation, let $x = (r_j - r_i) = (x_1, x_2, \ldots, x_p)$. Multivariate polynomials are finite sums of terms of the type $c_\alpha x_1^{\alpha_1} x_2^{\alpha_2} \ldots x_p^{\alpha_p}$, where $c_\alpha \in \mathbb{R}$, and $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_p) \in \mathbb{N}^p$ is a multi-index. The degree N of the multivariate polynomial is the largest sum of exponents corresponding to non-zero coefficients, i.e., $N = \max\{\sum_{i=1}^p \alpha_i \mid c_\alpha \neq 0\}$.

This broader family can represent terms that cannot be represented in the former, e.g., bilinear forms $(r_j - r_i)^T B (r_j - r_i)$ for any $B \in \mathbb{R}^{p \times p}$. Similar to the former case, multivariate polynomials can also be separated into feature vectors for efficient attention kernel compatibility, as long as coefficients c_α are of the type $h_q(q_i) h_k(k_j)$.

B.2 Relaxations of Translation and Rotation Invariance

We have presented PaPE and PaPE-RI. PaPE is translation invariant and PaPE-RI is also rotation invariant. We discussed in Section 5.3 how absolute positions can be useful in some edge cases, and we explicitly stated rotation invariance as a task-specific principle (Section 3). Therefore, we here propose extensions of PaPE to relax translation and rotation invariance when doing so may be useful. Note that these extensions are untested, but may inspire future work.

Relaxing translation invariance. We include a position-dependent term that turns Equation (9) into

$$S_{ij} = \langle a_i, \Delta r_{ij}^{\odot 2} \rangle + \langle b_i, \Delta r_{ij} \rangle + \langle q_i, k_j \rangle + \langle c_i, W_p r_i \rangle, \tag{46}$$

where $c_i = W_c x_i$ with $W_c \in \mathbb{R}^{m \times d}$ being learnable. The final term breaks translation invariance. We can further relax its influence through a loss that encourages the norm of c_i to be close to 0, making it approximately translation invariant, but enabling the model to still use absolute positions.

Relaxing rotation invariance. We can relax PaPE to be approximately rotation invariant by regularizing W_p towards the constrained form used in PaPE-RI and regularizing b_i towards 0, recovering PaPE-RI in the limit of strong regularization and unconstrained PaPE without regularization. Such regularization could, for example, be achieved with an auxiliary loss.

C Additional Experiments & Results

This section reports experiments and results that are informative to understand PaPE's performance, but does not fit within the main text.

C.1 Baselines

Here we compare against additional baselines. First, we include the learned positional embeddings from ViT [18] and denote it ViT-PE. This gives a comparison to another absolute positional encoding besides nD-sincos. We also include context-aware baselines: CPVT [9] and CvT [50]. ViT-PE is only compatible with discrete grids, meaning we can only evaluate it on UCF101, ImageNet-1K, and COCO. We follow the usual approach of using bicubic interpolation on the 2D grid when extrapolating with ViT-PE. CPVT and CvT further impose 2D constraints due to using 2D convolutions, restricting their evaluation to ImageNet-1K and COCO.

Table 8 reports the findings of the additional baselines. We find that PaPE consistently obtains higher accuracy than the additional baselines. This validates PaPE's high generality even further. Table 9

Table 9: **Extrapolation with additional baselines.** PaPE remains best at extrapolation by large margins ranging from 17.6 to 21.0 at resolution 1024^2.

Positional Encoding	224^2	320^2	384^2	448^2	512^2	640^2	768^2	896^2	1024^2
CPVT	79.2	79.6	79.1	78.0	76.6	73.4	68.5	62.9	56.6
CvT	78.9	79.5	79.0	78.2	77.1	73.7	68.4	61.4	53.9
ViT-PE	79.2	79.1	78.1	76.6	75.1	71.7	67.4	62.6	57.3
PaPE (ours)	**80.6**	**81.6**	**81.7**	**81.6**	**81.3**	**80.5**	**79.2**	**77.3**	**74.9**

Table 10: **Extrapolation results with position interpolation.** The brackets indicate the increase (green) or decrease (red) compared to the corresponding configuration without interpolation in Table 1. **Best** and next-best. nD-sincos benefits greatly from interpolation whereas RoPE and RoPE-Mixed benefit from resolution 640^2 and above. Position interpolation is detrimental to all of the attention bias methods (nD-ALiBi, LookHere, PaPE, and PaPE-RI).

Positional Encoding	224^2	320^2	384^2	448^2	512^2	640^2	768^2	896^2	1024^2
nD-sincos [48]	79.9	79.9 (+3.1)	78.8 (+6.2)	77.4 (+11.5)	75.8 (+18.3)	71.8 (+30.5)	66.9 (+39.7)	61.3 (+44.5)	56.0 (+46.7)
RoPE [41]	80.3	80.6 (-0.3)	80.0 (-0.6)	**78.9** (-0.8)	**77.6** (-0.6)	**74.5** (+0.5)	**69.9** (+2.9)	**64.8** (+8.6)	**59.8** (+17.5)
RoPE-Mixed [26]	80.0	72.9 (-7.5)	72.1 (-7.1)	73.0 (-4.1)	67.8 (-5.8)	63.5 (+2.0)	56.1 (+12.7)	49.5 (+22.4)	42.9 (+25.8)
nD-ALiBi [19]	80.0	77.9 (-0.5)	77.2 (-0.6)	76.0 (-0.6)	74.4 (-0.9)	70.6 (-1.6)	65.4 (-2.9)	59.8 (-4.3)	53.9 (-5.2)
LookHere [20]	80.5	**80.9** (-0.5)	**80.2** (-0.9)	**78.9** (-1.5)	77.4 (-2.3)	73.6 (-3.9)	68.8 (-5.4)	63.7 (-5.9)	59.0 (-5.4)
PaPE (ours)	**80.6**	80.7 (-0.9)	79.6 (-2.1)	78.1 (-3.5)	76.5 (-4.8)	72.0 (-8.5)	66.4 (-12.8)	60.4 (-16.9)	54.8 (-20.1)
PaPE-RI (ours)	80.0	80.3 (+0.1)	79.4 (+0.2)	78.1 (0.0)	76.4 (-0.5)	72.2 (-1.4)	66.5 (-3.2)	59.7 (-4.9)	52.8 (-5.8)

also validates this for extrapolation. We find PaPE to remain a strong positional encoding when compared to ViT-PE, CPVT, and CvT.

C.2 Extrapolation with and without Position Interpolation

Here, we further elaborate on the extrapolation performance of PaPE and the baseline methods.

Without position interpolation. The main extrapolation experiment evaluates how the methods perform without position interpolation. Concretely, during training we only see positions in the range $[1, 14]$ because we train at a resolution of 224^2 with a patch size of 16, since $224/16 = 14$. At higher resolutions, however, this expands up to $[1, 64]$, since $1024/16 = 64$. As previously mentioned, we see in Table 1 that PaPE excels in this setting.

With position interpolation. Prior works find that sinusoidal and RoPE-based positional encodings extrapolate better with position interpolation [8, 17]. It works by mapping the test-time position range back into the original training range. We do this by scaling the positions by a factor of $224/R$, where R is the target resolution. For example, at $R = 448$, each position is multiplied by 0.5, compressing the effective positions back into the $[1, 14]$ training range. Table 10 shows how PaPE and the baselines perform with position interpolation. We clearly see how nD-sincos in particular is greatly improved with position interpolation. RoPE and RoPE-Mixed drop for resolutions below 640^2, but greatly increase their accuracy from there and up. These results confirm that position interpolation is useful for sinusoidal encodings and partially useful for RoPE-based encodings. We also find that all four attention bias positional encodings, including PaPE, are negatively affected by position interpolation.

Best-performing configuration. Although position interpolation partially mitigates the shortcomings of sinusoidal and RoPE-based encodings, there is still no encoding that approaches the extrapolation performance of PaPE *without* position interpolation. This is evident in Table 11, which reports extrapolation scores for the best-performing configuration of each positional encoding. In this comparison, nD-sincos, RoPE, and RoPE-Mixed rely on interpolation, whereas nD-ALiBi, LookHere, PaPE, and PaPE-RI do not. We observe in particular that PaPE remains the strongest encoding across all resolutions, with LookHere consistently ranking as the next-best method for extrapolation. As a result, the 10.5 improvement at resolution 1024^2 persists even after taking position interpolation into account.

Table 11: **Extrapolation results for best-performing configuration per positional encoding.** With position interpolation: nD-sincos, RoPE, and RoPE-Mixed. Without: nD-ALiBi, LookHere, PaPE, and PaPE-RI. **Best** and next-best. Although position interpolation is detrimental to PaPE, it remains the best encoding for classification extrapolation—even when applying position interpolation to the encodings that benefit from it.

Positional Encoding	224^2	320^2	384^2	448^2	512^2	640^2	768^2	896^2	1024^2
nD-sincos [48]	79.9	79.9	78.8	77.4	75.8	71.8	66.9	61.3	56.0
RoPE [41]	80.3	80.6	80.0	78.9	77.6	74.5	69.9	64.8	59.8
RoPE-Mixed [26]	80.0	72.9	72.1	73.0	67.8	63.5	56.1	49.5	42.9
nD-ALiBi [19]	80.0	78.4	77.8	76.6	75.3	72.2	68.3	64.1	59.1
LookHere [20]	80.5	81.4	81.1	80.4	79.7	77.5	74.2	69.6	64.4
PaPE (ours)	**80.6**	**81.6**	**81.7**	**81.6**	**81.3**	**80.5**	**79.2**	**77.3**	**74.9**
PaPE-RI (ours)	80.0	80.2	79.2	78.1	76.9	73.6	69.7	64.6	58.6

Table 12: **Extrapolation on ImageNetV2.** PaPE maintains its rank as the **best** positional encoding at extrapolating, increasing the accuracy at 1024^2 by 9.9 over the next-best (LookHere).

Positional Encoding	224^2	320^2	384^2	448^2	512^2	640^2	768^2	896^2	1024^2
nD-sincos [48]	68.3	64.2	58.1	50.7	42.3	27.1	16.6	10.1	5.6
RoPE [41]	**69.2**	70.3	69.7	67.9	65.9	60.1	51.8	41.2	29.1
RoPE-Mixed [26]	68.2	69.2	67.7	64.3	59.5	44.2	28.3	17.0	10.4
nD-ALiBi [19]	68.1	69.4	67.7	65.7	63.9	59.4	54.6	49.1	43.4
LookHere [20]	68.9	70.8	70.2	69.5	68.3	65.5	61.0	56.5	51.6
PaPE (ours)	**69.2**	**70.9**	**71.2**	**70.6**	**70.1**	**68.9**	**66.7**	**64.5**	**61.5**
PaPE-RI (ours)	68.8	69.3	67.9	66.2	64.3	59.5	54.3	48.8	43.1

Table 13: **Extrapolation on ImageNet-Renditions.** PaPE maintains its rank as the **best** positional encoding at extrapolating, increasing the accuracy at 1024^2 by 8.2 over the next-best (LookHere).

Positional Encoding	224^2	320^2	384^2	448^2	512^2	640^2	768^2	896^2	1024^2
nD-sincos [48]	27.0	18.4	13.3	8.7	5.4	2.5	1.3	0.9	0.4
RoPE [41]	29.2	26.8	24.1	21.2	18.3	12.8	8.3	4.9	2.7
RoPE-Mixed [26]	28.2	24.3	20.2	16.4	12.4	6.3	2.6	1.3	0.8
nD-ALiBi [19]	27.2	24.5	21.2	18.3	16.2	12.8	10.2	8.1	6.2
LookHere [20]	28.6	26.4	23.4	21.1	18.8	15.3	12.1	9.5	7.5
PaPE (ours)	**29.4**	**28.0**	**26.7**	**25.4**	**24.3**	**22.0**	**19.7**	**17.8**	**15.7**
PaPE-RI (ours)	26.8	22.9	20.3	18.3	16.7	13.0	9.9	7.8	6.1

Table 14: **Extrapolation on ObjectNet.** PaPE maintains its rank as the **best** positional encoding at extrapolating, increasing the accuracy at 1024^2 by 7.8 over the next-best (LookHere).

Positional Encoding	224^2	320^2	384^2	448^2	512^2	640^2	768^2	896^2	1024^2
nD-sincos [48]	28.9	24.6	20.9	15.1	10.7	5.9	3.5	2.0	1.0
RoPE [41]	31.0	33.2	33.2	32.2	30.0	24.3	18.7	13.4	9.2
RoPE-Mixed [26]	28.5	29.8	28.2	25.6	22.0	13.8	7.6	4.5	2.9
nD-ALiBi [19]	29.2	31.6	30.2	28.9	26.9	23.0	20.0	16.9	14.9
LookHere [20]	29.8	32.3	32.0	31.2	29.9	27.1	24.4	21.6	19.1
PaPE (ours)	**31.1**	**34.3**	**34.8**	**34.6**	**34.1**	**32.6**	**30.9**	**28.7**	**26.9**
PaPE-RI (ours)	29.4	30.5	30.3	28.7	27.0	23.3	18.7	15.7	13.2

Table 15: **Parameter-matched per-fold accuracy on UCF101.** PaPE (**best**) confirms its position as highest scoring over RoPE (next-best). The difference is statistically significant at p=0.026.

Positional Encoding	Fold 1	Fold 2	Fold 3	Mean ± std
nD-sincos [48]	39.3	41.1	38.1	39.5 ± 1.5
RoPE [41]	45.4	46.2	44.2	45.2 ± 1.0
RoPE-Mixed [26]	40.4	41.6	41.9	41.3 ± 0.8
nD-ALiBi [19]	41.4	43.9	40.3	41.9 ± 1.8
PaPE (ours)	**51.4**	**48.2**	**49.1**	**49.5** ± 1.6
PaPE-RI (ours)	40.2	41.9	43.4	41.8 ± 1.6

Table 16: **Parameter-matched extrapolation on ImageNet-1K.** PaPE preserves its strong classification extrapolation as the **best** positional encoding with a 9.6 improvement in accuracy at 1024^2 over the next-best (LookHere).

Positional Encoding	224^2	320^2	384^2	448^2	512^2	640^2	768^2	896^2	1024^2
nD-sincos [48]	79.6	76.8	72.7	65.8	58.0	39.5	22.3	11.8	6.9
RoPE [41]	80.2	80.8	80.1	79.0	77.3	72.4	64.3	52.9	39.8
RoPE-Mixed [26]	80.1	80.2	78.3	74.8	69.9	55.2	36.9	22.9	14.7
nD-ALiBi [19]	80.0	80.4	79.9	79.0	77.9	75.2	71.2	65.6	59.1
LookHere [20]	80.4	81.5	81.3	80.8	80.1	78.2	75.2	70.6	65.3
PaPE (ours)	**80.6**	**81.6**	**81.7**	**81.6**	**81.3**	**80.5**	**79.2**	**77.3**	**74.9**
PaPE-RI (ours)	79.8	80.3	78.2	75.7	73.2	67.7	60.9	53.4	45.7

C.3 Out-of-distribution Extrapolation

We report extrapolation results for the out-of-distribution datasets in Table 12 (ImageNetV2), Table 13 (ImageNet-Renditions), and Table 14 (ObjectNet). PaPE consistently ranks the highest across all 3 datasets, with notable increases in accuracy from 7.8 to 9.9 at resolution 1024^2 over the next-best positional encoding, LookHere. We find that the extrapolation capability of PaPE is unaffected by the distribution shifts.

C.4 Parameter-matched

Here, we describe the experimental setup of the experiments in Section 5.4. We retrain the baselines and PaPE-RI on UCF101, DvsGesture, and ImageNet-1K to match the parameters of PaPE. We choose UCF101 because it is representative of a significant result and because the 3-fold cross-validation allows for a statistical significance test. We choose DvsGesture because the results are close (nD-sincos and PaPE are tied as highest scoring), making it representative of the results where the score differences are small. Additionally, DvsGesture is spatially sparse and temporally dense while UCF101 is the opposite, making the two datasets representative of different types of domains. We include ImageNet-1K because it is a large-scale dataset, and it is essential for the extrapolation experiments. Note that PaPE-RI has fewer parameters than PaPE because $m = p$ for PaPE-RI ($p = 3$ for UCF101 and DvsGesture and $p = 2$ for ImageNet-1K), which is why we also retrain PaPE-RI with matched parameter counts. PaPE solely adds parameters that are used in the queries and keys. As such, we consider it the fairest comparison to match parameters by increasing the query and key head sizes of all other methods. Note also that this setting is particularly favorable to RoPE where larger query-key head sizes not only add more semantic capability, but also increase the capacity of the positional encoding [4]. We do not change the head size of the values as PaPE does not change this either. A ViT-B on UCF101 has 86.3M parameters while a ViT-B with PaPE ($m = 50$) has 97.4M parameters. The other methods match the additional 11.1M parameters of PaPE by increasing the query-key head size from 64 to 114.

We report per-fold accuracies on UCF101 for the parameter-matched experiments in Table 15. We observe that PaPE stays best with RoPE being next-best, and that the difference remains statistically significant at p=0.026.

Table 17: **Parameter-matched by reducing hidden size.** Reducing the hidden size from $d = 768$ to $d = 680$ does not diminish the performance of PaPE, supporting that PaPE's advantage stems from an improved positional encoding and not from increasing the number of parameters.

Size	UCF101	DvsGesture	GEN1	ImageNet-1K	COCO
Reduced ($d = 680$)	50.8	93.8	34.9	80.7	38.8
Full ($d = 768$)	49.5	93.4	34.8	80.6	38.9

Table 18: **UCF101 per-fold accuracy.** PaPE (**best**) achieves noticeable gains over RoPE (<u>next-best</u>) on all folds. The difference is statistically significant at p=0.011.

Positional Encoding	Fold 1	Fold 2	Fold 3	Mean ± std
nD-sincos [48]	37.8	39.1	39.3	38.7 ± 0.8
RoPE [41]	<u>45.3</u>	<u>43.8</u>	<u>42.8</u>	<u>43.9</u> ± 1.2
RoPE-Mixed [26]	40.6	41.1	41.9	41.2 ± 0.7
nD-ALiBi [19]	41.2	41.5	42.1	41.6 ± 0.5
PaPE (ours)	**51.4**	**48.2**	**49.1**	**49.5** ± 1.6
PaPE-RI (ours)	43.0	40.2	43.3	42.2 ± 1.7

Table 16 reports extrapolation results where the baselines and PaPE-RI are parameter-matched with PaPE. Matching parameters does not change that PaPE extrapolates noticeably better. The largest baseline increase in extrapolation is for LookHere (64.4 vs. 65.3 at 1024^2). Even with this increase, PaPE still increases the absolute extrapolation accuracy by up to 9.6, showing that its strong classification extrapolation stems from an improved positional encoding.

Finally, we reduce the hidden size, d, of PaPE to match the total parameter count to that of the baselines. We achieve this by reducing d from 768 to 680. Table 17 reports the results of this experiment. We find that the reduced version of PaPE does not obtain worse scores than the full version, strengthening the case for PaPE's generality stemming from an improved positional encoding rather than an increase in parameters.

C.5 UCF101 Per-Fold Accuracy

We have evaluated UCF101 using the official 3-fold cross-validation splits, but we only reported the mean accuracy in Table 2. Here, we report the accuracy of each fold as well as the mean and standard deviations in Table 18. We find from the additional results that the difference between PaPE (best) and RoPE (next-best) is statistically significant at p=0.011, confirming PaPE's spatio-temporal relevance.

C.6 Ablation Study: Extrapolation of m

Table 19 reports extrapolation results for the ablations of m in Section 5.6. The accuracy at 1024^2 is lowest at 68.2 for $m \in \{4, 8\}$ and highest at 69.8 for $m = 50$. We conclude that m does not affect

Table 19: **Ablation of m with extrapolation.** We follow the ablation setup in Section 5.6. m does not have a large effect on extrapolation.

m	224^2	320^2	384^2	448^2	512^2	640^2	768^2	896^2	1024^2
2	78.7	79.9	79.6	79.1	78.5	77.1	75.2	72.4	69.0
4	78.4	79.5	79.3	78.7	78.0	76.5	74.4	71.6	68.2
8	79.0	79.7	79.4	78.9	78.1	76.4	74.3	71.6	68.2
16	79.1	80.1	80.0	79.5	78.8	77.4	75.3	72.8	69.8
32	78.9	80.1	79.8	79.4	78.7	77.4	75.5	72.8	69.7
50	79.2	80.3	80.1	79.5	78.9	77.5	75.4	72.9	69.8
64	79.0	80.1	79.9	79.4	78.6	77.1	75.1	72.5	69.0

Table 20: **Efficiency.** Resource usage is averaged over 1000 samples with a batch size of 1, using 224^2 resolution images from ImageNet-1K. PaPE with $m = 50$ produces a moderate resource increase from 7% to 20%.

			Training			Inference		
Method	**Total Params.**	**Pos. Params.**	**VRAM (MB)**	**GFLOPS**	**Time (ms)**	**VRAM (MB)**	**GFLOPS**	**Time (ms)**
nD-sincos	86.4M	0	1669	105.9	12.5	362	35.1	1.4
nD-ALiBi	86.4M	0	1673	105.2	12.5	365	35.1	1.4
RoPE	86.4M	0	1669	105.9	12.6	363	35.1	1.4
RoPE-Mixed	86.4M	18.4K	1671	105.9	12.7	363	35.1	1.4
LookHere	86.4M	0	1694	105.2	12.6	387	35.1	1.4
PaPE (m=2)	86.9M	0.5M	1680	107.1	13.1	366	35.5	1.5
PaPE (m=4)	87.3M	0.9M	1688	107.6	13.1	369	35.6	1.5
PaPE (m=8)	88.2M	1.8M	1705	108.7	13.3	374	35.7	1.5
PaPE (m=16)	90.0M	3.6M	1778	110.8	13.5	394	36.1	1.5
PaPE (m=32)	93.5M	7.1M	1804	114.9	14.0	416	36.8	1.5
PaPE (m=50)	97.5M	11.1M	1901	119.7	14.6	433	37.6	1.6
PaPE (m=64)	100.6M	14.2M	1947	123.3	14.9	447	38.2	1.6
PaPE-RI	86.5M	110.6K	1674	106.4	12.9	365	35.3	1.5
Relative increase (Ref. PaPE $m = 50$)	13%	-	14%	14%	17%	20%	7%	14%

extrapolation greatly. Together with the efficiency results reported in Table 20, this shows that it is possible to significantly reduce the overhead of PaPE without compromising extrapolation.

D Efficiency

As shown in Section 4.1, PaPE increases the effective head size by $p^2 + 2p + 2$. It also introduces positional parameters: W_a, W_b, and W_p. We measure the number of additional positional parameters and the impact on memory usage, FLOPS, and step times for training as well as inference for different m in Table 20. The analysis considers a ViT-B/16 on ImageNet-1K at resolution 224^2 and a batch size of 1.

Overall, we find that PaPE increases resource usage moderately by 7%–20% for the $m = 50$ configuration that is used in the experiments, compared to the lowest resource baseline. The relative increase is 9%–23% for the largest considered option at $m = 64$. This shows that the overhead incurred by PaPE is within a reasonable range, and that PaPE remains a practical positional encoding for vision transformers.

A standard ViT-B (including its classification head) has 86.4M parameters. With PaPE, this footprint grows only by 1%–16%. Although this exceeds the baseline sizes, PaPE still delivers its advantages with a moderate increase in parameters.

Step times are measured on an RTX 4090 GPU. PaPE increases training step time by 0.6–2.4 ms and inference time by 0.1–0.2 ms over the fastest baselines. In absolute terms, this overhead is tiny, so PaPE remains fast. However, the relative increase is between 5%–19% for training and 7%–14% for inference, which may matter for some practical applications.

E Motivation for W_p

We introduced $W_p \in \mathbb{R}^{m \times p}$ in Equation (6) and noted that it is empirically beneficial, even if not strictly required to satisfy our goal. Here we elaborate on the motivation for introducing it.

W_p is intended to make optimization easier rather than to add model capacity (for $m > \min(d, p)$). Once $m > \min(d, p)$, PaPE has already spanned the representational space it needs and larger m mainly gives the optimizer more directions to discover useful patterns, rather than attaining greater capacity.

We can view W_p as expressing the p positional dimensions through m learned basis functions. As such, W_p can ease training by providing more opportunities for exploring different directions than the ones defined by the unprojected p dimensions. Expanding on this view, we can consider m as a bottleneck below and above a threshold of $\min(d, p)$.

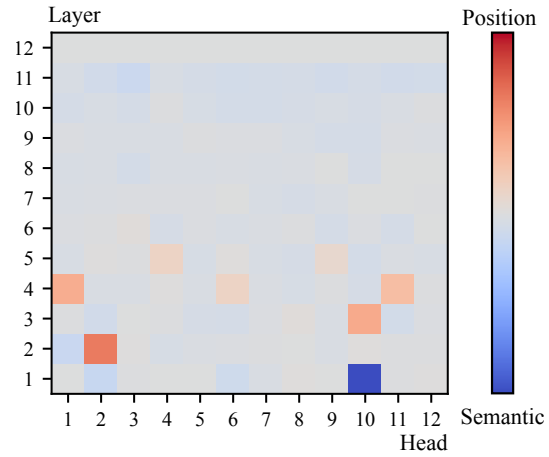

Figure 3: **Model analysis on ImageNet-1K.** Red ($z > 0$) highlights heads that lean heavily on positional information, while blue ($z < 0$) marks heads that prioritize semantic content in deciding what to attend to. Positions are used most strongly in early layers.

Below the threshold ($m < \min(d, p)$): increasing m should improve both representational capacity and the optimization landscape, since m is limiting what the model can express.

Above the threshold ($m > \min(d, p)$): in the b-term, larger m can still ease optimization, but doesn't add representational capacity. After training, results are unaffected if W_b and W_p are compressed to a full-rank representation. In the a-term the softplus and quadratic nonlinearities mean larger m can still help both representation and optimization, though gains are likely bounded since the m values are aggregated via a dot product.

F Model Analysis

The decomposed attention in PaPE eases the investigation of how the model uses position and semantic information. Although model analysis is not the focus of this work, here we demonstrate an analysis method that is enabled by PaPE.

The attention matrix can—disregarding the dot product scaling factor—be represented as $A_{ij} = \frac{1}{\gamma_i} P_{ij} Y_{ij}$, with positional components

$$P_{ij} = \exp\left(\left\langle a_i, \Delta r_{ij}^{\odot 2} \right\rangle + \langle b_i, \Delta r_{ij} \rangle\right), \tag{47}$$

semantic components $Y_{ij} = \exp\left(\langle q_i, k_j \rangle\right)$, and normalization factors $\gamma_i = \sum_j P_{ij} Y_{ij}$. We introduce the score z,

$$z = \mathbb{E}_{ij}\left[\frac{1}{\gamma_i}(P_{ij} - Y_{ij})\right], \tag{48}$$

to quantify the relative importance of positional vs. semantic terms of a given attention head. Since attention patterns are often sparse, it is helpful to refine the analysis by considering only the top-attended keys for each query, which we define as the minimal set of keys whose cumulative attention in A_i reaches a threshold τ.

Figure 3 breaks down how each head in the ImageNet-1K model focuses on positions and semantics on average by visualizing the scores z, using the threshold $\tau = 80\%$. Most heads balance positions and semantics quite evenly. Yet, a few stand out: some heads (e.g., L2H2) are clearly position-focused, whereas L1H10 is highly driven by semantics. Interestingly, these specialized heads cluster in the early layers (L1–L5), suggesting that emphasizing positional or semantic signals is especially beneficial when transforming low-level inputs into higher-level representations.

G Comparison of nD-ALiBi and PaPE-RI

The methods nD-ALiBi and PaPE-RI have methodological similarities, which we compare here.

Table 21: **Hyperparameters.** The hyperparameters are shared for all positional encodings for a given dataset.

Parameter	UCF101	DvsGesture	GEN1	ImageNet-1K	COCO	ScanNet	ModelNet40	nuScenes
Batch size	32	32	64	1024	64	12	32	3
Learning rate	0.00003	0.00005	0.0001	0.0006	0.0006	0.006	0.001	0.003
Weight decay	0.05	0.05	0.05	0.05	0.05	0.05	0.01	0.03
Momentum (B_1)	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9
Momentum (B_2)	0.999	0.999	0.999	0.999	0.999	0.999	0.999	0.99
Warmup epochs	20	15	7.5	15	15	40	5	1
Total epochs	200	300	150	300	150	800	300	10
Training resolution ($W \times H$)	320×240	128^2	304×240	224^2	640^2	-	-	704×256
Patch size	16	16	16	16	16	1024	1024	8
Drop path	0.1	0.1	0.1	0.1	0.1	0.3	0.3	-
Label smoothing	-	0.1	-	0.1	-	-	-	-

nD-ALiBi works by subtracting the distance matrix D,

$$D_{ij} = \|r_j - r_i\|_2, \tag{49}$$

between token positions from the S matrix (pre-softmax attention matrix), with a constant scaling factor for each attention head. For simplicity, we consider the case of one head without loss of generality,

$$S_{\text{ALiBi}} = QK^T + \alpha D \tag{50}$$

$$\alpha \in \mathbb{R}_{<0}. \tag{51}$$

Similar behavior is obtained by PaPE-RI, inducing the same properties—translation and rotation invariance. The main difference is that PaPE-RI is designed to subtract the *squared* distance matrix, where the coefficient can also depend on the token content, leading to

$$S_{\text{PaPE-RI}} = QK^T + \alpha(X) \odot D^2 \tag{52}$$

$$\alpha_{ij}(X) = \alpha_{ij}(x_i) \in \mathbb{R}_{<0}. \tag{53}$$

A further difference is that PaPE-RI is compatible with efficient attention kernels, while nD-ALiBi is not.

H Training Details

We now detail the training configurations used for all datasets. All models are optimized with AdamW [32], using a OneCycle [39] learning rate schedule with cosine decay. The only exception is UniTR on nuScenes, which uses Adam [27]. Table 21 summarizes the common hyperparameters for all datasets. Unless otherwise specified, all models use the same Transformer settings (i.e. #layers, #heads, dimensionality, and head size) as ViT-B [18]. Dataset-specific configurations are detailed in the following paragraphs.

UCF101. We follow ViViT [3], using 2-tubelets for each video sample and drawing up to 5 samples per video, with 10 frames separating consecutive 2-tubelets. The augmentations are RandAugment and HorizontalFlip. For both training and evaluation, we use the official 3-fold cross-validation splits and report the mean test performance. Mirroring the ImageNet-1K protocol, we further partition each training split into new training and validation sets. We do this by assigning the first video group of every action class to the validation set.

DvsGesture. We utilize Spiking Patches [57] to extract asynchronous and spatially sparse spatio-temporal tokens, configured with a spike threshold, σ, of 256 and a refractory period, T, of 100 ms. The model is a ViT-B where we follow the token embedding method of Øhrstrøm et al. [57]. Since the temporal positions are at a microsecond resolution, we rescale them by $1/50000$. We apply two sets of augmentations: NDA [29] and EventDrop [23].

GEN1. Similar to DvsGesture, we extract tokens using Spiking Patches with $\sigma = 256$ and $T = 100$ ms, and process them with a ViT-B backbone. Here, however, we scale temporal positions by $1/100000$ and sample the most recent 500K events relative to the prediction time. Because the tokens are asynchronous, they are not directly compatible with the image-like tensors expected by the ViTDet neck and YOLOv10 head. Therefore, we first convert the backbone outputs into an image-like

tensor. The way it works is that for each spatial location, we fill its slot with the most recent token at that spatial position. We augment with EventDrop and rotations.

ImageNet-1K. We treat the official validation split as our test set and divide the original training split into a new training set and a validation set. The ratio is 99% for training and 1% for validation, using a stratified split that preserves class balance in each subset. For the classifier head, we initialize the weights to 0 and the bias to $-\log(1000)$, so that the initial class probabilities are uniform. We use the following set of augmentations: RandAugment [11], HorizontalFlip, MixUp [55], and CutMix [54].

COCO. The detection model pairs a ViT-B backbone with a ViTDet [30] neck (strides 8, 16, and 32) and a YOLOv10 [45] head and loss. For data augmentation, we keep RandAugment but explicitly drop shear and rotation, as they corrupt ground-truth bounding boxes. In addition, we apply Large Scale Jitter [22], following ViTDet.

ScanNet and ModelNet40. We use Point Transformer V3 [51] with the dataset-specific configurations from the official code repository. This includes the same augmentations as well as model configuration and size. However, we do not initialize the weights from pretrained models.

nuScenes. We use UniTR [46], following the same architecture, model size, hyperparameters, and augmentations as in the official code repository.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The claims are clearly defined and are highlighted using enumerated red circles at the end of the introduction.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Section 6 discusses the main limitations and Appendix D quantifies it.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: Proofs are in Appendix A and are referenced in the main paper.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The method is fully described in Section 4 and the complete experimental setup (including detailed hyperparameters) is presented in Section 5.1 and Appendix H. Code and model weights are released publicly and referenced in the main text.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The code is publicly available and referenced in the main text.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [Yes]

Justification: See Section 5.1 and Appendix H for all details listed in the question.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: A "partially" option would have been more correct as UCF101 results are reported with standard deviations (see Appendix C.5), but this is not the case for all datasets as we prioritized the limited computational resources on increasing the diversity and quantity of datasets, in order to assess the generality of the method.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [No]

   Justification: Total compute is not provided, but we provide per-step resource usage in terms of memory, FLOPS, and time for both training and inference in Appendix D.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The paper proposes a vision position encoding and does, as such, not violate any parts of the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: The paper proposes a vision position encoding. This is not tied to any particular application and while (as the examples below suggest) any general advancement may be applied negatively, we do not see direct paths to such applications.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.

- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: We do not believe that a paper about vision position encoding poses such risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All assets are properly credited.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The code is properly documented.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: LLMs are solely used for writing, editing, and formatting purposes.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.