MINKOWSKIPE: MINKOWSKI POSITIONAL ENCODING FOR SPATIOTEMPORAL COUPLING

Anonymous authors
Paper under double-blind review

ABSTRACT

Modeling spatiotemporal coupling is a key challenge in building physical intelligence across scales, from microscopic to macroscopic. Existing models capture such structure broadly through physics-motivated dynamical formulations or learning-motivated architectures. The former provide stronger priors but may constrain flexibility, whereas the latter are more flexible but leave the spatiotemporal coupling largely implicit. We therefore seek an approach that combines flexible learning with an explicit geometric bias for jointly modeling time and space. To this end, we propose *Minkowski Positional Encoding* (MinkowskiPE), which uses joint temporal and spatial coordinates to parameterize Lorentz transformations applied to query and key features. With MinkowskiPE, the query–key attention score depends on position only through the relative spacetime displacement between the two tokens and is therefore invariant to global translation of the coordinates. This paradigm retains the standard dot-product attention interface and remains compatible with efficient attention implementations. We evaluate MinkowskiPE on microscopic molecular dynamics and macroscopic video prediction tasks, achieving the best results on all nine multi-trajectory molecular evaluations and reducing KTH video-prediction MSE by 9.9% relative to the best compared baseline while using roughly one-tenth as many parameters.

1 INTRODUCTION

Space and time are fundamental to the description of physical systems across scales, from microscopic molecular motion to macroscopic scene dynamics. Accordingly, physical intelligence requires the ability to capture their joint spatiotemporal structure. In molecular dynamics, each atomic state is associated with a time step and a position in three-dimensional space (Siebenmorgen et al., 2024; Liu et al., 2026). In video, each image patch is associated with a frame and a spatial location within that frame (Bertasius et al., 2021; Arnab et al., 2021). Despite their differences in scale, modality, and data structure, both settings can therefore be viewed as collections of observations with spatiotemporal coordinates.

Existing methods for spatiotemporal modeling broadly fall into two categories. The first research line is motivated by the view of temporal modeling as dynamical evolution, as in Figure 1(a). This includes recurrent and convolutional propagation (Shi et al., 2015; Wang et al., 2017; Gao et al., 2022a), as well as continuous-time evolution parameterized by differential equations (Chen et al., 2018; Liu et al., 2026). Other models incorporate stronger dynamical structure through physical priors or constraints (Raissi et al., 2019; Greydanus et al., 2019; Pfaff et al., 2021). Across these methods, temporal structure is organized through a prescribed propagation or evolution mechanism, introducing stronger inductive structure but potentially limiting the model's flexibility in learning spatiotemporal dependencies from data.

The second research line is learning-motivated, focusing on the data-driven modeling of spatiotemporal coupling, as in Figure 1(b). Transformer-based models usually capture the coupling of time and space through the structure of learned attention, such as factorized spatial and temporal attention, multiscale attention, and trajectory-aware attention (Bertasius et al., 2021; Arnab et al., 2021; Fan et al., 2021; Patrick et al., 2021). Such models provide more flexibility in modeling how information interacts across space and time, but the relation between temporal and spatial coordinates generally remains implicit.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

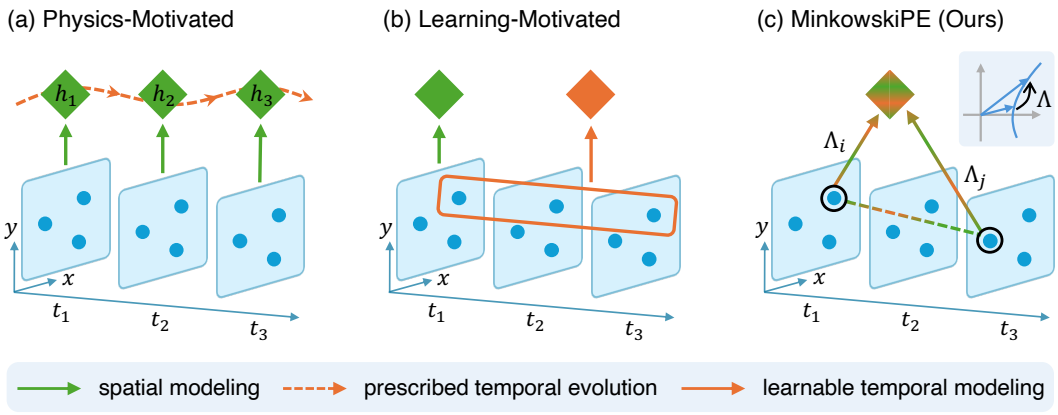

Figure 1: Different perspectives on spatiotemporal modeling. (a) Physics-motivated approaches impose explicit structure on temporal evolution. (b) Learning-motivated approaches learn spatiotemporal dependencies directly from data. (c) MinkowskiPE combines physics-inspired geometric priors with flexible data-driven learning.

These two directions highlight a trade-off between physical structure and data-driven flexibility in spatiotemporal coupling: physics-motivated methods often limit expressiveness, whereas learning-based methods often lack explicit physical priors. This naturally raises a question: *Can we design a spatiotemporal coupling paradigm that combines the flexibility of data-driven learning with a strong physical prior?* To explore this question, we focus on positional encoding as a mechanism for modeling temporal and spatial coordinates while preserving their distinct roles (Vaswani et al., 2017). Further, Minkowski geometry provides such a structure, placing time and space within a unified coordinate system while assigning them different geometric roles (Carroll, 2019).

Our Contributions. We propose *Minkowski Positional Encoding* (MinkowskiPE), a relative positional encoding for spatiotemporal modeling, as illustrated in Figure 1(c). Each token is associated with a spacetime coordinate that parameterizes Lorentz transformations applied to its query and key features. For multidimensional spacetime coordinates, we learn multiple projection directions and apply the corresponding transformations across feature blocks. As a result, the pairwise attention depends on position only through the relative spacetime displacement. MinkowskiPE preserves the standard dot-product attention interface, allowing it to directly benefit from efficient attention implementations such as FlashAttention (Dao et al., 2022).

Empirically, we evaluate MinkowskiPE on two spatiotemporal prediction tasks, molecular dynamics and video prediction. Molecular dynamics involves the three-dimensional motion of atoms over time, whereas video prediction concerns the evolution of two-dimensional visual scenes across frames. Together, these tasks evaluate the same positional formulation across different physical scales, data modalities, and spatial structures. On molecular dynamics prediction, MinkowskiPE ranks among the top two in 26 of 30 evaluations across ten protein–ligand systems in the single-trajectory setting, and achieves the best performance on all nine evaluations across three multi-trajectory datasets. On KTH video prediction, MinkowskiPE achieves the best performance on four of five metrics among the compared methods, reducing MSE by 9.9% while using roughly one-tenth as many parameters as the lowest-MSE baseline.

2 Related Work

Spatiotemporal Modeling. Existing models encode temporal evolution through a range of architectural and dynamical mechanisms. Recurrent and convolutional models organize temporal information through sequential propagation or local temporal transformations (Shi et al., 2015; Wang et al., 2017; Gao et al., 2022a), while continuous-time approaches parameterize evolution using differential equations (Chen et al., 2018; Liu et al., 2026). Other methods incorporate stronger dynamical or physical structure through differential-equation constraints (Raissi et al., 2019; Guen & Thome, 2020), Hamiltonian dynamics (Greydanus et al., 2019), or graph-based simulation (Sanchez-

Gonzalez et al., 2020; Pfaff et al., 2021). Transformer-based models instead learn spatiotemporal dependencies through attention mechanisms such as factorized spatial–temporal attention (Bertasius et al., 2021; Arnab et al., 2021), multiscale representations (Fan et al., 2021), local windows (Liu et al., 2022; Gao et al., 2022b), and trajectory-aware attention (Patrick et al., 2021). Our work differs by introducing inductive bias at the positional representation level rather than prescribing temporal evolution or a specific attention organization.

Positional Representations. Transformers incorporate positional information through learned relative embeddings (Shaw et al., 2018), relative attention formulations (Dai et al., 2019), attention biases (Raffel et al., 2020; Press et al., 2022), and feature-space transformations (Su et al., 2024). These mechanisms were later generalized to multidimensional coordinates in vision and multimodal models (Heo et al., 2024; Lu et al., 2024; Chu et al., 2024; Wang et al., 2024), and further developed through Lie-group transformations, commuting operators, translation-invariant constructions, and group representations (Ostmeier et al., 2025; Yu et al., 2025; Schenck et al., 2025; Liu et al., 2025a; Zhang et al., 2026; Yao & Yang, 2025). Related work has also explored hyperbolic feature transformations for one-dimensional positional encoding (Dai et al., 2025). More recent work has extended positional representations to spatiotemporal settings, such as video (Yang et al., 2025; Wei et al., 2025; Liu et al., 2025b). MinkowskiPE instead uses joint temporal and spatial event coordinates to parameterize Lorentz transformations within a single relative positional mechanism.

3 Preliminaries

3.1 Relative Positional Encoding

Let $q_i, k_j \in \mathbb{R}^{d_{\text{em}}}$ denote the query and key vectors associated with positions x_i and x_j. A relative positional encoding is designed such that their pairwise interaction depends on the positions only through their relative displacement $\Delta x_{ij} = x_j - x_i$, *i.e.*,

$$\langle f_q(q_i, x_i),\ f_k(k_j, x_j)\rangle = g(q_i, k_j, \Delta x_{ij}), \tag{1}$$

where $\langle \cdot, \cdot \rangle$ denotes the dot product in feature space, f_q and f_k denote feature transformations, and g denotes the resulting query–key score before attention scaling and softmax.

Equivalently, this score is invariant under a common translation $x_i \mapsto x_i + a$ and $x_j \mapsto x_j + a$. Existing positional encodings realize this property through relative embeddings, attention biases, or position-dependent transformations of query and key features (Shaw et al., 2018; Raffel et al., 2020; Press et al., 2022; Su et al., 2024).

3.2 Minkowski Geometry and Lorentz Transformations

We next introduce the geometric and algebraic structure used in our construction. A spatiotemporal *event* in $(1 + d)$-dimensional spacetime is represented by

$$x = (t, r) \in \mathbb{R}^{1+d}, \tag{2}$$

where $t \in \mathbb{R}$ denotes the temporal coordinate and $r \in \mathbb{R}^d$ is the spatial coordinate.

Minkowski geometry distinguishes temporal and spatial directions through an indefinite *metric*

$$\eta = \text{diag}(-1, +1, \cdots, +1) \in \mathbb{R}^{(1+d)\times(1+d)}. \tag{3}$$

The metric η defines the indefinite bilinear form $u^\top \eta v$ on spacetime coordinates. For two events x_i and x_j, the relative displacement $\Delta x_{ij} = x_j - x_i$ has the associated quadratic form

$$\Delta x_{ij}^\top \eta \Delta x_{ij} = -(t_j - t_i)^2 + \|r_j - r_i\|^2. \tag{4}$$

The signature of η distinguishes the temporal direction from the spatial directions, causing them to enter this form with opposite signs. The transformations preserving this bilinear form are *Lorentz transformations*. A matrix Λ is Lorentz if

$$\Lambda^\top \eta \Lambda = \eta, \tag{5}$$

or equivalently,

$$\Lambda^\top \eta = \eta \Lambda^{-1}. \tag{6}$$

This identity will be the key algebraic ingredient for constructing a relative positional interaction.

Notice that in the proposed MinkowskiPE method, the event coordinates themselves are not transformed by Λ. Instead, they parameterize Lorentz transformations applied to query–key features. We therefore use Minkowski geometry as a positional-representation structure rather than imposing Lorentz symmetry on the underlying data.

4 METHOD

Building on the relative position construction in Section 3, we formulate spatiotemporal tokens as events and construct a Lorentzian positional transformation whose pairwise interaction depends only on relative event displacement. Then we extend this construction to practical multi-head attention.

Event-Token Formulation. We consider a sequence of tokens $\{(h_i, x_i)\}_{i=1}^{N}$, where $h_i \in \mathbb{R}^{d_{\text{em}}}$ denotes the embedding of the token i, and x_i denotes the associated spatiotemporal coordinate. We write the coordinate as $x_i = (t_i, r_i)$, with $t_i \in \mathbb{R}$ being the temporal coordinate, $r_i \in \mathbb{R}^d$ being the spatial coordinate. For two events, we define their relative displacement as $\Delta x_{ij} = x_j - x_i$. This event-based representation applies to a broad range of spatiotemporal applications. For example, in video, a token may correspond to an image patch at a particular frame, with $r_i \in \mathbb{R}^2$. In molecular dynamics, a token may correspond to an atom at a particular time, with $r_i \in \mathbb{R}^3$.

4.1 MINKOWSKI POSITIONAL ENCODING

We seek to use the spatiotemporal coordinate x_i to parameterize a Lorentz transformation acting on query and key features. Following Equation (6), we have

$$\Lambda(x_i)^\top \eta \Lambda(x_j) = \eta \Lambda(x_i)^{-1} \Lambda(x_j). \tag{7}$$

To ensure that the positional interaction depends only on relative displacement $\Delta x_{ij} = x_j - x_i$, a sufficient condition is

$$\Lambda(x_1 + x_2) = \Lambda(x_1)\Lambda(x_2), \quad \Lambda(0) = I, \tag{8}$$

which gives $\Lambda(x_i)^{-1}\Lambda(x_j) = \Lambda(x_j - x_i)$.

For multidimensional coordinates, a general Lorentz representation satisfying Equation (8) requires mutually commuting transformations across coordinate directions. To obtain a simple and tractable construction, we restrict it to a *one-parameter* Lorentz subgroup. The event coordinate $x_i \in \mathbb{R}^D$, with $D = 1 + d$, is first mapped to a scalar parameter

$$\rho_i = w^\top x_i, \tag{9}$$

by a learnable linear projection $w \in \mathbb{R}^D$, so that $\rho_j - \rho_i = w^\top \Delta x_{ij}$. We therefore parameterize the transformation associated with event x_i by $\Lambda(\rho_i)$. Appendix A gives a group-theoretic characterization of the construction, showing that the homomorphic formulation above imposes commuting-generator structure and that, for 2D Lorentzian feature blocks, any smooth homomorphism from the additive group into $SO^+(1,1)$ reduces to the scalar-projection one-parameter form used above.

MinkowskiPE. We then instantiate the one-parameter construction above using the standard (1+1)-dimensional Lorentz boost acting on two-dimensional query and key feature blocks

$$\Lambda(\rho) = \begin{bmatrix} \cosh\rho & \sinh\rho \\ \sinh\rho & \cosh\rho \end{bmatrix}, \quad \eta = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}. \tag{10}$$

Let $q_i, k_i \in \mathbb{R}^2$ denote a query and key feature block associated with event x_i, with $\rho_i = w^\top x_i$ defined as above. We define the position-encoded query and key as

$$\tilde{q}_i = \Lambda(\rho_i) q_i, \quad \tilde{k}_i = \eta \Lambda(\rho_i) k_i. \tag{11}$$

The metric factor η in the key transformation allows the standard Euclidean dot product to realize the corresponding Minkowski bilinear interaction. Specifically,

$$\tilde{q}_i^\top \tilde{k}_j = q_i^\top \Lambda(\rho_i)^\top \eta \Lambda(\rho_j) k_j = q_i^\top \eta \Lambda(\rho_j - \rho_i) k_j = q_i^\top \eta \Lambda\big(w^\top \Delta x_{ij}\big)\, k_j. \tag{12}$$

Thus, although each token is transformed according to its own event coordinate, the positional contribution to the pairwise query–key interaction depends on x_i and x_j only through their relative displacement Δx_{ij}, as shown in Equation (12).

4.2 Practical Parameterization

The single-block construction in Section 4.1 extends directly to a full attention head by assigning a separate projection direction to each two-dimensional query–key feature block. For a head dimension d_h, we partition the query and key features into $P = d_h/2$ two-dimensional blocks and introduce a learnable projection matrix

$$W = [w_1, \ldots, w_P] \in \mathbb{R}^{D \times P}, \tag{13}$$

where each column $w_p \in \mathbb{R}^D$ defines the positional projection for the p-th block. The same projection matrix W is shared across all Transformer layers and attention heads.

For coordinate x_i, this produces the block-wise parameters $\rho_i = W^\top x_i \in \mathbb{R}^P$, with $\rho_{i,p} = w_p^\top x_i$. By linearity, $\rho_{j,p} - \rho_{i,p} = w_p^\top \Delta x_{ij}$, so each block receives a different learned scalar projection of the same relative event displacement.

For efficient implementation, the block-wise Lorentz transformations can be written equivalently in element-wise form. For $q_i = (q_{i,1}, \ldots, q_{i,d_h})^\top$, we have

$$\tilde{q}_i = \begin{bmatrix} q_{i,1} \\ q_{i,2} \\ q_{i,3} \\ q_{i,4} \\ \vdots \\ q_{i,d_h-1} \\ q_{i,d_h} \end{bmatrix} \odot \begin{bmatrix} \cosh \rho_{i,1} \\ \cosh \rho_{i,1} \\ \cosh \rho_{i,2} \\ \cosh \rho_{i,2} \\ \vdots \\ \cosh \rho_{i,P} \\ \cosh \rho_{i,P} \end{bmatrix} + \begin{bmatrix} q_{i,2} \\ q_{i,1} \\ q_{i,4} \\ q_{i,3} \\ \vdots \\ q_{i,d_h} \\ q_{i,d_h-1} \end{bmatrix} \odot \begin{bmatrix} \sinh \rho_{i,1} \\ \sinh \rho_{i,1} \\ \sinh \rho_{i,2} \\ \sinh \rho_{i,2} \\ \vdots \\ \sinh \rho_{i,P} \\ \sinh \rho_{i,P} \end{bmatrix}, \tag{14}$$

where $\odot$ denotes element-wise multiplication. The key features are transformed analogously, together with the Minkowski metric factor defined in Section 4.1.

5 Experiments

As illustrated in Figure 2, we evaluate MinkowskiPE on molecular dynamics and video prediction, two spatiotemporal prediction tasks that differ in physical scale and data structure. Molecular dynamics represents atomic motion in three-dimensional space, whereas video prediction represents

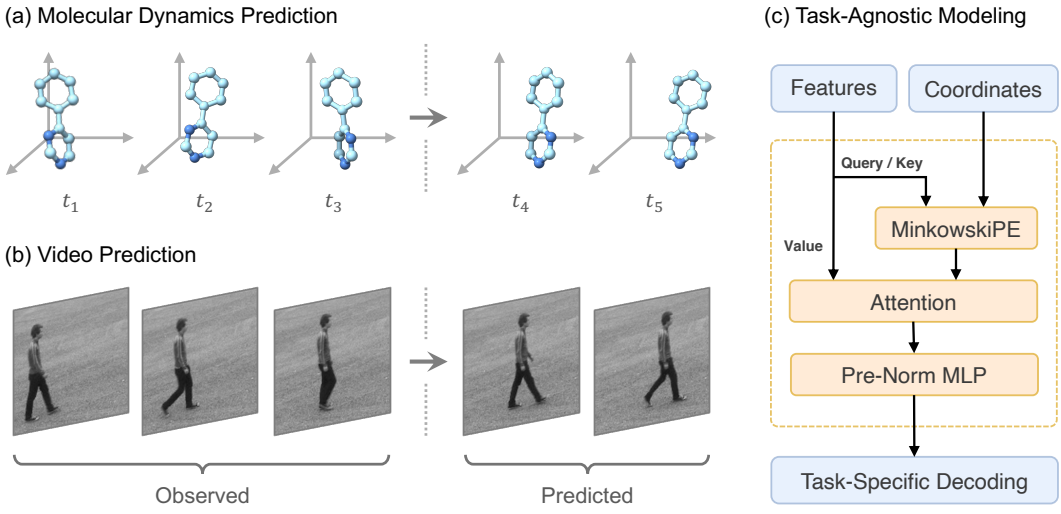

Figure 2: Experimental overview. (a) Molecular dynamics and (b) video predictions instantiate spatiotemporal forecasting at different physical scales. (c) Both tasks follow the same task-agnostic modeling scheme, in which task-specific features are paired with spatiotemporal coordinates, processed by a Transformer with MinkowskiPE, and mapped to future states by task-specific decoders.

visual evolution on a two-dimensional image plane. Despite these differences, both domains can be modeled within the same task-agnostic spatiotemporal framework. We describe the task formulation, model instantiation, and experimental protocol for each domain below.

5.1 Molecular Dynamics Prediction

Task Formulation. We consider the protein–ligand binding dynamics prediction under the semi-flexible setting, where the protein is treated as a fixed environment while the ligand evolves dynamically over time. Given the surrounding protein structure and an observed segment of ligand motion, the task is to predict the future three-dimensional positions of all ligand heavy atoms over multiple time steps. This requires the model to capture both the temporal evolution of individual atoms and the spatial interactions that constrain the ligand trajectory within its molecular environment.

Model Instantiation. For molecular dynamics, each ligand atom at each observed time step is treated as an event token with spatiotemporal coordinate (t, x, y, z), where t denotes the time step and (x, y, z) specifies its three-dimensional position. The token features encode molecular identity and motion information, while the protein is represented as static residue-level structural context. Each Transformer block first performs frame-wise cross-attention from ligand atoms to protein residues, followed by global self-attention over ligand event tokens across the observed trajectory. MinkowskiPE is applied within the attention modules using the associated spatiotemporal coordinates. After the final Transformer block, the representation of each ligand atom at the last observed frame is passed to an MLP decoder, which predicts the displacements over all future frames.

Experimental Setup. We evaluate on the MISATO dataset (Siebenmorgen et al., 2024) under both single-trajectory and multi-trajectory settings. For single-trajectory prediction, a separate model is trained for each of the ten protein–ligand systems, using 10 observed frames to predict the subsequent 20 frames. For multi-trajectory prediction, a separate model is trained on each of MISATO-100, MISATO-1000, and MISATO-Full, which represent progressively larger collections of protein–ligand trajectories, using 80 observed frames to predict the subsequent 20 frames.

We evaluate prediction quality using MAE, Matching, and Stability, which measure coordinate accuracy and structural consistency from complementary perspectives. We compare against VerletMD, GNN-MD, and DenoisingLD, three representative baselines implemented in NeuralMD (Liu et al., 2026) and motivated by numerical integration, autoregressive graph-based simulation (Fu et al., 2023), and diffusion-based molecular dynamics (Arts et al., 2023), respectively. We further include NeuralMD-ODE, NeuralMD-SDE (Liu et al., 2026), and the recent generative model BioMD (Feng et al., 2026). More experimental details are provided in Appendix B.

Table 1: Results on single-trajectory predictions (in10/out20). All results are averaged over three random seeds. The best result in each column is shown in bold, and the second-best is underlined.

Method	Metric	5WIJ	4ZX0	3EOV	4K6W	1KTI	1XP6	4YUR	4G3E	6B7F	3B9S
VerletMD	MAE	14.629	21.278	27.960	15.428	18.157	13.753	16.764	5.111	31.934	19.473
	Matching	5.459	7.971	13.588	7.505	7.467	4.672	9.555	3.388	21.691	0.923
	Stability	24.360	19.168	13.067	15.441	19.352	28.129	16.542	31.852	11.050	57.801
GNN-MD	MAE	2.280	2.370	3.512	3.695	6.641	2.378	7.031	2.709	4.136	2.578
	Matching	0.803	0.555	1.216	1.038	0.386	0.966	0.920	0.893	1.194	1.414
	Stability	54.475	68.613	40.984	42.480	81.831	49.239	47.555	61.802	39.067	49.306
NeuralMD-ODE	MAE	2.252	**1.878**	3.858	3.656	6.675	1.924	6.957	2.191	3.921	3.039
	Matching	0.464	0.428	1.062	0.928	0.337	0.537	0.584	0.505	0.459	0.659
	Stability	82.046	81.401	47.328	49.438	86.430	75.533	69.775	71.436	75.692	76.065
NeuralMD-SDE	MAE	2.260	2.158	**3.395**	3.765	6.646	2.061	7.038	2.345	**3.842**	3.132
	Matching	0.615	0.696	0.962	1.076	0.167	0.615	0.749	0.521	0.741	0.444
	Stability	67.464	59.109	50.108	49.700	98.508	69.423	60.344	68.729	57.917	77.801
DenoisingLD	MAE	2.501	3.138	4.055	3.942	7.051	2.218	7.128	3.588	4.431	2.811
	Matching	0.815	1.072	1.209	0.839	0.268	0.676	0.834	1.069	0.672	0.472
	Stability	52.418	44.228	41.469	53.820	91.986	64.951	49.676	40.823	61.583	71.852
BioMD	MAE	3.191	2.200	4.193	3.362	**4.251**	2.273	**5.163**	2.535	5.156	4.243
	Matching	**0.386**	**0.424**	**0.712**	0.778	0.153	0.544	0.477	0.228	**0.189**	0.095
	Stability	**87.150**	**83.080**	**64.430**	57.680	98.320	78.050	78.840	94.420	**96.530**	**100.000**
MinkowskiPE	MAE	**2.143**	1.982	3.491	**2.477**	6.669	**1.911**	6.491	**1.818**	3.899	**2.486**
	Matching	0.392	0.441	0.989	**0.742**	**0.129**	**0.509**	**0.422**	**0.190**	0.205	**0.071**
	Stability	86.882	81.138	52.287	**58.499**	**99.383**	**79.397**	**81.934**	**97.325**	95.825	**100.000**

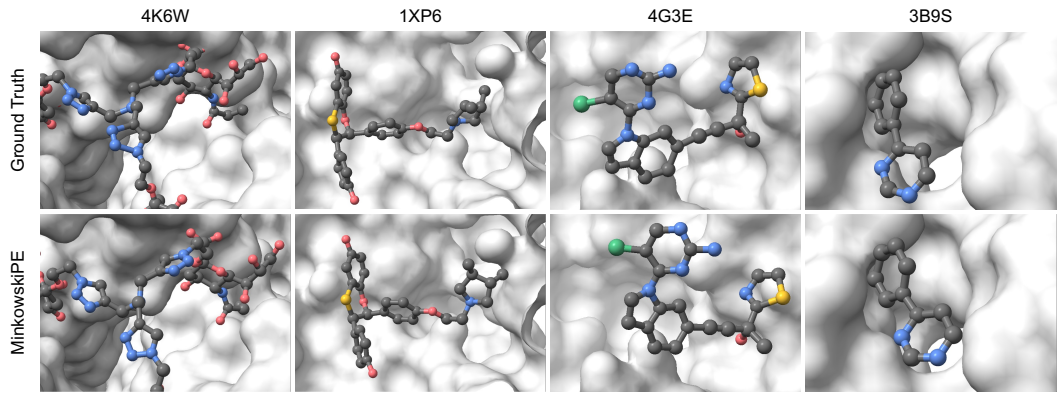

Figure 3: Representative molecular dynamics predictions. Final-step ligand configurations predicted by MinkowskiPE for 4K6W, 1XP6, 4G3E, and 3B9S, compared with the corresponding reference configurations. The surrounding protein surface is locally clipped for visualization clarity.

Table 2: Results on multi-trajectory predictions (in80/out20). All results are averaged over three random seeds. The best result in each column is shown in bold, and the second-best is underlined.

Method	MISATO-100			MISATO-1000			MISATO-Full		
	MAE	Matching	Stability	MAE	Matching	Stability	MAE	Matching	Stability
VerletMD	14.317	6.927	19.78	27.374	10.042	20.10	24.242	10.101	18.76
GNN-MD	4.198	0.489	77.67	4.500	0.553	75.06	4.434	0.509	77.59
NeuralMD-ODE	4.210	0.407	84.76	4.548	0.446	84.69	4.448	0.428	84.90
NeuralMD-SDE	4.215	0.407	84.71	4.555	0.446	84.59	4.455	0.430	84.70
DenoisingLD	<u>4.184</u>	<u>0.396</u>	<u>85.39</u>	<u>4.467</u>	<u>0.430</u>	<u>85.42</u>	4.400	<u>0.412</u>	<u>85.64</u>
BioMD	4.747	0.471	80.28	4.522	0.472	82.47	<u>4.390</u>	0.420	85.15
MinkowskiPE	**4.181**	**0.395**	**85.45**	**4.367**	**0.415**	**86.01**	**4.315**	**0.392**	**86.29**

Results. Table 1 reports the single-trajectory results. Across the ten protein–ligand systems and three evaluation metrics, MinkowskiPE achieves the best result in 17 of 30 evaluations and ranks among the top two in 26 of 30. In particular, it achieves the best performance across all three metrics on 4K6W, 1XP6, 4G3E, and 3B9S. Its strong performance on both Matching and Stability indicates that the gains are not confined to coordinate accuracy, but also extend to the structural quality of the predicted ligand configurations. Figure 3 visualizes the final predicted configurations for these four systems alongside the corresponding ground truth, providing a direct view of the resulting molecular structures.

Table 2 reports the multi-trajectory results. Across all three dataset scales, MinkowskiPE achieves the best performance on all nine evaluations. The advantage is consistent across both coordinate and structural metrics and persists as the number and diversity of training trajectories increase. Together with the single-trajectory results, these findings show that MinkowskiPE performs well not only when modeling individual molecular systems, but also when learning shared dynamics across heterogeneous protein–ligand complexes.

5.2 Video Prediction

Task Formulation. We next consider video prediction, where a model observes a sequence of past frames and forecasts the subsequent visual evolution. Unlike molecular dynamics, where spatial coordinates correspond to continuous three-dimensional atom positions, video observations are organized on a regular two-dimensional image plane. The task therefore requires modeling both temporal changes across frames and spatial structure within each frame, including the motion and deformation of visual patterns over time.

Model Instantiation. Each video frame is first encoded by a convolutional encoder into a 16×16 latent feature grid. Every latent feature is treated as an event token with spatiotemporal coordinate

Table 3: Results on KTH video prediction (in10/out20). We adopt the baseline results and experimental settings from OpenSTL (Tan et al., 2023b), which report the best result across three random seeds. For protocol-matched comparison, bold and underline denote the best and second-best results using our best run. The three-seed mean of MinkowskiPE is additionally reported for reference.

Method	Params (M)	MSE ↓	MAE ↓	SSIM ↑	PSNR ↑	LPIPS ↓
ConvLSTM-S	14.9	47.65	445.5	0.8977	26.99	0.26686
MIM	39.8	40.73	380.8	0.9025	27.78	<u>0.18808</u>
PredRNN	23.6	41.07	380.6	0.9097	27.95	0.21892
PredRNN++	38.3	39.84	370.4	**0.9124**	<u>28.13</u>	0.19871
PredRNN.V2	23.6	<u>39.57</u>	<u>368.8</u>	<u>0.9099</u>	28.01	0.21478
SimVP+IncepU	12.2	41.11	397.1	0.9065	27.46	0.26496
SimVP+gSTA	15.6	45.02	417.8	0.9049	27.04	0.25240
TAU	15.0	45.32	421.7	0.9086	27.10	0.22856
MinkowskiPE (best)	2.5	**35.66**	**347.7**	0.9078	**28.18**	**0.18490**
MinkowskiPE (mean)	2.5	35.78	347.9	0.9079	28.15	0.18342

(t, u, v), where t denotes the frame index and (u, v) specifies its spatial location on the latent grid. The observed tokens are concatenated with future tokens initialized from the last observed latent map, and augmented with learnable future-step embeddings. All tokens are then jointly processed by an eight-layer global Transformer equipped with MinkowskiPE. The resulting future latent features are reshaped into spatial feature maps and passed through a convolutional decoder to reconstruct all future frames.

Experimental Setup. We evaluate on the KTH human-action dataset following the OpenSTL protocol (Tan et al., 2023b). The model takes 10 grayscale frames at 128×128 resolution as input and predicts the subsequent 20 frames. We evaluate prediction quality using MSE, MAE, SSIM, PSNR, and LPIPS, covering pixel-level accuracy, structural similarity, and perceptual quality.

We compare against representative video-prediction baselines reported in OpenSTL, including ConvLSTM (Shi et al., 2015), MIM (Wang et al., 2019), PredRNN variants (Wang et al., 2017; 2018; 2023), SimVP variants (Gao et al., 2022a; Tan et al., 2025), and TAU (Tan et al., 2023a). To match its reporting protocol, which selects the best model over three trials, we use our lowest-MSE run among three random seeds for the primary comparison with OpenSTL baselines. We additionally report the mean performance across the three seeds for completeness. More experimental details are provided in Appendix B.

Results. Table 3 reports the results on the KTH video prediction task. Under the protocol-matched comparison, MinkowskiPE achieves the best performance on MSE, MAE, PSNR, and LPIPS, while remaining competitive on SSIM. Compared with the strongest baseline result for each metric, it reduces MSE, MAE, and LPIPS by 9.9%, 5.7%, and 1.7%, respectively, while using only 2.5M parameters. The three-seed mean remains better than the reported baseline results on the same four metrics. Overall, these results show that the same event-based positional formulation used for molecular dynamics also performs effectively on dense visual prediction with a different spatial dimensionality and token structure.

5.3 Effect of MinkowskiPE

To isolate the contribution of the proposed positional mechanism from the underlying task-specific architectures, we conduct controlled ablations on both molecular dynamics and video prediction. In each case, we remove only the positional encoding from the corresponding attention modules, and keep the model architecture, optimization protocol, and training configuration unchanged. As shown in Table 4, MinkowskiPE improves all evaluated metrics in both domains. On MISATO-1000, it reduces MAE and Matching by 2.7% and 3.9% respectively, and improves Stability by 1.2%. On KTH, it reduces MSE and MAE by 5.7% and 3.6% respectively, and improves LPIPS by 4.5%, with consistent gains in SSIM and PSNR. These controlled comparisons show that the improvements extend across multiple evaluation metrics and two different spatiotemporal domains, supporting that the observed gains are not solely attributable to the task-specific architectures.

Table 4: Controlled ablation of MinkowskiPE on molecular dynamics and video prediction. We compare the same model architecture with MinkowskiPE and with positional encoding removed, while keeping all others unchanged. All results are averaged over three random seeds.

Method	MISATO-1000			KTH				
	MAE ↓	Matching ↓	Stability ↑	MSE ↓	MAE ↓	SSIM ↑	PSNR ↑	LPIPS ↓
No PE	4.488	0.432	84.96	37.93	360.8	0.9045	27.93	0.19203
MinkowskiPE	**4.367**	**0.415**	**86.01**	**35.78**	**347.9**	**0.9079**	**28.15**	**0.18342**

6 Discussion

Minkowski Structure as Representation Bias. MinkowskiPE does not assume that the underlying data obey relativistic dynamics. Rather, it uses the algebraic structure of Minkowski geometry as a representation-level inductive bias for spatiotemporal attention. Joint temporal and spatial coordinates parameterize Lorentz transformations in query–key feature space. Together with the linear coordinate projections, preservation of the indefinite bilinear form ensures that the positional interaction depends only on relative event displacement. This construction provides a geometric mechanism for jointly modeling temporal and spatial information without prescribing a dynamical law or requiring Lorentz symmetry of the underlying data.

Cross-domain behavior. A notable property of MinkowskiPE is that the same positional construction can be applied across substantially different spatiotemporal domains without changing its underlying formulation. Molecular dynamics and video prediction differ not only in physical scale, but also in token semantics, spatial structure, and prediction architecture. Nevertheless, both can be represented through the same event-based view of time and space and modeled with the same positional mechanism. This suggests that MinkowskiPE provides a flexible geometric representation for spatiotemporal relations across different Transformer-based prediction settings.

Limitations and Outlook. MinkowskiPE adopts a fixed geometric bias rather than learning the geometry from data, and different applications may benefit from different geometric inductive biases. Importantly, this choice does not assume that the underlying dynamics obey Lorentz symmetry. If one instead seeks a positional encoding that explicitly respects the spacetime symmetry of a physical system, Galilean symmetry provides a natural candidate for many non-relativistic settings (Lévy-Leblond, 1971). In Appendix C, we analyze one such construction based on finite-dimensional unitary token-wise transformations and show that exact Galilean invariance imposes a strong structural constraint: the resulting pairwise interaction cannot retain the Galilean-invariant spatial component of the relative group element. This result does not rule out Galilean positional encodings more generally, but highlights a nontrivial obstruction for this particular class of constructions.

A second limitation is that our current construction uses learned scalar projections followed by one-parameter Lorentz transformations. More expressive multi-parameter constructions may capture richer interactions among temporal and spatial directions, but require mutually commuting Lorentz generators and are therefore substantially more constrained. We provide further analysis in Appendix A. Finally, our experiments focus on molecular and visual forecasting. Extending the framework to embodied dynamics, weather forecasting, and generative models, as well as exploring data-adaptive geometric structures, are natural directions for future work.

7 Conclusion

We introduced MinkowskiPE, a relative positional encoding that represents Transformer tokens as events in joint time–space coordinates. Using Lorentz transformations in query–key feature space, it couples temporal and spatial positional information, with attention scores depending on position only through relative spacetime displacement. The construction requires only a lightweight modification to query–key transformations and applies unchanged across different spatiotemporal tokenizations. Across molecular dynamics and video prediction, MinkowskiPE achieves strong performance across distinct spatiotemporal settings. Together, these results highlight positional geometry as a useful design axis for spatiotemporal Transformers.

AI USE STATEMENT

The authors used ChatGPT to polish the writing and assist in literature discovery and interpretation. CodeX was used for limited code review, debugging, and utility scripts, while the core implementation was developed by the authors. During experimentation, CodeX assisted with monitoring, aggregating, and organizing experimental results. All AI-assisted material was reviewed and verified by the authors, who take full responsibility for the final manuscript, code, and reported results.

REFERENCES

Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. ViViT: A video vision transformer. In *2021 IEEE/CVF international conference on computer vision (ICCV)*, 2021.

Marloes Arts, Victor Garcia Satorras, Chin-Wei Huang, Daniel Zugner, Marco Federici, Cecilia Clementi, Frank Noé, Robert Pinsler, and Rianne van den Berg. Two for one: Diffusion models and force fields for coarse-grained molecular dynamics. *Journal of Chemical Theory and Computation*, 19(18):6151–6159, 2023.

Valentine Bargmann. On unitary ray representations of continuous groups. *Annals of Mathematics*, 59(1):1–46, 1954.

Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *Proceedings of the 38th International Conference on Machine Learning*, 2021.

Sean M. Carroll. *Spacetime and Geometry: An Introduction to General Relativity*. Cambridge University Press, 2019.

Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Xiangxiang Chu, Jianlin Su, Bo Zhang, and Chunhua Shen. VisionLLaMA: A unified llama backbone for vision tasks. In *European Conference on Computer Vision*, 2024.

Chang Dai, Hongyu Shan, Mingyang Song, and Di Liang. HoPE: Hyperbolic rotary positional encoding for stable long-range dependency modeling in large language models. *arXiv preprint arXiv:2509.05218*, 2025.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime G Carbonell, Quoc Le, and Ruslan Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pp. 2978–2988, 2019.

Tri Dao, Daniel Y Fu, Stefano Ermon, Atri Rudra, and Christopher Re. FlashAttention: Fast and memory-efficient exact attention with IO-awareness. In *Advances in Neural Information Processing Systems*, 2022.

Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. Multiscale vision transformers. In *2021 IEEE/CVF international conference on computer vision (ICCV)*, 2021.

Bin Feng, Jiying Zhang, Xinni Zhang, Zijing Liu, and Yu Li. BioMD: All-atom generative model for biomolecular dynamics simulation. In *The Fourteenth International Conference on Learning Representations*, 2026.

Xiang Fu, Tian Xie, Nathan J. Rebello, Bradley Olsen, and Tommi S. Jaakkola. Simulate time-integrated coarse-grained molecular dynamics with multi-scale graph networks. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856.

Zhangyang Gao, Cheng Tan, Lirong Wu, and Stan Z Li. SimVP: Simpler yet better video prediction. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022a.

Zhihan Gao, Xingjian Shi, Hao Wang, Yi Zhu, Yuyang (Bernie) Wang, Mu Li, and Dit-Yan Yeung. Earthformer: Exploring space-time transformers for earth system forecasting. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022b.

Samuel Greydanus, Misko Dzamba, and Jason Yosinski. Hamiltonian neural networks. In *Advances in Neural Information Processing Systems*, 2019.

Vincent Le Guen and Nicolas Thome. Disentangling physical dynamics from unknown factors for unsupervised video prediction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020.

Byeongho Heo, Song Park, Dongyoon Han, and Sangdoo Yun. Rotary position embedding for vision transformer. In *European Conference on Computer Vision*. Springer, 2024.

Jean-Marc Lévy-Leblond. Galilei group and galilean invariance. In *Group theory and its applications*, pp. 221–299. Elsevier, 1971.

Haiping Liu, Lijing Lin, Jingyuan Sun, Zhegong Shangguan, Mauricio A. Alvarez, and Hongpeng Zhou. Rethinking rope: A mathematical blueprint for n-dimensional positional embedding. *arXiv preprint arXiv:2504.06308*, 2025a.

Shengchao Liu, Weitao Du, Hannan Xu, Yanjing Li, Zhuoxinran Li, Vignesh Bhethanabotla, Divin Yan, Christian Borgs, Anima Anandkumar, Hongyu Guo, et al. A multi-grained symmetric differential equation model for learning protein-ligand binding dynamics. *Nature Communications*, 17(1):1049, 2026.

Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video Swin Transformer. In *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, 2022.

Zikang Liu, Longteng Guo, Yepeng Tang, Tongtian Yue, Junxian Cai, Kai Ma, Qingbin Liu, Xi Chen, and Jing Liu. VRoPE: Rotary position embedding for video large language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, 2025b.

Zeyu Lu, ZiDong Wang, Di Huang, Chengyue Wu, Xihui Liu, Wanli Ouyang, and LEI BAI. FiT: Flexible vision transformer for diffusion model. In *Forty-first International Conference on Machine Learning*, 2024.

Sophie Ostmeier, Brian Axelrod, Maya Varma, Michael Moseley, Akshay S Chaudhari, and Curtis Langlotz. LieRE: Lie rotational positional encodings. In *Forty-second International Conference on Machine Learning*, 2025.

Mandela Patrick, Dylan Campbell, Yuki Asano, Ishan Misra, Florian Metze, Christoph Feichtenhofer, Andrea Vedaldi, and Joao F. Henriques. Keeping your eye on the ball: Trajectory attention in video transformers. In *Advances in Neural Information Processing Systems*, 2021.

Tobias Pfaff, Meire Fortunato, Alvaro Sanchez-Gonzalez, and Peter Battaglia. Learning mesh-based simulation with graph networks. In *International Conference on Learning Representations*, 2021.

Ofir Press, Noah Smith, and Mike Lewis. Train short, test long: Attention with linear biases enables input length extrapolation. In *International Conference on Learning Representations*, 2022.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67, 2020.

M. Raissi, P. Perdikaris, and G.E. Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019.

Alvaro Sanchez-Gonzalez, Jonathan Godwin, Tobias Pfaff, Rex Ying, Jure Leskovec, and Peter Battaglia. Learning to simulate complex physics with graph networks. In *Proceedings of the 37th International Conference on Machine Learning*, 2020.

Connor Schenck, Isaac Reid, Mithun George Jacob, Alex Bewley, Joshua Ainslie, David Rendleman, Deepali Jain, Mohit Sharma, Kumar Avinava Dubey, Ayzaan Wahid, Sumeet Singh, René Wagner, Tianli Ding, Chuyuan Fu, Arunkumar Byravan, Jake Varley, Alexey A. Gritsenko, Matthias Minderer, Dmitry Kalashnikov, Jonathan Tompson, Vikas Sindhwani, and Krzysztof Marcin Choromanski. Learning the roPEs: Better 2D and 3D position encodings with STRING. In *Forty-second International Conference on Machine Learning*, 2025.

Christian Schuldt, Ivan Laptev, and Barbara Caputo. Recognizing human actions: a local svm approach. In *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, volume 3, pp. 32–36. IEEE, 2004. doi: 10.1109/ICPR.2004.1334462.

Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. Self-attention with relative position representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pp. 464–468, 2018.

Xingjian Shi, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In *Proceedings of the 29th International Conference on Neural Information Processing Systems - Volume 1*, 2015.

Till Siebenmorgen, Filipe Menezes, Sabrina Benassou, Erinc Merdivan, Kieran Didi, André Santos Dias Mourão, Radosław Kitel, Pietro Liò, Stefan Kesselheim, Marie Piraud, et al. MISATO: machine learning dataset of protein–ligand complexes for structure-based drug discovery. *Nature computational science*, 4(5):367–378, 2024.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

Cheng Tan, Zhangyang Gao, Lirong Wu, Yongjie Xu, Jun Xia, Siyuan Li, and Stan Z Li. Temporal attention unit: Towards efficient spatiotemporal predictive learning. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023a.

Cheng Tan, Siyuan Li, Zhangyang Gao, Wenfei Guan, Zedong Wang, Zicheng Liu, Lirong Wu, and Stan Z. Li. OpenSTL: A comprehensive benchmark of spatio-temporal predictive learning. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023b.

Cheng Tan, Zhangyang Gao, Siyuan Li, and Stan Z. Li. SimVPv2: Towards simple yet powerful spatiotemporal predictive learning. *IEEE Transactions on Multimedia*, 27:5170–5184, 2025.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. Qwen2-VL: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

Yunbo Wang, Mingsheng Long, Jianmin Wang, Zhifeng Gao, and Philip S Yu. PredRNN: Recurrent neural networks for predictive learning using spatiotemporal lstms. In *Advances in Neural Information Processing Systems*, 2017.

Yunbo Wang, Zhifeng Gao, Mingsheng Long, Jianmin Wang, and Philip S Yu. PredRNN++: Towards a resolution of the deep-in-time dilemma in spatiotemporal predictive learning. In *International conference on machine learning*, 2018.

Yunbo Wang, Jianjin Zhang, Hongyu Zhu, Mingsheng Long, Jianmin Wang, and Philip S Yu. Memory in memory: A predictive neural network for learning higher-order non-stationarity from spatiotemporal dynamics. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Yunbo Wang, Haixu Wu, Jianjin Zhang, Zhifeng Gao, Jianmin Wang, Philip S. Yu, and Mingsheng Long. PredRNN: A recurrent neural network for spatiotemporal predictive learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(2):2208–2225, 2023.

Xilin Wei, Xiaoran Liu, Yuhang Zang, Xiaoyi Dong, Pan Zhang, Yuhang Cao, Jian Tong, Haodong Duan, Qipeng Guo, Jiaqi Wang, Xipeng Qiu, and Dahua Lin. VideoRoPE: What makes for good video rotary position embedding? In *Forty-second International Conference on Machine Learning*, 2025.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. In *International Conference on Learning Representations*, 2025.

Yupu Yao and Bowen Yang. GeoPE:a unified geometric positional embedding for structured tensors. *arXiv preprint arXiv:2512.04963*, 2025.

Hao Yu, Tangyu Jiang, Shuning Jia, Shannan Yan, Shunning Liu, Haolong Qian, Guanghao Li, Shuting Dong, and Chun Yuan. ComRoPE: Scalable and robust rotary position embedding parameterized by trainable commuting angle matrices. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

Yifan Zhang, Zixiang Chen, Yifeng Liu, Zhen Qin, Huizhuo Yuan, Kangping Xu, Yang Yuan, Quanquan Gu, and Andrew C Yao. Group representational position encoding. In *The Fourteenth International Conference on Learning Representations*, 2026.

A Additional Derivations of MinkowskiPE

This appendix develops the group-theoretic structure underlying the construction in Section 4. We first formalize how a homomorphic Lorentzian representation yields an exact relative-position interaction. And then we characterize general smooth homomorphisms from the additive spatiotemporal coordinate group into a Lorentz group, and show that for the $(1+1)$-dimensional feature blocks used by MinkowskiPE, the construction necessarily reduces to a linear scalar projection followed by a one-parameter Lorentz boost.

A.1 Relative Lorentzian Construction

Let $x \in \mathbb{R}^D$ denote an event coordinate, and let $\Gamma : (\mathbb{R}^D, +) \to SO^+(1, d_s)$ denote a position-dependent Lorentz representation acting on a $(1+d_s)$-dimensional feature space. We write

$$\eta_s = \mathrm{diag}(-1, 1, \ldots, 1) \tag{15}$$

for the Minkowski metric on this feature space. By definition,

$$\Gamma(x)^\top \eta_s \Gamma(x) = \eta_s, \tag{16}$$

and hence

$$\Gamma(x)^\top \eta_s = \eta_s \Gamma(x)^{-1}. \tag{17}$$

For query and key features q_i and k_j associated with event coordinates x_i and x_j, define

$$\tilde{q}_i = \Gamma(x_i) q_i, \quad \tilde{k}_j = \eta_s \Gamma(x_j) k_j. \tag{18}$$

Their Euclidean dot product becomes

$$\tilde{q}_i^\top \tilde{k}_j = q_i^\top \Gamma(x_i)^\top \eta_s \Gamma(x_j) k_j = q_i^\top \eta_s \Gamma(x_i)^{-1} \Gamma(x_j) k_j, \tag{19}$$

where the second equality follows from Equation (17).

Suppose that Γ is a homomorphism from the additive coordinate group into the Lorentz group,

$$\Gamma(x_1 + x_2) = \Gamma(x_1)\Gamma(x_2), \quad \Gamma(0) = I. \tag{20}$$

Then $\Gamma(-x) = \Gamma(x)^{-1}$, and therefore

$$\Gamma(x_i)^{-1}\Gamma(x_j) = \Gamma(x_j - x_i) = \Gamma(\Delta x_{ij}). \tag{21}$$

Substituting this relation into Equation (19) gives

$$\tilde{q}_i^\top \tilde{k}_j = q_i^\top \eta_s \Gamma(\Delta x_{ij}) k_j, \tag{22}$$

so the positional contribution depends on the event coordinates only through their relative displacement. The metric factor η_s in the key transformation is essential to this construction. If both query and key were transformed simply as $\Gamma(x_i)q_i$ and $\Gamma(x_j)k_j$, their interaction would instead contain $\Gamma(x_i)^\top \Gamma(x_j)$. Lorentz transformations are generally not orthogonal under the ordinary Euclidean inner product, so $\Gamma(x)^\top \neq \Gamma(x)^{-1}$. The insertion of η_s converts the transpose into the group inverse through Equation (17), allowing the two absolute transformations to reduce exactly to a transformation of the relative coordinate.

A.2 Group-Theoretic Characterization

We next characterize smooth Lorentz representations satisfying Equation (20). Let $\Gamma : (\mathbb{R}^D, +) \to SO^+(1, d_s)$ be a smooth homomorphism, where D is the dimension of the event coordinate and $1 + d_s$ is the dimension of the Lorentzian feature representation.

For each coordinate direction, define the infinitesimal generator

$$G_\mu = \frac{\partial}{\partial x_\mu} \Gamma(x) \Big|_{x=0}, \quad \mu = 1, \ldots, D. \tag{23}$$

Since every $\Gamma(x)$ preserves η_s, each generator belongs to the Lorentz Lie algebra and satisfies

$$G_\mu^\top \eta_s + \eta_s G_\mu = 0. \tag{24}$$

Because $(\mathbb{R}^D, +)$ is Abelian, the images of different coordinate directions commute

$$\Gamma(se_\mu)\Gamma(te_\nu) = \Gamma(te_\nu)\Gamma(se_\mu) \tag{25}$$

for arbitrary $s, t \in \mathbb{R}$. Differentiating at the identity gives

$$[G_\mu, G_\nu] = 0 \quad \text{for all } \mu, \nu. \tag{26}$$

Each coordinate direction therefore generates a one-parameter subgroup,

$$\Gamma(se_\mu) = \exp(sG_\mu). \tag{27}$$

Since $x = \sum_{\mu=1}^{D} x_\mu e_\mu$, the homomorphism property gives

$$\Gamma(x) = \prod_{\mu=1}^{D} \exp(x_\mu G_\mu). \tag{28}$$

Using Equation (26), this becomes

$$\Gamma(x) = \exp\left(\sum_{\mu=1}^{D} x_\mu G_\mu\right). \tag{29}$$

Conversely, any collection of mutually commuting Lorentz generators $\{G_\mu\}_{\mu=1}^{D}$ defines a smooth homomorphism through Equation (29). Thus, within this homomorphic construction, an exact multidimensional relative Lorentzian representation is characterized by a mutually commuting set of Lorentz generators.

The (1+1)-dimensional Case. MinkowskiPE operates on two-dimensional Lorentzian feature blocks. For such a block, the relevant connected Lorentz group is $SO^+(1,1)$, whose Lie algebra $\mathfrak{so}(1,1)$ is one-dimensional. A basis generator is

$$G = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad \eta_{1,1} = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}, \tag{30}$$

which satisfies Equation (24).

Because $\mathfrak{so}(1,1)$ is one-dimensional, every generator G_μ must be proportional to G, *i.e.*,

$$G_\mu = w_\mu G \tag{31}$$

for some scalar coefficient w_μ. Substituting this relation into Equation (29) gives

$$\Gamma(x) = \exp\left(\sum_{\mu=1}^{D} x_\mu w_\mu G\right) = \exp\left[(w^\top x)G\right]. \tag{32}$$

Defining $\rho = w^\top x$, we therefore obtain

$$\Gamma(x) = \Lambda(\rho) = \Lambda(w^\top x), \tag{33}$$

where Λ denotes the corresponding one-parameter Lorentz boost.

Since $G^2 = I$,

$$\Lambda(\rho) = e^{\rho G} = \cosh(\rho)I + \sinh(\rho)G = \begin{bmatrix} \cosh\rho & \sinh\rho \\ \sinh\rho & \cosh\rho \end{bmatrix}. \tag{34}$$

Equation (33) gives a structural characterization of the scalar-projection form used in MinkowskiPE. Within a (1+1)-dimensional Lorentzian feature block, any smooth homomorphism from the additive event-coordinate group into $SO^+(1,1)$ is of the form

$$x \longmapsto \rho = w^\top x \longmapsto \Lambda(\rho),$$

including the trivial case $w = 0$. Thus, the learned scalar projection in MinkowskiPE is not merely an implementation convenience but the general smooth homomorphic parameterization within this feature-block class.

Higher-dimensional Lorentz Representations. The one-parameter construction above is not the only possible homomorphic Lorentz representation. In higher-dimensional feature spaces, genuinely multi-parameter constructions can be obtained from higher-dimensional commuting subalgebras of $\mathfrak{so}(1, d_s)$. However, the commutation requirement places strong structural constraints on the allowed generators.

For example, let K_a denote the Lorentz boost generator along the a-th positive-signature direction and J_{ab} the generator of a rotation in the (a, b) plane. For $a \neq b$,

$$[K_a, K_b] = -J_{ab}, \tag{35}$$

up to convention-dependent signs. Hence independent boosts along different directions do not commute and cannot directly serve as independent generators of a homomorphism from $(\mathbb{R}^D, +)$.

More generally, for noncommuting generators A and B, the Baker–Campbell–Hausdorff formula gives

$$e^A e^B = \exp\left(A + B + \frac{1}{2}[A, B] + \cdots\right), \tag{36}$$

so composition introduces additional commutator terms and the simple additive relation

$$\Gamma(x_1)\Gamma(x_2) = \Gamma(x_1 + x_2) \tag{37}$$

does not generally hold.

Higher-dimensional commuting constructions remain possible, including subgroups that act only within positive-signature directions. Our choice of $(1 + 1)$-dimensional Lorentzian blocks instead gives the minimal feature-space realization that mixes the two metric signatures while admitting an exact one-parameter homomorphic form. More expressive multi-parameter Lorentzian positional encodings would require additional choices of commuting subalgebras and are left for future work.

B Additional Experimental Details

B.1 Implementation Details of MinkowskiPE

MinkowskiPE Implementation. In all experiments, MinkowskiPE is applied to the query and key features, while the value features remain unchanged. For each attention head, adjacent feature dimensions are grouped into two-dimensional blocks and transformed using the Lorentzian construction described in Section 4. The same projection matrix W is shared across all Transformer layers and attention heads. For molecular dynamics, the hidden dimension is 128 with 8 attention heads, giving a head dimension of 16 and $P = 8$ Lorentz blocks. For video prediction, the hidden dimension is 144 with 8 attention heads, giving a head dimension of 18 and $P = 9$ blocks.

Coordinate Normalization. Before applying MinkowskiPE, event coordinates are normalized to task-dependent ranges. For molecular dynamics, all spatial coordinates within an input window are first centered by the mass-weighted ligand center of mass (COM) at the last observed frame. Let

$$r_{\max} = \max_{\mathbf{r} \in \mathcal{R}} \left\| \mathbf{r} - \mathbf{r}_{\mathrm{COM}}^{\mathrm{last}} \right\|_2, \tag{38}$$

where $\mathcal{R}$ contains all valid ligand positions in the window and the alpha-carbon (Cα) positions of the cropped protein residues. The spatial coordinates used by MinkowskiPE are then

$$\mathbf{r}_{\mathrm{PE}} = \frac{\mathbf{r} - \mathbf{r}_{\mathrm{COM}}^{\mathrm{last}}}{\max(10, r_{\max}/2)}, \tag{39}$$

while the temporal coordinates over the observed frames are linearly mapped to $[-1, 1]$. For video, the temporal and two spatial coordinates are directly constructed on a regular grid in $[-1, 1]^3$. This normalization keeps the inputs in a numerically well-behaved regime for the hyperbolic functions.

Numerical Implementation. The coordinate projection and the evaluation of $\cosh$ and $\sinh$ are performed in float32, after which the transformed query and key features are cast back to the attention dtype. We implement attention using PyTorch scaled dot-product attention. On the NVIDIA H200 GPUs used in our experiments, this operation is dispatched to the FlashAttention backend.

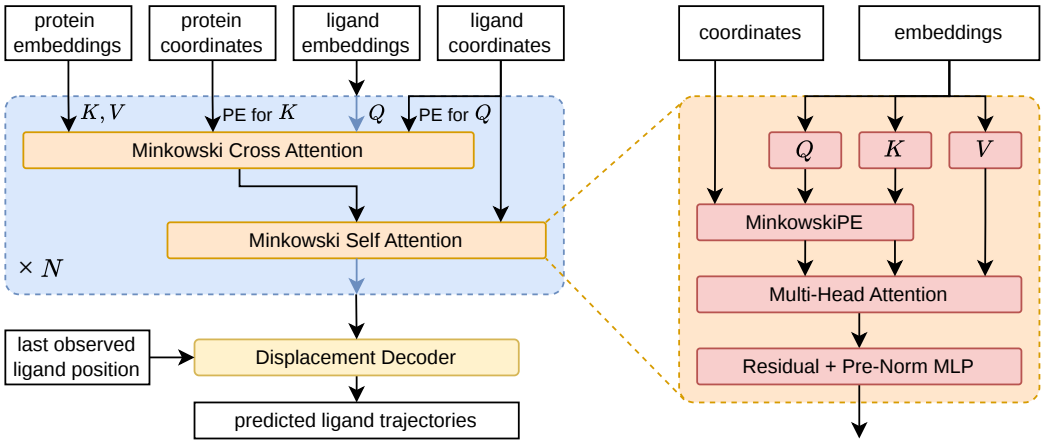

Figure 4: Architecture used for molecular dynamics prediction. Each block performs frame-wise cross-attention from ligand atoms to the surrounding protein residues, followed by global self-attention over ligand event tokens across the observed trajectory. MinkowskiPE is applied to the query–key interactions using the corresponding spatiotemporal event coordinates. The representations at the last observed ligand frame are decoded into future atomic displacements.

B.2 Molecular Dynamics Prediction

Data and Preprocessing. We use protein–ligand trajectories from MISATO (Siebenmorgen et al., 2024) and follow the preprocessing pipeline of NeuralMD (Liu et al., 2026). Each preprocessed trajectory contains 100 stored frames. Ligand inputs contain heavy atoms only, whereas the protein is represented at the residue level using backbone N, Cα, and C atoms. For each input window, we retain at most 128 protein residues whose Cα atoms are closest to the mass-weighted ligand center of mass at the last observed frame. The same center is used to translate the observed ligand, target ligand, and protein coordinates.

Ligand content features include atomic number, atomic mass, centered coordinates, normalized time, frame-to-frame velocity, and a ligand-type embedding. Protein content features include residue identity, the local backbone vectors N–Cα and C–Cα, the centered Cα coordinate, and a protein-type embedding. No geometric data augmentation is applied. Although the protein structure is static, its positional coordinate in frame-wise cross-attention is paired with the temporal coordinate of the corresponding ligand frame. Specifically, a protein residue with Cα position (x_j, y_j, z_j) is assigned the event coordinate (t_i, x_j, y_j, z_j) when attending to ligand atoms at frame t_i.

Single- and Multi-trajectory Settings. For single-trajectory prediction, we follow the NeuralMD protocol and train a separate model for each of the ten protein–ligand systems reported in Table 1. Training windows contain 10 observed frames and 20 target frames, following the same in10/out20 split and protocol as NeuralMD.

For multi-trajectory prediction, we use MISATO-100, MISATO-1000, and MISATO-Full. After preprocessing, the three datasets contain 80/10/10, 800/100/100, and 13,066/1,357/1,357 training/validation/test complexes, respectively. Each trajectory contributes one window containing 80 observed frames followed by 20 target frames.

Model Architecture. The molecular model consists of four repeated blocks, each containing frame-wise ligand-to-protein cross-attention followed by global self-attention over all ligand event tokens in the observed trajectory, as illustrated in Figure 4. The hidden dimension is 128, with 8 attention heads and a 512-dimensional feed-forward layer. Both the attention and feed-forward sublayers use pre-normalization and residual connections, with dropout 0.1. A final LayerNorm is applied after the Transformer stack.

After the final block, only the representation of each ligand atom at the last observed frame is passed to the prediction head. The displacement decoder consists of $128 \to 512 \to 60$ with a GELU nonlinearity and dropout 0.1. The 60 outputs correspond to the three-dimensional displacements of

the 20 future frames. All future frames are predicted in parallel, and the predicted displacements are added to the last observed ligand positions.

Optimization Objective. The model is trained using a combination of coordinate prediction error and pairwise structural consistency. For a predicted ligand trajectory $\hat{\mathbf{r}}_{f,i}$ and target trajectory $\mathbf{r}_{f,i}$, we use

$$\mathcal{L} = \mathcal{L}_{\text{coord}} + 2\mathcal{L}_{\text{pair}}, \quad \text{where} \quad \mathcal{L}_{\text{coord}} = \frac{\sum_{f,i} \|\hat{\mathbf{r}}_{f,i} - \mathbf{r}_{f,i}\|_1}{\sum_f N_f}, \tag{40}$$

and $\mathcal{L}_{\text{pair}}$ is the mean SmoothL1 loss, with $\beta = 0.5$, between predicted and target inter-atomic distances over all valid unordered atom pairs $i < j$ and future frames.

Training Details. We train all molecular dynamics models with Adam, zero weight decay, and a linear warmup followed by cosine learning-rate decay. The backbone is warmed up for two epochs, while the decoder learning rate is applied without warmup. For the single-trajectory setting, both the backbone and decoder use a learning rate of 10^{-4}, and four temporal windows are accumulated per optimizer step. For the multi-trajectory setting, the backbone learning rate is 3×10^{-5}, the decoder learning rate remains 10^{-4}, and the batch size is eight complexes. Training uses BF16 mixed precision for at most 100 epochs with early-stopping patience 20. No gradient clipping is applied. All reported results use random seeds 0, 42, and 123.

Evaluation Metrics. We follow the evaluation definitions of NeuralMD. Let $e_{f,ij} = \hat{d}_{f,ij} - d_{f,ij}$ denote the error between predicted and target pairwise distances at future frame f. The coordinate error is measured by

$$\text{MAE} = \frac{\sum_{f,i} \|\hat{\mathbf{r}}_{f,i} - \mathbf{r}_{f,i}\|_1}{\sum_f N_f}, \tag{41}$$

where the ℓ_1 norm sums the errors over the three Cartesian coordinates.

For each evaluated sample and frame, Matching is the root-mean-square error over all valid ordered atom pairs,

$$\text{Matching}_f = \sqrt{\frac{1}{|\mathcal{P}_f|} \sum_{(i,j)\in\mathcal{P}_f} e_{f,ij}^2}, \tag{42}$$

where $\mathcal{P}_f$ includes the diagonal and both (i, j) and (j, i). The reported Matching score is averaged over all evaluated sample–frame pairs.

Stability measures the fraction of pairwise distances whose absolute error does not exceed 0.5 Å:

$$\text{Stability}_f = 100 \times \frac{\sum_{(i,j)\in\mathcal{P}_f} \mathbb{1}\left[|e_{f,ij}| \le 0.5 \text{ Å}\right]}{|\mathcal{P}_f|}. \tag{43}$$

The final Stability score is the average over all evaluated sample–frame pairs.

For the single-trajectory comparison in Table 1, the VerletMD, GNN-MD, NeuralMD-ODE, NeuralMD-SDE, and DenoisingLD results are taken from NeuralMD, while BioMD and MinkowskiPE are evaluated in our implementation under the same data split and evaluation protocol. For the multi-trajectory comparison in Table 2, all baselines are retrained under the in80/out20 protocol described above.

B.3 Video Prediction

Data and Preprocessing. We follow the OpenSTL (Tan et al., 2023b) preprocessing and evaluation protocol for the KTH human-action dataset (Schuldt et al., 2004). Persons 01–16 are used for training and persons 17–25 for testing, producing 5,200 training clips and 3,167 test clips. Each clip contains 30 consecutive grayscale frames, with the first 10 used as observations and the subsequent 20 as prediction targets. All frames are resized to 128×128 and pixel intensities are scaled to $[0, 1]$.

During training, the same spatial augmentation is applied synchronously to all frames in a clip. The frames are first bilinearly resized by a factor of $1/0.95$, followed by a random 128×128 crop and horizontal flipping with probability 0.5. Following the OpenSTL preprocessing protocol, 30-frame clips are extracted with start-index strides of 3 frames for jogging and running, and 30 frames for boxing, handclapping, handwaving, and walking.

Model Architecture. Each observed grayscale frame is concatenated with its frame difference, giving a two-channel input. The convolutional encoder consists of three stages, $2 \to 36 \to 72 \to 144$, reducing the spatial resolution from 128×128 to 16×16. Each stage uses a 4×4 convolution with stride 2, followed by GroupNorm and GELU. The resulting latent feature dimension is 144.

The 10 observed latent maps are combined with 20 future latent maps, initialized from the final observed latent map. Each future token is augmented with time-step and token-type embeddings, together with a velocity embedding computed from the difference between the final two observed latent maps. The velocity embedding is modulated by a zero-initialized learned gate. The resulting $30 \times 16 \times 16 = 7{,}680$ event tokens are jointly processed by an eight-layer global Transformer equipped with MinkowskiPE. The Transformer has hidden dimension 144, 8 attention heads, and a 576-dimensional feed-forward layer.

The final 20 latent maps are decoded through progressive bilinear upsampling from 16×16 to 32×32, 64×64, and 128×128. The first two decoding stages use skip connections from the corresponding encoder features of the last observed frame. The convolutional channel dimensions of the three stages are

$$216 \to 72 \to 72, \qquad 108 \to 36 \to 36, \qquad 36 \to 18 \to 18,$$

followed by a final 3×3 convolution from 18 channels to one output channel. The decoder predicts residuals relative to the last observed frame, and all 20 future frames are generated in parallel.

The resulting model contains 2,482,086 parameters in total, reported as 2.5M in Table 3.

Training and Evaluation. The model is trained using the mean squared error between predicted and target frames. We use Adam with a learning rate of 10^{-3}, zero weight decay, and a OneCycle learning-rate schedule. The global batch size is 16, the gradient norm is clipped at 1.0, and training uses BF16 mixed precision. We train for at most 100 epochs with an early-stopping patience of 20 epochs. We follow the OpenSTL evaluation protocol. Within each run, we select the checkpoint with the lowest MSE on the test dataset. We report MSE, MAE, SSIM, PSNR, and LPIPS following the metric definitions used in OpenSTL, with LPIPS computed using the AlexNet backbone. All experiments use random seeds 0, 42, and 123.

C Constraints on Galilean-Invariant Positional Encoding

Our use of Lorentz transformations provides a representation-level geometric bias and does not impose Lorentz symmetry on the underlying dynamics. If one instead seeks a positional encoding that respects the physical spacetime symmetry of non-relativistic dynamics, Galilean symmetry is a natural starting point (Lévy-Leblond, 1971).

This appendix examines a particular construction based on finite-dimensional unitary token-wise transformations. We show that exact invariance under common Galilean transformations forces the resulting pairwise interaction to discard the spatial component of the relative group element. This obstruction depends on the unitary assumption, which is not satisfied by the Lorentz boosts used in MinkowskiPE. It therefore identifies a constraint on this specific route to Galilean-invariant positional encoding.

Galilean Relative Structure. Consider first the rotation-free Galilei group G_0, generated by spatial translations, time translations, and Galilean boosts. An element is written as

$$g = (a, b, v), \tag{44}$$

where $a \in \mathbb{R}^d$ is a spatial translation, $b \in \mathbb{R}$ is a time translation, and $v \in \mathbb{R}^d$ is a boost. The group law is

$$(a_1, b_1, v_1)(a_2, b_2, v_2) = (a_1 + a_2 + v_1 b_2,\ b_1 + b_2,\ v_1 + v_2). \tag{45}$$

We denote the corresponding subgroups by

$$T(a) = (a, 0, 0), \quad S(b) = (0, b, 0), \quad B(v) = (0, 0, v). \tag{46}$$

For spacetime coordinates (x, t) alone, a common Galilean boost acts as

$$x \mapsto x + vt, \quad t \mapsto t, \tag{47}$$

and therefore transforms a relative displacement according to

$$\Delta x_{ij} \mapsto \Delta x_{ij} + v\Delta t_{ij}. \tag{48}$$

Thus, unlike in the translational setting used by MinkowskiPE, Δx_{ij} itself is not invariant under a common Galilean boost.

To obtain a boost-invariant relative spatial quantity, one may augment each token with a velocity u and write

$$z = (x, t, u). \tag{49}$$

Identifying such a state with the group element

$$g(z) = (x, t, u), \tag{50}$$

the relative group element is

$$g(z_i)^{-1} g(z_j) = (r_{ij}, \Delta t_{ij}, \Delta u_{ij}), \tag{51}$$

where

$$r_{ij} = x_j - x_i - u_i(t_j - t_i), \quad \Delta t_{ij} = t_j - t_i, \quad \Delta u_{ij} = u_j - u_i. \tag{52}$$

Under a common Galilean boost,

$$x \mapsto x + vt, \quad u \mapsto u + v, \tag{53}$$

the quantity r_{ij} remains unchanged. Hence r_{ij} is the natural spatial component of the relative Galilean group element.

A Finite-dimensional Obstruction. We now consider a direct rotary-style token-wise encoding. Let $U : G_0 \to U(M)$ be a continuous finite-dimensional unitary encoding, where M is the feature dimension, and define the pairwise operator

$$A(g_i, g_j) = U(g_i)^\dagger U(g_j). \tag{54}$$

We require invariance under a common Galilean transformation,

$$A(hg_i, hg_j) = A(g_i, g_j) \quad \text{for all } h, g_i, g_j \in G_0. \tag{55}$$

This is the direct group-theoretic analogue of the relative-position property underlying rotary positional encodings.

Proposition 1. *Let $U : G_0 \to U(M)$ be continuous, with M finite, and suppose that the pairwise operator in Equation (54) satisfies Equation (55). Then $A(g_i, g_j)$ is independent of the spatial component r_{ij} of the relative group element $g_i^{-1} g_j$. It may depend only on Δt_{ij} and Δu_{ij}.*

Proof. Define

$$F(g) = A(e, g). \tag{56}$$

Choosing $h = g_i^{-1}$ in Equation (55) gives

$$A(g_i, g_j) = F(g_i^{-1} g_j). \tag{57}$$

Moreover, the token-wise factorization in Equation (54) implies

$$A(g_i, g_j) A(g_j, g_k) = A(g_i, g_k). \tag{58}$$

Combining Equation (57) and Equation (58), and writing $a = g_i^{-1} g_j$, $c = g_j^{-1} g_k$, we obtain

$$F(a)F(c) = F(ac). \tag{59}$$

Thus F is a continuous finite-dimensional unitary representation of G_0.

Let P_r, H, and K_r denote the Hermitian generators of spatial translations, time translations, and boosts:

$$F(T(a)) = e^{-ia\cdot P}, \quad F(S(b)) = e^{-ibH}, \quad F(B(v)) = e^{-iv\cdot K}. \tag{60}$$

The Galilei group law in Equation (45) gives

$$B(v)S(b)B(v)^{-1}S(b)^{-1} = T(vb), \tag{61}$$

and therefore the generators satisfy

$$[K_r, H] = iP_r. \tag{62}$$

Spatial translations are central in the rotation-free subgroup G_0, so

$$[P_r, K_s] = 0, \qquad [P_r, H] = 0. \tag{63}$$

Using Equation (62) and Equation (63),

$$\operatorname{tr}(P_r^2) = -i \operatorname{tr}(P_r[K_r, H]) = -i \operatorname{tr}([P_r K_r, H]) = 0. \tag{64}$$

Because P_r is Hermitian, $\operatorname{tr}(P_r^2)$ is the sum of the squared eigenvalues of P_r. Hence $P_r = 0$ for every $r = 1, \cdots, d$. It follows that

$$F(T(r)) = I \quad \text{for all } r \in \mathbb{R}^d. \tag{65}$$

Using Equation (51) and Equation (57)

$$A(g_i, g_j) = F(T(r_{ij})S(\Delta t_{ij})B(\Delta u_{ij})) = F(S(\Delta t_{ij})B(\Delta u_{ij}))\,, \tag{66}$$

which is independent of r_{ij}. □

Implications for Positional Encoding. The proposition does not imply that Galilean positional encoding is impossible in general. Rather, it identifies an obstruction specific to exact, finite-dimensional, unitary, token-wise factorizations of the form in Equation (54). Such an encoding may still represent relative time and relative velocity. For example, one may define

$$U(z) = \bigoplus_{k=1}^{M/2} R\big(\omega_k t + \kappa_k^\top u\big)\,, \tag{67}$$

where $R(\cdot)$ is a planar rotation. Its pairwise interaction depends on

$$\omega_k \Delta t_{ij} + \kappa_k^\top \Delta u_{ij}, \tag{68}$$

but contains no dependence on the Galilean-invariant spatial quantity r_{ij}, in agreement with Proposition 1.

Other approaches, including explicitly pairwise constructions based on Galilean invariants, non-unitary representations, and infinite-dimensional representations, fall outside the assumptions of Proposition 1. Projective representations and central extensions provide further directions that require a separate analysis (Bargmann, 1954; Lévy-Leblond, 1971). These alternatives remain open possibilities for incorporating Galilean structure into positional encoding.