Geometric Clifford Algebra Networks

Incorporating geometry beyond equivariance into neural network transformations.

David Ruhe[1,2], Jayesh Gupta[3], Steven de Keninck[2], Max Welling[1], Johannes Brandstetter[1]

1. Microsoft Research AI4Science 2. University of Amsterdam 3. Microsoft Autonomous Systems Research

Summary

In scientific research, one typicially deals with quantities beyond just scalar values. Vectors have directions and magnitudes expressing, for example, velocities or forces. In machine learning problems, we want to respect this structure. Much work has been done to equip neural networks with *equivariance constraints*, making them behave predictably under a change of the global coordinate system.

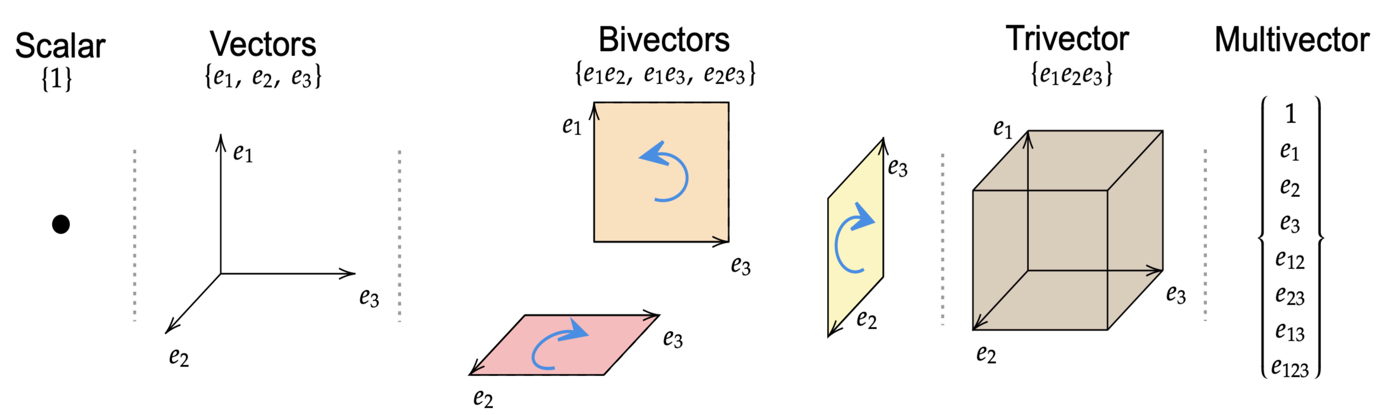
What can we do beyond equivariance to incorporate geometry in neural networks? This work introduces Geometric Clifford Algebra Networks (GCANs), a scalable approach for incorporating geometry-guided transformations into neural networks. We call the resulting parameterizations *geometric templates.* This approach particularly suits tasks where the target function is a geometric transformation of the input data, commonly encountered in dynamical systems science. We showcase the advantages of GCANs in dynamical systems modeling tasks, demonstrating superior generalization in low-data regimes and efficient optimization in high-data situations. We validate these benefits through tests on a rigid body transformation task and two large-scale fluid dynamic problems.

Geometric (Clifford) Algebra

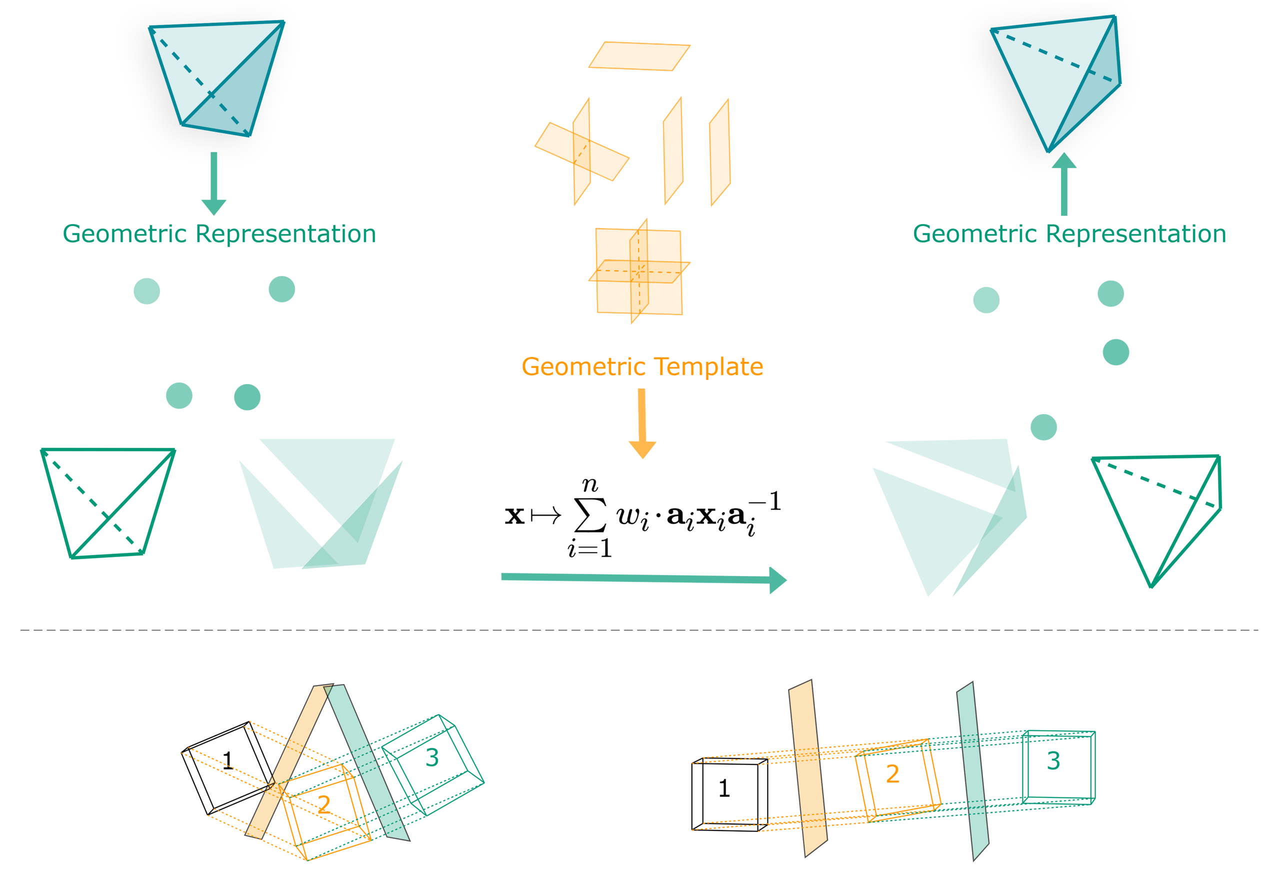
- Elements of the geometric algebra are called *multivectors*. In three dimensions ($\mathbb{G}_3$), one represents a multivector as

$$\mathbf{x} = \underbrace{x_0 1}_{\text{Scalar}} + \underbrace{x_1 e_1 + x_2 e_2 + x_3 e_3}_{\text{Vector}} + \underbrace{x_{12} e_{12} + x_{13} e_{13} + x_{23} e_{23}}_{\text{Bivector}} + \underbrace{x_{123} e_{123}}_{\text{Trivector}}$$

- Multivectors can be multiplied through the *geometric product.*



- We choose how basis vector square: $e_i e_i = e_i^2 \in \{+1, -1, 0\}$
- This leads to a signature p, q, r, where p elements square to 1, q elements to -1, and r elements to 0.
- Different signature lead to different geometries. One can pick a signature that is best suited for the geometry at hand.
- All elements and transformations of these geometries can be encoded in multivectors. E.g., vectors are simultaneously planes, arrows, and reflections.



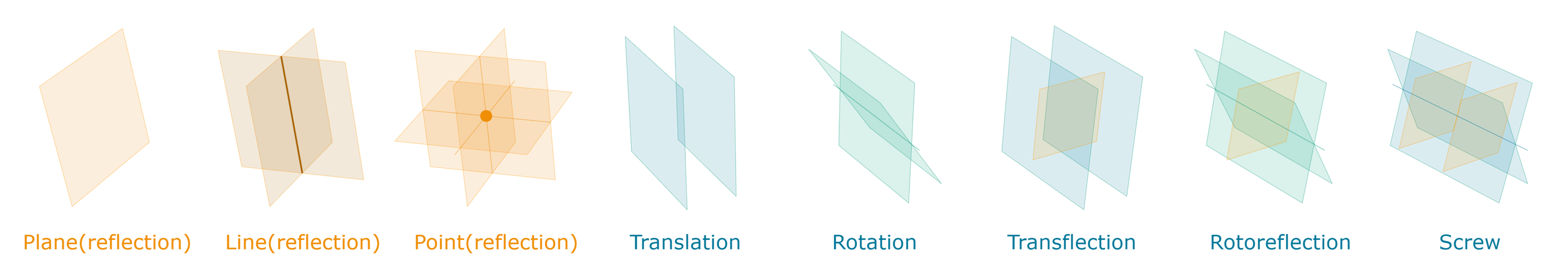
Methods

- We facilitate geometric transformations by picking a *transformation group* G whose actions we apply through a *group action layer*:

$$x \mapsto T_{g,w}(x) := \sum_{i=1}^{c} w_i \cdot \alpha(g_i, x_i)$$

where $w_i \in \mathbb{R}$ and $\alpha : G \times X \to X$ are *learned group actions.*

- Using geometric algebra, we put $G := \mathrm{Pin}(p, q, r)$.



- As such, the group action layer is parameterized as

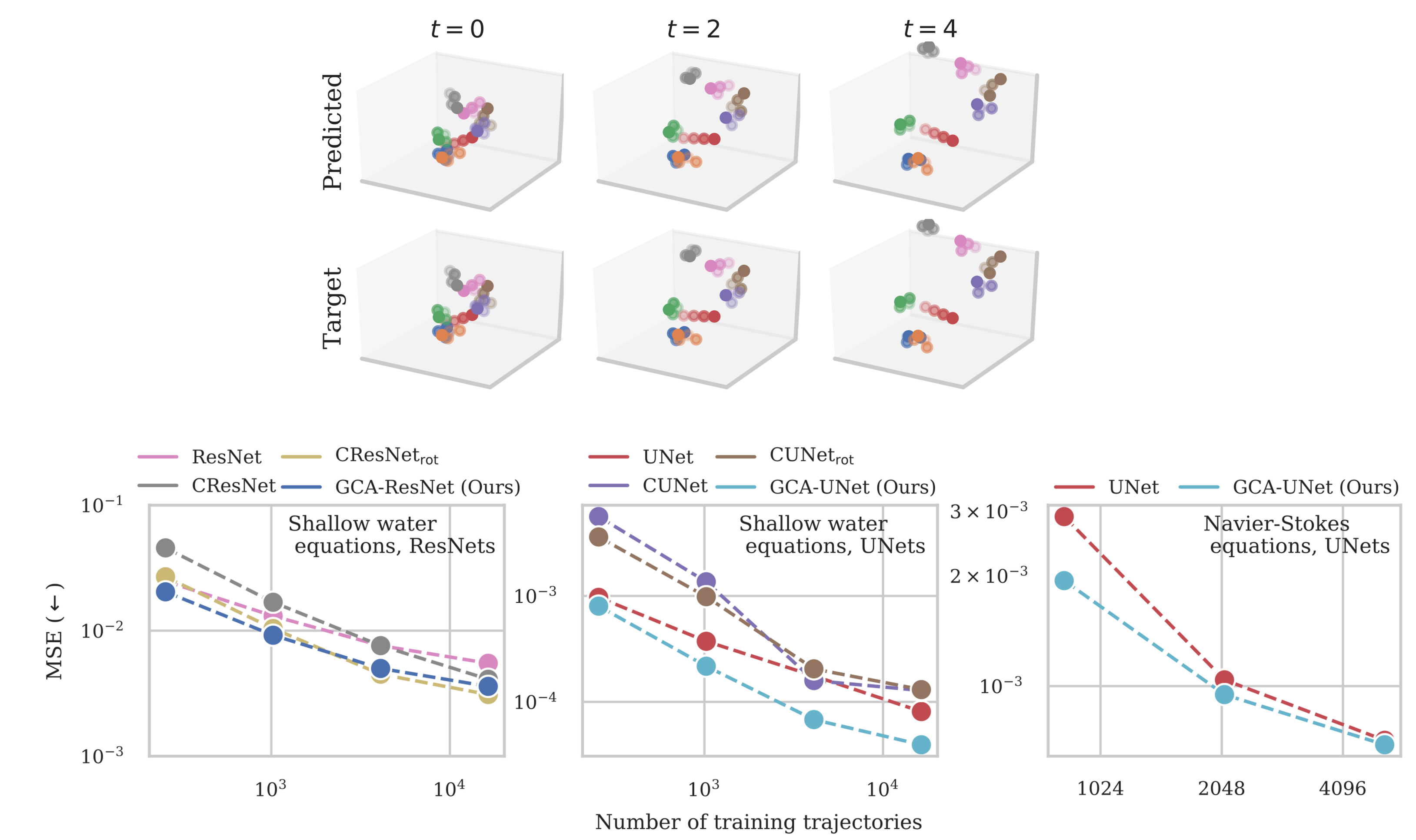
$$T_{g,w} = \sum_{i=1}^{c} w_i \cdot \mathbf{a}_i \mathbf{x}_i \mathbf{a}_i^{-1}$$

Where $X := \mathbb{G}_{p,q,r}$. Here, $\mathbf{a}_i \in \mathbb{G}_{p,q,r}$ parameterizes a learned group action.

URLs

Paper **Code**

Experiments



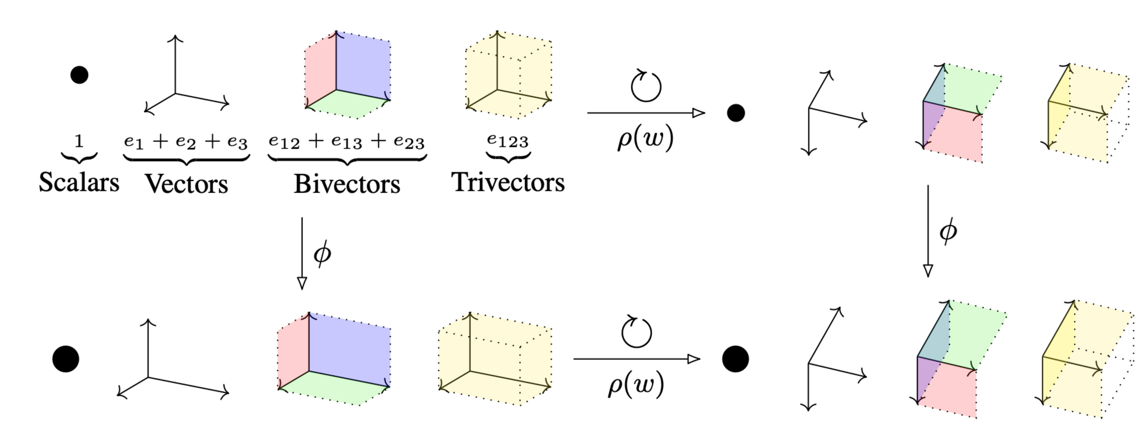
- Exploit the advantage of local transformations on a synthetic Tetris trajectory experiment. We report better results than uninformed and equivariant models.
- We show scalability and efficacy on large-scale PDE emulation experiments.

Key Takeaways

- Geometrically guided transformations are a scalable alternative approach for geometric deep learning.
- Geometric algebra provides an outstanding framework to encode and transform geometric data.
- Scientific dynamical systems are governed by geometric transformations, which is where our method excels.

Still Prefer Equivariance?

Paper



Code