GAUGE-EQUIVARIANT AGGREGATION FOR SPHERICAL ATTENTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Spherical learning tasks often have scalar inputs and outputs, but their spatial structure is naturally described by geometric quantities such as vector and tensor fields. In this work, we investigate if a spherical attention mechanism can benefit from using these intrinsic quantities. To formulate a spherical attention that uses geometric features consistently, we introduce a general gauge-theoretic formulation of normalized aggregation on Riemannian manifolds of arbitrary dimension. Our aggregation operator covers normalized convolution and attention, and we prove that it is equivariant under structure-preserving isometries when explicit assumptions are met. A spherical quadrature realization using exact spherical geometry and Levi-Civita transport is derived. The resulting attention mechanism is exactly gauge-equivariant and can process scalar, vector, and tensor hidden fields. We demonstrate that our implementation can evaluate its learned operators across spherical discretizations. On rotating shallow-water dynamics and Stanford 2D-3D-S omnidirectional semantic segmentation, ablations and comparisons with the S^2 Transformer and S^2 SegFormer demonstrate the benefits of geometric intermediate representations and content-dependent attention weights. Our results support geometric hidden fields as useful representations for spherical attention, although computational cost remains a limitation.

1 INTRODUCTION

Processing signals on the sphere is important in applications such as climate modeling (Bonev et al., 2023), omnidirectional image processing (Carlsson et al., 2024; Benny & Wolf, 2025), and astrophysics (Perraudin et al., 2019). Bonev et al. (2025) demonstrated the benefits of incorporating spherical geometry and quadrature into attention. In this work, we ask if spherical attention can benefit further by incorporating geometry into the hidden representations themselves. Even when a spherical signal is scalar, its spatial structure is naturally described by geometric quantities such as tangent vectors and tensors. For example, the Riemannian gradient and covariant Hessian describe the first- and second-order spatial variation of a scalar function on the sphere (Lee, 2018).

Geometric features, like those mentioned above, are intrinsic quantities on the sphere. Their existence is independent of coordinates, although their components are expressed relative to a choice of local frame (Weiler et al., 2023). Unlike Euclidean space, the sphere admits no globally smooth tangent frame, so these choices must be made locally (Cohen et al., 2019). A spherical attention mechanism that uses intrinsic geometric quantities should therefore produce output components that transform consistently when the same intrinsic input is expressed in different frames. This property is called *gauge equivariance* (Cohen et al., 2019; Weiler et al., 2023).

To formalize this requirement, we present a gauge-theoretic framework for normalized aggregation of geometric feature fields. The framework applies beyond the sphere, so we present it in greater generality in Section 3. Our formulation covers both normalized convolution and attention. We show that, under suitable assumptions on transport and spatial weight sharing, gauge-equivariant normalized aggregation is also equivariant under structure-preserving isometries. On the sphere, this framework gives rise to the $\mathrm{SO}(S^2)$ Transformer components introduced in Section 4, which we use in both forecasting and segmentation architectures in Section 5.

Building on existing gauge-equivariant convolution and attention frameworks, this paper makes three contributions. First, we formulate normalized aggregation on Riemannian manifolds of arbi-

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.

trary dimension and establish explicit conditions for equivariance under structure-preserving isometries. Second, we derive a spherical quadrature realization using exact spherical geometry and Levi-Civita transport, yielding gauge-equivariant attention for scalar, vector, and tensor fields. Third, we assess the value of geometric hidden representations for spherical tasks with scalar inputs and outputs through controlled ablations and comparisons on forecasting and semantic segmentation.

2 Related Works

Gauge-equivariant neural networks. Gauge-equivariant networks process geometric features in a way that respects their transformation laws under a change of local frame. Cohen et al. (2019) formulate gauge-equivariant convolutions on manifolds and introduce the Icosahedral CNN as an efficient realization for spherical signals. Weiler et al. (2023) expand on this by providing a comprehensive account of the general theory of coordinate-independent convolutional networks on manifolds equipped with G-structures. Our formulation of normalized aggregation builds on the gauge-theoretic framework developed in these works.

Gauge-equivariant attention. The Gauge Equivariant Transformer (GET) (He et al., 2021) formulates local attention on surfaces as a normalized integral. Its mesh implementation has exact gauge equivariance at discrete rotation angles and bounded error at other angles. The Equivariant Mesh Attention Networks (EMAN) framework (Basu et al., 2022) uses anisotropic gauge-equivariant mesh convolutions with invariant attention weights and is gauge equivariant under arbitrary transformations in $\mathrm{SO}(2)$. Both works restrict their formulations to surfaces embedded in $\mathbb{R}^3$. They do not consider sample density via quadrature, and their mesh implementations use discrete approximations to surface geometry and parallel transport. Our formulation of normalized aggregation applies to Riemannian manifolds of arbitrary dimension. Our spherical discretization uses exact spherical geometry, exact Levi-Civita transport along geodesics, and quadrature weights which account for the integration measure and sampling density across spherical discretizations.

Spherical attention. Several works adapt attention to the geometry and discretization of spherical signals. HEAL-SWIN (Carlsson et al., 2024) combines shifted-window attention with the hierarchical, equal-area HEALPix discretization (Górski et al., 2005). SphereUFormer (Benny & Wolf, 2025) introduces spherical local self-attention within a multiscale architecture for panoramic depth estimation and semantic segmentation. The work of Bonev et al. (2025) is closest to our continuous formulation; it introduces quadrature-based spherical attention and uses it in both Transformer and SegFormer architectures. Our spherical attention mechanism likewise approximates a continuous operator by quadrature, although ours processes scalar, vector, and tensor fields gauge-equivariantly.

3 Equivariant Normalized Aggregation

Gauges and G-structures. Let (M, η) be an oriented Riemannian manifold of dimension d. A *gauge* on an open subset $U \subset M$ is a smoothly varying choice of vector-space isomorphisms $w_p : \mathbb{R}^d \to T_pM$. Equivalently, a gauge assigns a basis $\{w_p(e_i)\}_{i=1}^d$ to each tangent space, where $\{e_i\}_{i=1}^d$ is the standard basis of $\mathbb{R}^d$. Gauges provide local coordinate representations of intrinsic geometric quantities such as tangent vectors, differential forms, and metrics (Schuller, 2016).

A *gauge transformation* is a smooth map $g : U \to GL(d)$ that changes a gauge w to $\widetilde{w}$ according to $\widetilde{w}_p = w_p \circ g_p$. It follows that the coordinates of a tangent vector $v \in T_pM$ transform by g_p^{-1}, namely, $\widetilde{w}_p^{-1}(v) = g_p^{-1} w_p^{-1}(v)$.

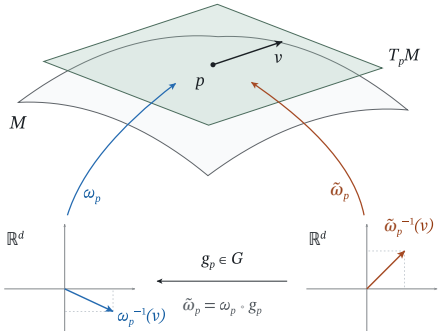

Figure 1: Two gauges $w_p, \widetilde{w}_p$ at $p \in M$. The intrinsic vector $v \in T_pM$ has two different coordinate vectors $w_p^{-1}(v)$ and $\tilde{w}_p^{-1}(v)$, related by g_p^{-1}.

A G-*structure* on M, denoted by GM, specifies a family of gauges which are related by gauge-transformations that take values in a subgroup $G \leq GL(d)$. Such families arise naturally from geometric structures on M. For example, the orientation of M determines a $GL_+(d)$-structure consisting of gauges that define positively oriented frames of each tangent space. Any two such gauges are related by a change-of-basis matrix with positive determinant. Similarly, the Riemannian metric η determines an $O(d)$-structure consisting of gauges that define orthonormal frames, whose changes of basis are orthogonal matrices. The orientation and metric together determine an $\mathrm{SO}(d)$-structure consisting of positively oriented orthonormal frames.

The choice of G controls which geometry to preserve and the equivariance constraint imposed on a GM-aggregation transform. A larger structure group permits a broader class of admissible frame changes and therefore imposes a stronger gauge-equivariance requirement.

Geometric feature fields. Fix a G-structure GM on M. A *field type* is a smooth group representation $\rho\colon G \to \mathrm{GL}(c, \mathbb{R})$, satisfying $\rho(gh) = \rho(g)\rho(h)$ for all $g, h \in G$. A c-channel *geometric feature field* of type ρ is a map f which assigns to each point $p \in M$ an element f_p of a c-dimensional feature space F_p attached to the point p, called the *fiber* over p. Importantly, the feature space F_p cannot be identified with $\mathbb{R}^c$ until a gauge from the considered G-structure is chosen. After fixing a gauge w, the feature field can be locally expressed in coordinates as $f^w\colon U \to \mathbb{R}^c$. Under a gauge transformation $g\colon U \to G$, with $\widetilde{w}_p = w_p \circ g_p$, this coordinatization transforms according to $f_p^{\widetilde{w}} = \rho(g_p)^{-1} f_p^w$. Thus, the field type specifies how feature coordinates change under a change of frame. For example, scalar fields have $\rho(g) = 1$ and are unchanged by gauge transformations, while tangent vector fields have $c = d$ and $\rho(g) = g$. We use Γ_ρ to denote the set of all smooth feature fields of type ρ on M.

For a geometric feature field $f \in \Gamma_\rho$ and distinct points $p, q \in M$, the feature vectors f_p and f_q cannot, in general, be directly compared or combined since they belong to the distinct fibers F_p and F_q, respectively. Parallel transport solves this problem: let w and $\widetilde{w}$ be gauges at p and q, respectively, and $\mathcal{P}_{p\leftarrow q} : T_qM \to T_pM$ a parallel transport map that is compatible with the G-structure. Define

$$g_{p\leftarrow q}^{w\widetilde{w}} := w_p^{-1} \circ \mathcal{P}_{p\leftarrow q} \circ \widetilde{w}_q \in G. \tag{1}$$

The membership of $g_{p\leftarrow q}^{w\widetilde{w}}$ in G is precisely what it means for $\mathcal{P}_{p\leftarrow q}$ to be compatible with the G-structure. The geometric feature vector f_q transported to the fiber over p is given, in the gauge w, by $f_{p\leftarrow q}^w := \rho(g_{p\leftarrow q}^{w\widetilde{w}}) f_q^{\widetilde{w}}$.

GM-Aggregation. The motivation for GM-aggregation is to define a continuous aggregation operator that maps between geometric fields in a way that ensures a change of gauge does not change the represented computation. Doing so defines convolutional and attention-like mechanisms which can retain directional information. As the name suggests, the output field at a point p is obtained by aggregating the input field around p and transforming it to the output field type.

A sufficiently small neighborhood of p can be parameterized by a set of tangent vectors in T_pM around the origin via the Riemannian exponential map $\exp_p$. When aggregating an input field $f \in \Gamma_{\rho_{\mathrm{in}}}$ around a point p, we can therefore probe the neighboring feature vectors by mapping a tangent vector $v \in T_pM$ to a neighboring point $\exp_p(v)$, sampling the field at that point, and transporting the sampled feature vector back to the fiber over p along the geodesic $\exp_p(tv)$. This process defines an $F_{\mathrm{in},p}$-valued field $\hat{f}_p$ on T_pM. In coordinates, $\hat{f}_p^w : \mathbb{R}^d \to \mathbb{R}^{c_{\mathrm{in}}}$ is given by $\hat{f}_p^w(v) = f_{p\leftarrow \exp_p(w_p v)}^w$, where w is a gauge about p.

For a compact G, a GM-*aggregation* is a map $A : \Gamma_{\rho_{\mathrm{in}}} \to \Gamma_{\rho_{\mathrm{out}}}$ that is given pointwise by

$$(Af)_p^w := \frac{\displaystyle\int_{\mathcal{N}} \alpha\big(f_p^w, \hat{f}_p^w(v)\big) K(v) \hat{f}_p^w(v)\, dv}{\displaystyle\int_{\mathcal{N}} \alpha\big(f_p^w, \hat{f}_p^w(v)\big)\, dv} \tag{2}$$

for any gauge w about p from the chosen G-structure. Here, $\mathcal{N} \subset \mathbb{R}^d$ is a neighborhood of the origin that controls the region around p over which aggregation takes place, $\alpha : \mathbb{R}^{c_{\mathrm{in}}} \times \mathbb{R}^{c_{\mathrm{in}}} \to \mathbb{R}$

is a feature vector affinity function, and $K : \mathbb{R}^d \to \mathbb{R}^{c_{\text{out}} \times c_{\text{in}}}$ is a value transformation kernel. To be well defined, both integrals must be finite, the denominator must be nonzero at each $p \in M$, and the resulting field must be smooth. The affinity function and value transformation kernel are independent of the spatial position p and the gauge w. Moreover, the neighborhood $\mathcal{N}$ is invariant under the action of G on $\mathbb{R}^d$ in addition to being independent of p. To ensure gauge consistency, we require α and K to satisfy

$$\alpha(f_1, f_2) = \alpha\big(\rho_{\text{in}}(g)f_1, \rho_{\text{in}}(g)f_2\big), \qquad K(gv) = \rho_{\text{out}}(g)K(v)\rho_{\text{in}}(g)^{-1}, \tag{3}$$

for all $g \in G$, $v \in \mathbb{R}^d$, and $f_1, f_2 \in \mathbb{R}^{c_{\text{in}}}$. That is, the affinity is gauge invariant and the value transformation kernel is gauge equivariant (Cohen et al., 2019; Weiler & Cesa, 2019).

As special cases, define *GM-convolution* to be the case with trivial affinity $\alpha \equiv 1$ and *GM-attention* to be the case with affinity (Vaswani et al., 2017)

$$\alpha(f_1, f_2) = \exp\left(\frac{\langle W_Q f_1, W_K f_2\rangle_{\rho_{\text{attn}}}}{\sqrt{c_{\text{attn}}}}\right), \tag{4}$$

where $\rho_{\text{attn}} : G \to \text{GL}(c_{\text{attn}}, \mathbb{R})$ is a latent attention representation, $\langle\cdot,\cdot\rangle_{\rho_{\text{attn}}}$ is a G-invariant inner product on $\mathbb{R}^{c_{\text{attn}}}$, and $W_Q, W_K \in \mathbb{R}^{c_{\text{attn}} \times c_{\text{in}}}$ are gauge-equivariant linear maps satisfying

$$\rho_{\text{attn}}(g)W_Q = W_Q\rho_{\text{in}}(g), \quad \rho_{\text{attn}}(g)W_K = W_K\rho_{\text{in}}(g), \tag{5}$$

for all $g \in G$. The GM-attention construction can be extended to a GM-multi-head self-attention variant (Appendix B.2).

Equation 2 can recover continuum Euclidean attention (Calvello et al., 2025), where the input and output fields are scalar; equivariant Euclidean attention (Fuchs et al., 2020; Chatzipantazis et al., 2023), where $M = \mathbb{R}^3$ and $G = \text{SO}(3)$; mesh formulations (He et al., 2021; Basu et al., 2022), where $M \subset \mathbb{R}^3$ is a surface and $G = \text{SO}(2)$; and continuous spherical attention (Bonev et al., 2025), where the input and output fields are scalars. Some of the models cited here additionally allow the affinity to depend on the tangent displacement in TM. We chose not to explicitly include this generalization in our formulation because our spherical implementation does not require it. Our arguments for gauge equivariance and isometry equivariance (Theorem 1) only require that the affinity function be determined by a single gauge-invariant template. They can therefore be extended directly to aggregation operators with displacement-dependent affinities.

Isometry equivariance. An isometry $\phi : M \to M$ acts on tangent vectors through the differential $d\phi_p : T_pM \to T_{\phi(p)}M$. Because isometries are diffeomorphisms, $d\phi_p$ maps tangent frames to tangent frames. We say that ϕ is a *G-structure-preserving isometry* if $d\phi$ maps frames compatible with the G-structure to frames compatible with the G-structure: for each $p \in M$ and each gauge w from the G-structure about p, there exists a gauge $\widetilde{w}$ from the G-structure about $\phi(p)$ such that $d\phi_p(w_p(e_i)) = \widetilde{w}_{\phi(p)}(e_i)$ for each $i = 1, \ldots, d$. The set of all G-structure-preserving isometries forms a group under composition and is denoted by Isom_{GM}. This group acts on feature fields $f \in \Gamma_\rho$. Expressed with respect to a gauge w about p, the action is $(\phi \triangleright_\rho f)^w_p := \rho(g_\phi^{w\widetilde{w}}(p))f^{\widetilde{w}}_{\phi^{-1}(p)}$, where $\widetilde{w}$ is any gauge from the G-structure about $\phi^{-1}(p)$ and

$$g_\phi^{w\widetilde{w}}(p) := w_p^{-1} \circ d\phi_{\phi^{-1}(p)} \circ \widetilde{w}_{\phi^{-1}(p)} \in G.$$

Theorem 1 (Isometry Equivariance)**.** *Let (M, η) be an oriented Riemannian manifold of dimension d, equipped with a G-structure GM, where $G \leq GL(d)$ is a compact Lie subgroup. Choose a G-compatible connection whose parallel transport is preserved by every $\phi \in \text{Isom}_{GM}$. Then every well-defined GM-aggregation $A : \Gamma_{\rho_{\text{in}}} \to \Gamma_{\rho_{\text{out}}}$ constructed using this transport is equivariant with respect to Isom_{GM}:*

$$A\big(\phi \triangleright_{\rho_{\text{in}}} f\big) = \phi \triangleright_{\rho_{\text{out}}} (Af)$$

for every $f \in \Gamma_{\rho_{\text{in}}}$ and $\phi \in \text{Isom}_{GM}$.

The intuition of the proof is that a G-structure-preserving isometry preserves the exponential map, parallel transport, and Riemannian volume (Lee, 2018). Spatial weight sharing applies the same aggregation templates at p and $\phi(p)$, while gauge sharing ensures that the induced G-valued change of frame is absorbed by the templates' gauge constraints. Thus, the aggregation response is carried equivariantly from p to $\phi(p)$. A detailed proof is provided in Appendix A.2.

When G is $\text{O}(d)$ or $\text{SO}(d)$, the Levi-Civita connection satisfies the hypothesis. Moreover, $\text{Isom}_{\text{O}(M)}$ is the full isometry group and $\text{Isom}_{\text{SO}(M)}$ consists of orientation-preserving isometries (Weiler et al., 2023, Theorems 13.2.6–13.2.7).

4 SPHERICAL REALIZATION

We now propose a quadrature realization of GM-aggregation on the unit sphere $S^2 \subset \mathbb{R}^3$ equipped with an $\mathrm{SO}(2)$-structure, $\mathrm{SO}(S^2)$, which consists of outward-oriented orthonormal tangent frames.

Discretizing the sphere. Let $X = \{p_i\}_{i=1}^n \subset S^2$ and $Q_{\mathrm{node}} = \{q_i\}_{i=1}^n \subset \mathbb{R}_{>0}$ define a spherical quadrature rule such that $\sum_{i=1}^n q_i = 4\pi$ and $\int_{S^2} f \, d\mu \approx \sum_{i=1}^n q_i f(p_i)$ for suitable integrands $f : S^2 \to \mathbb{R}$, where $d\mu$ is the Riemannian surface measure (Atkinson & Han, 2012). From these initial data, we construct all the geometric ingredients needed to discretize an $\mathrm{SO}(S^2)$ aggregation.

Let $W = \{w_i\}_{i=1}^n \subset \mathbb{R}^{3\times 2}$, where w_i is a gauge that defines an outward-oriented orthonormal frame of $T_{p_i}S^2 := \{v \in \mathbb{R}^3 : \langle p_i, v\rangle = 0\}$; that is, the columns of w_i are orthonormal and their cross product is p_i. Gauge equivariance and coordinate independence of $\mathrm{SO}(S^2)$ aggregation ensure that this choice of frame can indeed be arbitrary. We compute the parallel transporters with respect to these frames: $P := \{g_{i\leftarrow j}\}_{i,j=1}^n$, where $g_{i\leftarrow j} := g_{p_i\leftarrow p_j}^{w_i w_j}$ (equation 1).

Since aggregation is computed on tangent spaces, define a set of edge vectors $\mathcal{E} = \{v_{ij}\}_{i,j=1}^n$, where $v_{ij} := w_i^\top \log_{p_i}(p_j) \in \mathbb{R}^2$. Thus, v_{ij} is the coordinate expression, in the gauge w_i, of the tangent vector in $T_{p_i}S^2$ that is mapped to p_j under the Riemannian exponential. The quadrature weights Q_{node} for the spherical measure induce quadrature weights $Q_{\mathrm{edge}} = \{q_{ij}\}_{i,j=1}^n$ for tangent-space integration via

$$q_{ij} := q_j \frac{\|v_{ij}\|}{\sin \|v_{ij}\|},$$

where the ratio is defined by continuity to be 1 when $v_{ij} = 0$. This follows from the change-of-variables formula (Lee, 2018).

Define a spherical discretization to be the tuple $\mathcal{D} = (X, Q_{\mathrm{node}}, \mathcal{E}, Q_{\mathrm{edge}}, W, P)$ [1]. Notice that each component of the discretization can be constructed from a spherical quadrature rule. The computations involve matrix multiplications, evaluations of the spherical logarithm, and applications of Rodrigues' rotation formula (Murray et al., 1994), and can be performed once up front and cached.

Geometric fields. Since $\mathrm{SO}(2)$ is a compact group, we can write an arbitrary continuous real representation $\rho : \mathrm{SO}(2) \to \mathrm{GL}(\mathbb{R}^c)$ as a direct sum of irreducible representations $\rho = m_0\rho_0 \oplus \cdots \oplus m_n\rho_n$ after choosing a suitable basis for $\mathbb{R}^c$ (Weiler & Cesa, 2019). Here m_k is the multiplicity of the irreducible scalar representation $\rho_k(r_\theta) = 1$ when $k = 0$ and frequency k representation $\rho_k(r_\theta) = r_{k\theta}$ when $k > 0$, where r_θ denotes the rotation matrix in $\mathrm{SO}(2)$ for an angle θ. Thus, fields of type ρ_0 are scalar fields, those of type ρ_1 are tangent vector fields, and those of type ρ_2 are spin-2 fields. Vector fields can encode information such as gradients and directed boundary normals, while spin-2 fields can encode information such as shear and unoriented texture axes (Weickert, 1998; Weiler & Cesa, 2019). Our default shallow-water models use a latent geometric field of type $m(\rho_0 \oplus \rho_1 \oplus \rho_2)$.

In Appendix B.1, we provide parameterizations of the pointwise gauge-equivariant query and key projections satisfying equation 5 and the steerable value kernels satisfying equation 3. Pointwise equivariant linear maps cannot mix inequivalent irreducible types, but steerable value kernels can (Weiler & Cesa, 2019) and our models use them to lift scalar inputs to geometric latent fields. Moreover, if either the input or output field has a frequency $k > 0$ component, then the value kernel can be anisotropic.

Discretizing $\mathrm{SO}(S^2)$ Aggregation. Fix a spherical discretization $\mathcal{D}$. For $f \in \Gamma_{\rho_{\mathrm{in}}}$, let $f_i := f_{p_i}^{w_i}$ denote the sampled input feature at p_i and $f_{i\leftarrow j} := \rho_{\mathrm{in}}(g_{i\leftarrow j})f_j$ denote the feature vector at p_j transported back to the fiber over p_i, expressed in the frame w_i. We choose an $\mathrm{SO}(2)$-invariant aggregation neighborhood to be a ball $\mathcal{N} = B_0(R) \subset \mathbb{R}^2$ about the origin for some $0 < R < \pi$, where π is the injectivity radius of the unit sphere (Lee, 2018). Define $\mathcal{N}_i := \{j : \|v_{ij}\| < R\}$ so that $\mathcal{N}_i$ consists of the indices of all nodes that contribute to aggregation at p_i. The discretization of

[1] Pairwise constructions $g_{i\leftarrow j}, v_{ij}$, and q_{ij} are technically not defined for antipodal points. For the sake of our local aggregation this does not cause issues.

an $\mathrm{SO}(S^2)$ aggregation transform $A : \Gamma_{\rho_{\mathrm{in}}} \to \Gamma_{\rho_{\mathrm{out}}}$ with respect to $\mathcal{D}$ is

$$(A_{\mathcal{D}} f)_i := \frac{\sum_{j \in \mathcal{N}_i} q_{ij}\, \alpha\big(f_i, f_{i \leftarrow j}\big)\, K(v_{ij}) f_{i \leftarrow j}}{\sum_{j \in \mathcal{N}_i} q_{ij}\, \alpha\big(f_i, f_{i \leftarrow j}\big)}.$$

As in the general setting, the two special cases are $\mathrm{SO}(S^2)$ convolution, obtained with the trivial affinity $\alpha \equiv 1$, and $\mathrm{SO}(S^2)$ attention, obtained using the attention affinity in equation 4 with the standard dot product. Note that the steerable value kernel in the definition of $\mathrm{SO}(S^2)$ attention is retained in place of a pointwise linear map. This design gives our attention mechanism relative positional awareness.

The discrete aggregation is a quadrature approximation of a continuous operator equivariant under $\mathrm{Isom}_{\mathrm{SO}(S^2)} \cong \mathrm{SO}(3)$, and its learnable components α and K are independent of the discretization. The equivariance constraints on the affinity, value kernel, and aggregation neighborhood ensure exact gauge equivariance of the discrete construction. This property concerns coordinate independence and does not, by itself, imply exact isometry equivariance on a fixed discretization since representing the rotated field and approximating the continuous operator introduce discretization error.

$\mathrm{SO}(S^2)$ **Transformer blocks.** An $\mathrm{SO}(S^2)$ Transformer block is our coordinate-independent variant of the pre-norm transformer block (Vaswani et al., 2017; Xiong et al., 2020), capable of processing geometric feature fields. The attention mechanism is our $\mathrm{SO}(S^2)$ multi-head self-attention. Layer normalization (Ba et al., 2016) is replaced with a gauge-equivariant generalization, the linear layers in the FFN are replaced with pointwise equivariant linear maps, and the nonlinearity is a scalar gate (Weiler et al., 2018). The precise details of each of these components can be found in Appendix B.2.

5 Experiments

5.1 Shallow-Water Equations on the Rotating Sphere

The rotating shallow-water equations describe the dynamics of a thin fluid layer on a rotating sphere (Williamson et al., 1992). The state consists of three scalar fields: geopotential φ, vorticity ζ, and divergence δ. Trajectories are generated with the spectral shallow-water solver of `torch-harmonics` (Bonev et al., 2023), with Earth's radius, gravity, and rotation rate on a 256×512 solver grid and lead time of $\Delta t = 1$ hour. The channels φ, ζ, δ are standardized using channel-wise means and standard deviation statistics shared across all models. The same spectral solutions are provided to each model and are subsequently synthesized on their nodes. To encode the fixed planetary rotation axis, we append the Coriolis field $c(p) = \langle p, \hat{z} \rangle$ as an additional input channel to each model, where $p \in S^2$ and $\hat{z}$ is the unit rotation axis vector. This channel is static and is omitted from the predicted state. All models predict the residual $f_{t+\Delta t} - f_t$.

Our models. We consider two $\mathrm{SO}(S^2)$ Transformer architectures with different tradeoffs between computational cost and performance, serving distinct experimental purposes. The *depth-1* (D1) model takes its input on the level-5 icosphere, applies an $\mathrm{SO}(S^2)$ convolution that restricts to the level-4 icosphere and projects into the hidden field. Four $\mathrm{SO}(S^2)$ Transformer blocks process the latent signal. Then the decoder upsamples back to level 5 by parallel-transported interpolation and applies an $\mathrm{SO}(S^2)$ convolution to produce the output. Spatial downsampling makes D1 our lower-cost configuration, but reducing the number of spatial samples by a factor of four poses a challenge for retaining high-frequency information. We therefore use D1 for ablations that assess the benefit of geometric hidden features when processing the signal at a coarser spatial resolution.

The *single-level* (SL) model instead consists of an $\mathrm{SO}(S^2)$ convolution encoder, four $\mathrm{SO}(S^2)$ Transformer blocks, and an $\mathrm{SO}(S^2)$ convolution decoder, all operating on the same discretization. Each operator is defined for any point set on S^2 equipped with suitable quadrature weights. We use this model to test the transfer capability of the $\mathrm{SO}(S^2)$ Transformer across spherical discretizations.

Reference models. The primary reference model is the S^2 Transformer (Bonev et al., 2025). This comparison evaluates the complete spherical attention architectures, while the D1 ablations more

Table 1: The first three rows are the same SL checkpoints evaluated on different grids. The remaining rows report D1 and the reference models on their native grids. *points*: the grid's size. *native*: quadrature approximation on the grid being evaluated (Eq. 20). $\ell \leq 85$: the normalized, channel-averaged squared L^2 norm of a weighted least-squares fit (Eq. 21). Persistence scores 1551 and 986 in $10^3 \times \mathcal{L}_2^2$ units at the two horizons. The S^2 Transformer's five-step standard deviations are 3.0×10^6 and 1.7×10^4 for the native and $\ell \leq 85$ metrics, respectively.

		One step (1 h), $10^3 \times \mathcal{L}_2^2$		5 steps (5 h), $10^3 \times \mathcal{L}_2^2$	
Grid	points	native	$\ell \leq 85$	native	$\ell \leq 85$
$\mathrm{SO}(S^2)$ *Transformer SL (ours), 491,404 parameters*					
L5 icosphere (train)	10,242	$\mathbf{0.82 \pm 0.01}$	$\mathbf{0.76 \pm 0.01}$	$\mathbf{9.05 \pm 0.16}$	$\mathbf{9.03 \pm 0.16}$
L6 icosphere	40,962	2.80 ± 0.20	2.78 ± 0.20	15.63 ± 3.07	15.62 ± 3.07
Legendre–Gauss 128×256	32,768	3.91 ± 0.21	3.89 ± 0.21	16.92 ± 3.30	16.91 ± 3.30
$\mathrm{SO}(S^2)$ *Transformer D1 (ours), 491,404 parameters*					
L5 icosphere (train)	10,242	4.72 ± 0.07	4.14 ± 0.06	63.07 ± 9.00	62.29 ± 9.05
SFNO, 503,866 parameters					
equiangular 256×512 (train)	131,072	0.87 ± 0.04	0.87 ± 0.04	76.42 ± 4.73	76.39 ± 4.72
S^2 *Transformer, single level, 535,424 parameters*					
Legendre–Gauss 128×256 (train)	32,768	5.80 ± 0.26	5.03 ± 0.24	4.7×10^6	2.5×10^4

directly assess the contribution of the hidden field type. We configure the S^2 Transformer to mirror our single-level architecture, using DISCO convolutions (Ocampo et al., 2023) for the encoder and decoder and four S^2 Transformer blocks to process the scalar latent field. This configuration differs from their published SWE benchmark. Appendix C.2 reports an additional comparison using the released S^2 Transformer architecture on a 128×256 solver grid with a 30-minute lead time. In this setting, D1 achieves a one-step native-grid squared error of 0.402, compared with 3.011 for the S^2 Transformer and 0.236 for SFNO, in $10^3 \times \mathcal{L}_2^2$ units. In addition, we include a parameter-matched Spherical Fourier Neural Operator (SFNO) (Bonev et al., 2023) as a strong baseline.

Training and evaluation. All models are trained with Adam (Kingma & Ba, 2015), using a batch size of 4 for 100 epochs. On a Tesla V100, the average wall-clock time per epoch was 0.9 min for D1, 1.7 min for SL, 0.9 min for S^2 Transformer, and 0.1 min for SFNO. Each epoch uses a fresh set of 256 initial-condition/target pairs from fixed seed schedules shared across all models. The reference and our D1 models use the `ReduceLROnPlateau` scheduler while our SL models use a cosine scheduler (Loshchilov & Hutter, 2017). Learning rates are chosen per model by grid search and are held fixed across 3 seeds. Training minimizes the quadrature approximation of the channel-averaged squared $\mathcal{L}_2$ loss on each model's training grid (Eq. 20). Models are evaluated on a held-out set of 64 test trajectories and results are reported as (seed mean) $\pm$ seed sd. At evaluation, native-grid errors use the quadrature weights of the evaluation grid. For comparisons across grids, we also report the normalized, channel-averaged squared L^2 norm of a weighted least-squares fit of the error to a shared spherical harmonic basis of degree at most $\ell_{\max} = 85$ (Eq. 21). This fitted metric remains subject to grid-dependent aliasing.

Prediction accuracy and zero-shot transfer study. We trained the single-level model on the level-5 icosphere and then evaluated it zero-shot on the level-6 icosphere and a 128×256 Legendre–Gauss grid. For each grid, the solver's spectral solutions of input/target trajectories were synthesized at its nodes from the spectral coefficients, and the discretization together with the sampled input field was then passed directly to the SL model. The model weights are frozen, the encoder/decoder kernel supports and attention neighborhoods are determined by fixed geodesic radii, and the same normalization statistics from training are used. The results are presented in Table 1.

When each model is evaluated on its native grid, SL achieves the lowest errors under both the native-grid and common-band metrics. Although zero-shot transfer degrades performance relative to evaluation on the training grid, the predictions remain competitive on the unseen grids. The single-level $\mathrm{SO}(S^2)$ Transformer evaluated zero-shot on the 128×256 Legendre–Gauss grid achieves a 33% lower one-step $\mathcal{L}_2^2$ error than the S^2 Transformer, despite the latter being trained on that

Table 2: Depth-1 ablation results. Persistence scores 1551 in $10^3 \times \mathcal{L}_2^2$ units.

			$10^3 \times \mathcal{L}_2^2$	$10^2 \times \mathcal{L}_2$ per channel		
Mixer	Hidden type	Params		φ	ζ	δ
$\mathrm{SO}(S^2)$ MHSA	$\rho_0^{16} \oplus \rho_1^{16} \oplus \rho_2^{16}$	491,404	**4.72 ± 0.07**	**4.81 ± 0.06**	**5.15 ± 0.04**	**9.59 ± 0.10**
$\mathrm{SO}(S^2)$ conv	$\rho_0^{16} \oplus \rho_1^{16} \oplus \rho_2^{16}$	475,276	6.59 ± 0.22	5.26 ± 0.20	5.99 ± 0.10	11.58 ± 0.23
$\mathrm{SO}(S^2)$ MHSA	$\rho_0^{26} \oplus \rho_1^{26}$	493,580	6.55 ± 0.18	5.32 ± 0.14	5.47 ± 0.08	11.76 ± 0.15
$\mathrm{SO}(S^2)$ MHSA	ρ_0^{72}	510,780	57.62 ± 5.32	18.02 ± 0.68	12.44 ± 0.11	35.30 ± 1.97

grid. The degradation in zero-shot performance is concentrated in the highest degree band, 48–85 (Table 4). It could be that the denser transfer grids resolve high-frequency content that was poorly resolved on the training grid. SL outperforms D1 by a factor of 5.75 at a cost of having a 1.8 times slower training epoch.

Ablation study. To test the effect of the hidden field type on the $\mathrm{SO}(S^2)$ Transformer's performance, we train three D1 models differing only in their hidden field type. In particular, we consider our model's default $\rho_0^{16} \oplus \rho_1^{16} \oplus \rho_2^{16}$, the type without frequency-2 channels $\rho_0^{26} \oplus \rho_1^{26}$, and the scalar-only type ρ_0^{72}. The multiplicities of the ablation models are chosen to match the parameter count of the default model to within 4%. Additionally, we test the contribution of the normalized, field-dependent attention weights by training a model that differs from D1 only in the spatial mixer of the transformer blocks, replacing $\mathrm{SO}(S^2)$ multi-head self-attention with $\mathrm{SO}(S^2)$ convolution. The errors of these models' predictions after one step are presented in Table 2. In the D1 architecture, removing spin-2 channels increases one-step error by 39% and restricting the hidden field to scalars increases it by a factor of 12.2. Replacing MHSA with convolution increases the error by 40%.

5.2 Stanford 2D-3D-S Semantic Segmentation

This experiment tests if geometric hidden representations are also useful for semantic interpretation of spherical images. The Stanford 2D-3D-S dataset (Armeni et al., 2017) includes RGB panoramas on an equirectangular grid with per-pixel semantic labels. There are 13 semantic classes and one "unknown" label. We use the preprocessing pipeline of Bonev et al. (2025) to produce a set of 1,413 panoramas on a 128×256 equiangular grid, which we randomly split at the panorama level into training (1,343), validation (35), and test (35) sets. Input channels are normalized using per-channel means and standard deviations computed on the training set with the grid's spherical quadrature weights. No data augmentation is used.

Following Bonev et al. (2025), we evaluate the components from Section 4 within the SegFormer architecture (Xie et al., 2021). We compare the S^2 SegFormer with our $\mathrm{SO}(S^2)$ SegFormer, which replaces the reference model's components with their $\mathrm{SO}(S^2)$ counterparts. Each stage of our model is conditioned on gravity to exploit the preferred vertical direction of indoor scenes. A detailed description of the $\mathrm{SO}(S^2)$ SegFormer is given in Appendix D.2. We also evaluate a variant with scalar hidden fields and a comparable parameter count. A planar $\mathbb{R}^2$ SegFormer and an S^2 UNet, using the configurations of Bonev et al. (2025, Table 1), are reported as additional baselines.

All models, including our own, follow the training protocol used in Bonev et al. (2025): AdamW (Loshchilov & Hutter, 2019) with weight decay 0.1, a cosine schedule over 200 epochs, and batches of 8 panoramas. The average wall-clock time for one epoch was 3.9 min for $\mathrm{SO}(S^2)$ SegFormer, 3.0 for the scalar ablation, 0.5 min for S^2 SegFormer, and 0.2 min for both $\mathbb{R}^2$ SegFormer and S^2 UNet. The loss is the quadrature-weighted cross-entropy without class weights. The base learning rate and drop-path probabilities of our models are selected once on the validation set and the baselines use their released defaults. Each panorama is scored on its own. The quadrature-weighted micro-averaged intersection over union (IoU) across the 14 classes and the mean IoU over the classes present in either the target or prediction of each panorama are reported. Scores are averaged over the 35 panoramas in the test set and over three training seeds. See Appendix D.1 for more details on the loss and metrics.

Table 3: Stanford 2D-3D-S semantic segmentation results on the test split of 35 panoramas. Metrics are reported as mean ± standard deviation across three seeds.

Model	Params	CE ↓	IoU ↑	mIoU ↑
$\mathrm{SO}(S^2)$ SegFormer (ours)	593,518	$\mathbf{0.552 \pm 0.023}$	$\mathbf{0.701 \pm 0.012}$	$\mathbf{0.427 \pm 0.011}$
$\mathrm{SO}(S^2)$ SegFormer, scalar	599,158	0.666 ± 0.014	0.645 ± 0.004	0.372 ± 0.005
S^2 SegFormer	609,246	0.637 ± 0.014	0.676 ± 0.003	0.411 ± 0.004
$\mathbb{R}^2$ SegFormer	692,398	0.747 ± 0.024	0.655 ± 0.005	0.384 ± 0.005
S^2 UNet	741,374	0.705 ± 0.024	0.634 ± 0.007	0.353 ± 0.003

The results in Table 3 show that, with the fewest parameters, the $\mathrm{SO}(S^2)$ SegFormer outperforms every other model on each metric. The scalar variant has approximately 21% higher mean CE and 8% lower IoU than the model with geometric hidden fields. Moreover, the $\mathrm{SO}(S^2)$ SegFormer achieves an IoU 2.5 percentage points higher than the S^2 SegFormer, a larger improvement than the gain of 2.1 percentage points from the planar $\mathbb{R}^2$ SegFormer to the S^2 SegFormer. The $\mathrm{SO}(S^2)$ SegFormer's advantage is most pronounced near boundaries and thin, vertically extended structures that the other models classify as part of the surrounding wall (Table 6). In particular, the $\mathrm{SO}(S^2)$ architecture achieves higher recall on the classes "columns," "boards," and "windows" than both the S^2 SegFormer and the scalar ablation. Performance on large regions such as "floor" and "ceiling" is much closer across these models. This aligns with our motivation for using ρ_1 and ρ_2 features, which can represent image gradients and the traceless part of local structure tensors, respectively, quantities used in classical edge detection and orientation estimation (Canny, 1986; Knutsson, 1989).

6 Discussion

The results from our two experiments support the utility of intermediate geometric features for spherical attention architectures. In both experiments, the $\mathrm{SO}(S^2)$ architectures with ρ_0, ρ_1, and ρ_2 hidden representations outperformed the scalar comparison models. Geometric quantities are natural intermediates in mathematical approaches to these problems, and our results suggest that neural networks also benefit from being able to represent them.

The primary limitation of the $\mathrm{SO}(S^2)$ architectures is the computational cost of geometry-dependent per-edge computations. If the radial kernel basis and representation frequencies are fixed, then direct evaluation of a steerable value kernel requires $\mathcal{O}(NkC^2)$ operations, where N is the number of nodes, k is the average number of neighbors per node, and C is the feature dimension. The parameter count scales as $\mathcal{O}(C^2)$, and materializing the per-edge coefficients can require $\mathcal{O}(NkC^2)$ memory during training. Moreover, the cached geometry requires an additional $\mathcal{O}(Nk)$ memory. Since the attention neighborhoods are defined by fixed geodesic radii, uniform refinement of the discretization gives $k = \Theta(N)$, and thus the number of interactions grows quadratically.

To obtain a practical implementation, we use algebraically exact factorizations and custom CUDA kernels for attention scoring and normalization, steerable value evaluation, and attention-weighted summation. The factorizations include computing query and key projections once per node, exploiting the gauge-equivariance constraint (equation 5), and regrouping the steerable value parameterization (Appendix B.1) so that channel mixing occurs at nodes rather than edges. These optimizations avoid materializing the full per-edge kernel coefficient tensors and reduce the leading computational cost to $\mathcal{O}(NC^2 + NkC)$. Fusing the three kernel stages into a streaming computation is an opportunity for further optimization.

Beyond optimizing the implementation further, we see two directions for future work. First, hybrid architectures could lift to hidden geometric representations in only a subset of deep layers. Such architectures could strike a balance between the efficiency of scalar models and the value of hidden geometric features. Second, the steerable value kernel in $\mathrm{SO}(S^2)$ attention could be replaced by a shared, pointwise equivariant linear map. Doing so would remove the cost of the full steerable value transformation while retaining the ability to process geometric fields. This would eliminate the learned displacement dependence of the value transformation, and thus an alternative positional encoding would likely have to be used.

AI USE STATEMENT

In this work, we used generative AI tools for literature searching, checking the manuscript for typos, assisting profiling and developing CUDA kernels, generating unit tests with pytest and hypothesis, autocomplete in the JetBrains IDE, and debugging. We have not used generative AI tools for generating drafts of the manuscript, editing the manuscript beyond basic grammatical errors and typos, generating mathematical arguments, model design decisions, or making other scope-related decisions. Additionally, we used generative AI tools for formatting booktabs tables, managing project dependencies via uv, generating docstrings, and other routine code-base maintenance. We have thoroughly reviewed all AI-assisted work, and we have read and revised all generated documentation and pytest scripts. We have handwritten test suites to ensure the AI assisted custom CUDA kernel paths agrees with the slow PyTorch implementations to float32 precision. We take full responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The supplementary materials provides the source code and designated directories for reproducing the results presented in this paper. This includes the exact configuration YAMLs and the learning rate grids used to optimize the models. A detailed proof of our theoretical contribution is presented in Appendix A.2, with step by step justification. We site the key theoretical foundation for our work (Weiler et al., 2023) to back up our definitions.

REFERENCES

Iro Armeni, Sasha Sax, Amir R. Zamir, and Silvio Savarese. Joint 2D-3D-semantic data for indoor scene understanding. *arXiv preprint arXiv:1702.01105*, 2017. URL https://arxiv.org/abs/1702.01105.

Kendall Atkinson and Weimin Han. *Spherical Harmonics and Approximations on the Unit Sphere: An Introduction*, volume 2044 of *Lecture Notes in Mathematics*. Springer, 2012. doi: 10.1007/978-3-642-25983-8. URL https://link.springer.com/book/10.1007/978-3-642-25983-8.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization, 2016. URL https://arxiv.org/abs/1607.06450.

Sourya Basu, Jose Gallego-Posada, Francesco Viganò, James Rowbottom, and Taco Cohen. Equivariant mesh attention networks. *Transactions on Machine Learning Research*, 2022. URL https://openreview.net/forum?id=3IqqJh2Ycy.

Yaniv Benny and Lior Wolf. SphereUFormer: A U-shaped transformer for spherical 360 perception. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 940–950, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Benny_SphereUFormer_A_U-Shaped_Transformer_for_Spherical_360_Perception_CVPR_2025_paper.html.

Boris Bonev, Thorsten Kurth, Christian Hundt, Jaideep Pathak, Maximilian Baust, Karthik Kashinath, and Anima Anandkumar. Spherical Fourier neural operators: Learning stable dynamics on the sphere. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 2806–2823. PMLR, 2023. URL https://proceedings.mlr.press/v202/bonev23a.html.

Boris Bonev, Max Rietmann, Andrea Paris, Alberto Carpentieri, and Thorsten Kurth. Attention on the sphere. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-2062. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/58fb4d7e4e5e6e03ff244b5e39805e98-Abstract-Conference.html.

Edoardo Calvello, Nikola B. Kovachki, Matthew E. Levine, and Andrew M. Stuart. Continuum attention for neural operators. *Journal of Machine Learning Research*, 26(300):1–52, 2025. URL https://www.jmlr.org/papers/v26/24-0879.html.

John Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):679–698, 1986. doi: 10.1109/TPAMI.1986.4767851.

Oscar Carlsson, Jan E. Gerken, Hampus Linander, Heiner Spieß, Fredrik Ohlsson, Christoffer Petersson, and Daniel Persson. HEAL-SWIN: A vision transformer on the sphere. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6067–6077, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Carlsson_HEAL-SWIN_A_Vision_Transformer_On_The_Sphere_CVPR_2024_paper.html.

Evangelos Chatzipantazis, Stefanos Pertigkiozoglou, Edgar Dobriban, and Kostas Daniilidis. SE(3)-equivariant attention networks for shape reconstruction in function space. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=RDy3IbvjMqT.

Taco Cohen, Maurice Weiler, Berkay Kicanaoglu, and Max Welling. Gauge equivariant convolutional networks and the icosahedral CNN. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 1321–1330. PMLR, 2019. URL https://proceedings.mlr.press/v97/cohen19d.html.

Stefan Elfwing, Eiji Uchibe, and Kenji Doya. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural Networks*, 107:3–11, 2018. doi: 10.1016/j.neunet.2017.12.012. URL https://arxiv.org/abs/1702.03118.

Fabian B. Fuchs, Daniel E. Worrall, Volker Fischer, and Max Welling. SE(3)-transformers: 3D roto-translation equivariant attention networks. In *Advances in Neural Information Processing Systems*, volume 33, pp. 1970–1981, 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/hash/15231a7ce4ba789d13b722cc5c955834-Abstract.html.

K. M. Górski, E. Hivon, A. J. Banday, B. D. Wandelt, F. K. Hansen, M. Reinecke, and M. Bartelmann. HEALPix: A framework for high-resolution discretization and fast analysis of data distributed on the sphere. *The Astrophysical Journal*, 622(2):759–771, 2005. doi: 10.1086/427976. URL https://arxiv.org/abs/astro-ph/0409513.

Lingshen He, Yiming Dong, Yisen Wang, Dacheng Tao, and Zhouchen Lin. Gauge equivariant transformer. In *Advances in Neural Information Processing Systems*, volume 34, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/e57c6b956a6521b28495f2886ca0977a-Abstract.html.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015. URL https://arxiv.org/abs/1412.6980.

Hans Knutsson. Representing local structure using tensors. In *Proceedings of the 6th Scandinavian Conference on Image Analysis*, pp. 244–251, Oulu, Finland, 1989. URL https://www.diva-portal.org/smash/get/diva2:274015/FULLTEXT01.pdf.

John M. Lee. *Introduction to Riemannian Manifolds*, volume 176 of *Graduate Texts in Mathematics*. Springer, Cham, 2nd edition, 2018. doi: 10.1007/978-3-319-91755-9. URL https://link.springer.com/book/10.1007/978-3-319-91755-9.

Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. In *International Conference on Learning Representations*, 2017. URL https://arxiv.org/abs/1608.03983.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Richard M. Murray, Zexiang Li, and S. Shankar Sastry. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, 1994. URL https://www.cds.caltech.edu/~murray/books/MLS/.

Jeremy Ocampo, Matthew A. Price, and Jason D. McEwen. Scalable and equivariant spherical CNNs by discrete-continuous (DISCO) convolutions. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=eb_cpjZZ3GH.

Nathanaël Perraudin, Michaël Defferrard, Tomasz Kacprzak, and Raphael Sgier. DeepSphere: Efficient spherical convolutional neural network with HEALPix sampling for cosmological applications. *Astronomy and Computing*, 27:130–146, 2019. doi: 10.1016/j.ascom.2019.03.004. URL https://arxiv.org/abs/1810.12186.

Frederic P. Schuller. Lectures on the geometric anatomy of theoretical physics. Lecture Series, Friedrich-Alexander-Universität Erlangen-Nürnberg, 2016.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, pp. 5998–6008, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

Joachim Weickert. *Anisotropic Diffusion in Image Processing*. ECMI Series. B. G. Teubner, Stuttgart, 1998. URL https://www.mia.uni-saarland.de/weickert/book.html.

Maurice Weiler and Gabriele Cesa. General $E(2)$-equivariant steerable CNNs. In *Advances in Neural Information Processing Systems*, volume 32, 2019. URL https://papers.nips.cc/paper_files/paper/2019/hash/45d6637b718d0f24a237069fe41b0db4-Abstract.html.

Maurice Weiler, Mario Geiger, Max Welling, Wouter Boomsma, and Taco S. Cohen. 3D steerable CNNs: Learning rotationally equivariant features in volumetric data. In *Advances in Neural Information Processing Systems*, volume 31, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/488e4104520c6aab692863cc1dba45af-Abstract.html.

Maurice Weiler, Patrick Forré, Erik Verlinde, and Max Welling. Equivariant and coordinate independent convolutional networks: A gauge field theory of neural networks. Online monograph, 2023. URL https://maurice-weiler.gitlab.io/cnn_book/EquivariantAndCoordinateIndependentCNNs.pdf.

David L. Williamson, John B. Drake, James J. Hack, Rüdiger Jakob, and Paul N. Swarztrauber. A standard test set for numerical approximations to the shallow water equations in spherical geometry. *Journal of Computational Physics*, 102(1):211–224, 1992. doi: 10.1016/S0021-9991(05)80016-6.

Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M. Alvarez, and Ping Luo. SegFormer: Simple and efficient design for semantic segmentation with transformers, 2021. URL https://arxiv.org/abs/2105.15203.

Ruibin Xiong, Yunchang Yang, Di He, Kai Zheng, Shuxin Zheng, Chen Xing, Huishuai Zhang, Yanyan Lan, Liwei Wang, and Tieyan Liu. On layer normalization in the transformer architecture. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 10524–10533. PMLR, 2020. URL https://proceedings.mlr.press/v119/xiong20b.html.

A Formal Treatment of GM-Aggregation

The formal framework underlying GM-aggregation was developed in Weiler et al. (2023). Our contribution is the extension of their work to the more general setting of field-dependent kernels and thus provides an attention mechanism. The proof of Theorem 1 follows an argument similar to their proof that unnormalized GM-convolution is Isom_{GM} equivariant (Weiler et al., 2023, Theorem 13.2.5).

A.1 DEFINITIONS

G-structures and feature fields. Let $\pi_{FM} : FM \to M$ denote the frame bundle of M, whose fiber over p consists of all ordered bases of the tangent space at p. Equipping each fiber F_pM with the right $\mathrm{GL}(d)$-action

$$(e_i)_{i=1}^d \triangleleft g := \left(\sum_{j=1}^d e_j g_{ji} \right)_{i=1}^d$$

makes FM a principal $\mathrm{GL}(d)$-bundle. A G-structure is a smooth principal G-subbundle $\pi_{GM} : GM \to M$ of FM, where G is a Lie subgroup of $GL(d)$.

For a smooth representation $\rho : G \to \mathrm{GL}(c)$, define the corresponding associated feature vector bundle to be the orbit space

$$GM \times_\rho \mathbb{R}^c := (GM \times \mathbb{R}^c)/G$$

of the right G-action on $GM \times \mathbb{R}^c$ given by

$$\big((e_i)_{i=1}^d, f\big) \cdot g := \big((e_i)_{i=1}^d \triangleleft g, \rho(g^{-1})f\big).$$

The elements of $GM \times_\rho \mathbb{R}^c$ are equivalence classes $[((e_i)_{i=1}^d, f)]$, where the first component of a representative specifies a tangent frame and the second specifies feature coordinates in the frame it induces on the associated fiber. By construction, $GM \times_\rho \mathbb{R}^c$ is a smooth vector bundle of rank c that is G-associated to GM. We define a c-channel feature field of type ρ on M to be a smooth global section of $GM \times_\rho \mathbb{R}^c$.

GM-Aggregation. Let $\mathcal{A}_{\mathrm{in}} = GM \times_{\rho_{\mathrm{in}}} \mathbb{R}^{c_{\mathrm{in}}}$ and $\mathcal{A}_{\mathrm{out}} = GM \times_{\rho_{\mathrm{out}}} \mathbb{R}^{c_{\mathrm{out}}}$ be associated feature vector bundles. For $f \in \Gamma(\mathcal{A}_{\mathrm{in}})$, define the transporter pullback of f to be the map $\mathrm{Exp}^* f : \mathrm{dom}(\mathrm{Exp}) \to \mathcal{A}_{\mathrm{in}}$ given on each tangent fiber by $\mathrm{Exp}_p^* f(v) := \mathcal{P}_{\mathcal{A}_{\mathrm{in}}, p \leftarrow \exp_p(v)}\big(f(\exp_p(v))\big)$, where $p = \pi_{TM}(v)$ and transport follows the reverse of the geodesic $t \mapsto \exp_p(tv)$, $t \in [0,1]$ (Weiler et al., 2023, Definition 12.2.4). This is the formal coordinate-free definition of $\hat{f}$ from § 3.

An unconstrained aggregation transform is a map $A : \Gamma(\mathcal{A}_{\mathrm{in}}) \to \Gamma(\mathcal{A}_{\mathrm{out}})$ that is given pointwise by

$$(Af)(p) = \frac{\displaystyle\int_{T_pM} \chi(v)\, \beta\big(\alpha(f(p), \hat{f}_p(v))\big)\, \mathcal{K}(v) \hat{f}_p(v)\, dv}{\displaystyle\int_{T_pM} \chi(v)\, \beta\big(\alpha(f(p), \hat{f}_p(v))\big)\, dv}, \tag{6}$$

where dv is the Lebesgue measure induced by the Riemannian metric on T_pM and $\hat{f}_p := \mathrm{Exp}_p^* f$. Here, $\chi : TM \to \mathbb{R}_{\geq 0}$, with $0_p \in \mathrm{supp}(\chi_p) \Subset \mathrm{dom}(\exp_p)$, is called the aggregation window; $\beta : \mathbb{R} \to \mathbb{R}_{>0}$ is the affinity activation; $\alpha : \mathcal{A}_{\mathrm{in}} \oplus_M \mathcal{A}_{\mathrm{in}} \to \mathbb{R}$ is the affinity; and $\mathcal{K} : TM \to \mathrm{Hom}(\mathcal{A}_{\mathrm{in}}, \mathcal{A}_{\mathrm{out}})$ is the value transformation kernel, satisfying $\mathcal{K}(v) \in \mathrm{Hom}(\mathcal{A}_{\mathrm{in},p}, \mathcal{A}_{\mathrm{out},p})$ for $v \in T_pM$. The integrands are extended by zero outside $\mathrm{dom}(\exp_p)$. Each component is assumed to be smooth, and the transform is well defined whenever both integrals are finite, the denominator is nonzero, and the resulting field is smooth.

A GM-aggregation is an aggregation transform whose components are determined by templates that are independent of spatial position and gauge. Specifically, there exist templates $c : \mathbb{R}^d \to \mathbb{R}_{\geq 0}$, $a : \mathbb{R}^{c_{\mathrm{in}}} \oplus \mathbb{R}^{c_{\mathrm{in}}} \to \mathbb{R}$, and $K : \mathbb{R}^d \to \mathbb{R}^{c_{\mathrm{out}} \times c_{\mathrm{in}}}$ such that, for every $p \in M$ and every compatible choice of fiber identifications,

$$\chi_p = c \circ \psi_{TM,p}, \tag{7}$$

$$\alpha_p = a \circ [\psi_{\mathcal{A}_{\mathrm{in}},p} \oplus \psi_{\mathcal{A}_{\mathrm{in}},p}], \tag{8}$$

$$\mathcal{K}_p = (\psi_{\mathrm{Hom},p})^{-1} \circ K \circ \psi_{TM,p}, \tag{9}$$

where χ_p, α_p, and $\mathcal{K}_p$ denote the corresponding fiber restrictions, and the fiber identifications $\psi_{\bullet,p}$ are induced by a common local trivialization of GM in the chosen G-atlas. This requires c to be G-invariant, a to be gauge invariant, and K to be gauge equivariant, satisfying the constraint in equation 3.

Equation 6 is the formal coordinate-free version of equation 2. In the coordinate formulation, we absorb β into α and take χ implicitly to be the indicator function of $\mathcal{N}$. This relaxes the smoothness assumption on χ, but, as shown in the proof of Theorem 1, the indicator of a G-invariant neighborhood with compact closure preserves equivariance whenever the aggregation remains well defined.

Isometry actions. For $\phi \in \mathrm{Isom}(M)$, let $\phi_{*,TM}$ denote the pushforward action of ϕ on TM. Then ϕ acts on the frame bundle by $\phi_{*,FM}((e_i)_{i=1}^d) := (\phi_{*,TM}(e_i))_{i=1}^d$. We define the subgroup $\mathrm{Isom}_{GM} := \{\phi \in \mathrm{Isom}(M) : \phi_{*,FM}(G_pM) = G_{\phi(p)}M \text{ for all } p \in M\}$. This subgroup acts on GM by restriction: $\phi_{*,GM} := \phi_{*,FM}|_{GM}$ for $\phi \in \mathrm{Isom}_{GM}$. The induced action of Isom_{GM} on an associated feature vector bundle $\mathcal{A} = GM \times_\rho \mathbb{R}^c$ is $\phi_{*,\mathcal{A}}([((e_i)_{i=1}^d, f)]) := [(\phi_{*,GM}((e_i)_{i=1}^d), f)]$. Finally, the action of an isometry $\phi \in \mathrm{Isom}_{GM}$ on a feature field $f \in \Gamma(\mathcal{A})$ is

$$\phi \rhd_\rho f := \phi_{*,\mathcal{A}} \circ f \circ \phi^{-1}. \tag{10}$$

A.2 Proof of Theorem 1

Let $\mathcal{A}_{\mathrm{in}} = GM \times_{\rho_{\mathrm{in}}} \mathbb{R}^{c_{\mathrm{in}}}$ and $\mathcal{A}_{\mathrm{out}} = GM \times_{\rho_{\mathrm{out}}} \mathbb{R}^{c_{\mathrm{out}}}$ be associated feature vector bundles, α a GM-affinity function with template a, β an affinity activation, $\mathcal{K}$ a GM-value transformation with template K, and χ a GM-aggregation window with template c. Fix $f \in \Gamma(\mathcal{A}_{\mathrm{in}})$. We assume that the aggregation is well defined and use the zero-extension convention outside the domain of the exponential map. For brevity, write the transporter pullback as $\hat{f} := \mathrm{Exp}^* f$ and define the field-dependent aggregation weight by

$$w_f(v) := \chi(v)\beta\big(\alpha(f(p), \hat{f}_p(v))\big), \tag{11}$$

where $v \in T_pM$. Define the field-dependent kernel by

$$\mathcal{K}_f : TM \to \mathrm{Hom}(\mathcal{A}_{\mathrm{in}}, \mathcal{A}_{\mathrm{out}}), \qquad \mathcal{K}_f(v) := w_f(v)\mathcal{K}(v). \tag{12}$$

Then setting

$$N_f(p) := \int_{T_pM} \mathcal{K}_f(v)\hat{f}_p(v)\, dv, \qquad C_f(p) := \int_{T_pM} w_f(v)\, dv \tag{13}$$

gives the GM-aggregation transform as

$$(Af)(p) = \frac{N_f(p)}{C_f(p)}.$$

To prove the desired isometry equivariance of GM-aggregation, it suffices to show that the numerator and denominator transform equivariantly:

$$N_{\phi \rhd_{\rho_{\mathrm{in}}} f} = \phi \rhd_{\rho_{\mathrm{out}}} N_f, \qquad C_{\phi \rhd_{\rho_{\mathrm{in}}} f} = C_f \circ \phi^{-1}, \qquad \phi \in \mathrm{Isom}_{GM}.$$

Below, we suppress the representation subscript when it is determined by the field type and write $\phi \rhd C_f := C_f \circ \phi^{-1}$ for the scalar-field action. The induced action on each scalar fiber is the identity. All bundle maps in the following compositions are restricted to the relevant fibers, and $\phi_{*,\bullet}^{-1}$ denotes $(\phi^{-1})_{*,\bullet} = (\phi_{*,\bullet})^{-1}$.

First, we show that $f \mapsto C_f$ is Isom_{GM}-equivariant. Let $\phi \in \mathrm{Isom}_{GM}$ and $p \in M$. Then

$$
\begin{aligned}
C_{\phi \triangleright f}(p) &\overset{(1)}{=} \int_{T_pM} w_{\phi \triangleright f}(v)\, dv \\
&\overset{(2)}{=} \int_{T_pM} \chi(v)\beta\big(\alpha((\phi \triangleright f)(p), (\widehat{\phi \triangleright f})_p(v))\big)\, dv \\
&\overset{(3)}{=} \int_{T_pM} \chi(v)\beta\Big(\alpha\Big(\phi_{*,\mathcal{A}_{\mathrm{in}}} f(\phi^{-1}(p)), \phi_{*,\mathcal{A}_{\mathrm{in}}} \hat{f}_{\phi^{-1}(p)}(\phi^{-1}_{*,TM} v)\Big)\Big)\, dv \\
&\overset{(4)}{=} \int_{T_{\phi^{-1}(p)}M} \chi(\phi_{*,TM}\tilde{v})\beta\Big(\alpha\Big(\phi_{*,\mathcal{A}_{\mathrm{in}}} f(\phi^{-1}(p)), \phi_{*,\mathcal{A}_{\mathrm{in}}} \hat{f}_{\phi^{-1}(p)}(\tilde{v})\Big)\Big)\, d\tilde{v} \\
&\overset{(5)}{=} \int_{T_{\phi^{-1}(p)}M} \chi(\tilde{v})\beta\Big(\alpha\Big(\phi_{*,\mathcal{A}_{\mathrm{in}}} f(\phi^{-1}(p)), \phi_{*,\mathcal{A}_{\mathrm{in}}} \hat{f}_{\phi^{-1}(p)}(\tilde{v})\Big)\Big)\, d\tilde{v} \\
&\overset{(6)}{=} \int_{T_{\phi^{-1}(p)}M} \chi(\tilde{v})\beta\big(\alpha(f(\phi^{-1}(p)), \hat{f}_{\phi^{-1}(p)}(\tilde{v}))\big)\, d\tilde{v} \\
&\overset{(7)}{=} \int_{T_{\phi^{-1}(p)}M} w_f(\tilde{v})\, d\tilde{v} \\
&\overset{(8)}{=} C_f(\phi^{-1}(p)) \\
&\overset{(9)}{=} [\phi \triangleright C_f](p).
\end{aligned}
$$

Step (1) uses the definition of C_f in equation 13. Step (2) expands $w_{\phi \triangleright f}$ using equation 11. Step (3) uses the field action equation 10 and the transporter-pullback identity

$$
(\widehat{\phi \triangleright f})_p(v) = \phi_{*,\mathcal{A}_{\mathrm{in}}} \hat{f}_{\phi^{-1}(p)}(\phi^{-1}_{*,TM} v), \tag{14}
$$

which holds under the connection hypotheses of Theorem 13.1.4 of Weiler et al. (2023). The argument applies whenever the chosen feature-bundle transport is compatible with the isometry action in this sense. Step (4) makes the change of variables $\tilde{v} = \phi^{-1}_{*,TM} v$. Since ϕ is an isometry, its tangent map preserves the metric-induced Lebesgue measures, so no Jacobian factor is introduced.

For step (5), choose a compatible gauge about p. Its pullback by ϕ is a compatible gauge about $\phi^{-1}(p)$ because ϕ preserves the G-structure. The corresponding tangent-fiber identification is

$$
\psi_{TM,\phi^{-1}(p)} := \psi_{TM,p} \circ \phi_{*,TM,\phi^{-1}(p)}.
$$

The shared window template in equation 7 therefore gives

$$
\begin{aligned}
\chi(\phi_{*,TM}\tilde{v}) &= c\big(\psi_{TM,p}(\phi_{*,TM}\tilde{v})\big) \\
&= c\big(\psi_{TM,\phi^{-1}(p)}(\tilde{v})\big) = \chi(\tilde{v}).
\end{aligned}
$$

Similarly, the same choice of gauges induces

$$
\psi_{\mathcal{A}_{\mathrm{in}},\phi^{-1}(p)} := \psi_{\mathcal{A}_{\mathrm{in}},p} \circ \phi_{*,\mathcal{A}_{\mathrm{in}},\phi^{-1}(p)}.
$$

Using the shared affinity template in equation 8, step (6) follows from

$$
\begin{aligned}
&\alpha\Big(\phi_{*,\mathcal{A}_{\mathrm{in}}} f(\phi^{-1}(p)), \phi_{*,\mathcal{A}_{\mathrm{in}}} \hat{f}_{\phi^{-1}(p)}(\tilde{v})\Big) \\
&\quad = a\Big(\psi_{\mathcal{A}_{\mathrm{in}},p}\big(\phi_{*,\mathcal{A}_{\mathrm{in}}} f(\phi^{-1}(p))\big), \psi_{\mathcal{A}_{\mathrm{in}},p}\big(\phi_{*,\mathcal{A}_{\mathrm{in}}} \hat{f}_{\phi^{-1}(p)}(\tilde{v})\big)\Big) \\
&\quad = a\Big(\psi_{\mathcal{A}_{\mathrm{in}},\phi^{-1}(p)}\big(f(\phi^{-1}(p))\big), \psi_{\mathcal{A}_{\mathrm{in}},\phi^{-1}(p)}\big(\hat{f}_{\phi^{-1}(p)}(\tilde{v})\big)\Big) \\
&\quad = \alpha\big(f(\phi^{-1}(p)), \hat{f}_{\phi^{-1}(p)}(\tilde{v})\big).
\end{aligned}
$$

Steps (7)–(9) use the definitions of w_f, C_f, and the scalar-field action, respectively. The same pointwise identities also establish

$$
w_{\phi \triangleright f}(v) = w_f(\phi^{-1}_{*,TM} v). \tag{15}
$$

We now show that $f \mapsto N_f$ is Isom_{GM}-equivariant. First, we prove that the field-dependent kernel transforms as

$$\mathcal{K}_{\phi \triangleright f}(v) = [\phi_{*,\mathcal{K}} \mathcal{K}_f](v) := \phi_{*,\mathcal{A}_{\text{out}}} \circ \mathcal{K}_f(\phi^{-1}_{*,TM} v) \circ \phi^{-1}_{*,\mathcal{A}_{\text{in}}}, \tag{16}$$

where $\phi_{*,\mathcal{K}}$ denotes the induced action on kernel fields defined in Definition 13.2.2 of Weiler et al. (2023). Thus, the kernel induced by the transformed field at p is obtained from the original kernel at $\phi^{-1}(p)$ by the action induced by ϕ.

Choose a compatible gauge about $\phi^{-1}(p)$ and its pushforward gauge about p. For $v \in T_pM$, we have

$$\begin{aligned}
[\phi_{*,\mathcal{K}} \mathcal{K}_f](v) &\overset{(10)}{=} \phi_{*,\mathcal{A}_{\text{out}}} \circ \mathcal{K}_f(\phi^{-1}_{*,TM} v) \circ \phi^{-1}_{*,\mathcal{A}_{\text{in}}} \\
&\overset{(11)}{=} \phi_{*,\mathcal{A}_{\text{out}}} \circ \left[w_f(\phi^{-1}_{*,TM} v) \mathcal{K}(\phi^{-1}_{*,TM} v) \right] \circ \phi^{-1}_{*,\mathcal{A}_{\text{in}}} \\
&\overset{(12)}{=} w_{\phi \triangleright f}(v) \left[\phi_{*,\mathcal{A}_{\text{out}}} \circ \mathcal{K}(\phi^{-1}_{*,TM} v) \circ \phi^{-1}_{*,\mathcal{A}_{\text{in}}} \right] \\
&\overset{(13)}{=} w_{\phi \triangleright f}(v)\, \phi_{*,\mathrm{Hom}}\big(\mathcal{K}(\phi^{-1}_{*,TM} v)\big) \\
&\overset{(14)}{=} w_{\phi \triangleright f}(v) \left[\phi_{*,\mathrm{Hom}} \circ (\psi_{\mathrm{Hom},\phi^{-1}(p)})^{-1} \circ K \circ \psi_{TM,\phi^{-1}(p)} \circ \phi^{-1}_{*,TM} \right](v) \\
&\overset{(15)}{=} w_{\phi \triangleright f}(v) \left[(\psi_{\mathrm{Hom},p})^{-1} \circ K \circ \psi_{TM,p} \right](v) \\
&\overset{(16)}{=} w_{\phi \triangleright f}(v) \mathcal{K}(v) \\
&\overset{(17)}{=} \mathcal{K}_{\phi \triangleright f}(v).
\end{aligned}$$

Step (10) expands the induced kernel-field action in equation 16. Step (11) uses equation 12. Step (12) uses linearity of $\phi_{*,\mathcal{A}_{\text{out}}}$ and the weight identity equation 15. Step (13) uses the induced action on each Hom fiber,

$$\phi_{*,\mathrm{Hom}}(L) := \phi_{*,\mathcal{A}_{\text{out}}} \circ L \circ \phi^{-1}_{*,\mathcal{A}_{\text{in}}}.$$

Step (14) uses the shared kernel template in equation 9. For step (15), the chosen gauges satisfy

$$\psi_{\mathrm{Hom},p} = \psi_{\mathrm{Hom},\phi^{-1}(p)} \circ \phi^{-1}_{*,\mathrm{Hom},p},$$
$$\psi_{TM,p} = \psi_{TM,\phi^{-1}(p)} \circ \phi^{-1}_{*,TM,p}.$$

Steps (16) and (17) use equation 9 and equation 12, respectively.

Having established the transformation law of $\mathcal{K}_f$, we can verify the desired equivariance of the numerator:

$$\begin{aligned}
N_{\phi \triangleright f}(p) &\overset{(18)}{=} \int_{T_pM} \mathcal{K}_{\phi \triangleright f}(v) \widehat{(\phi \triangleright f)}_p(v)\, dv \\
&\overset{(19)}{=} \int_{T_pM} \phi_{*,\mathcal{A}_{\text{out}}} \mathcal{K}_f(\phi^{-1}_{*,TM} v) \phi^{-1}_{*,\mathcal{A}_{\text{in}}} \widehat{(\phi \triangleright f)}_p(v)\, dv \\
&\overset{(20)}{=} \int_{T_pM} \phi_{*,\mathcal{A}_{\text{out}}} \mathcal{K}_f(\phi^{-1}_{*,TM} v) \phi^{-1}_{*,\mathcal{A}_{\text{in}}} \phi_{*,\mathcal{A}_{\text{in}}} \hat{f}_{\phi^{-1}(p)}(\phi^{-1}_{*,TM} v)\, dv \\
&\overset{(21)}{=} \int_{T_pM} \phi_{*,\mathcal{A}_{\text{out}}} \mathcal{K}_f(\phi^{-1}_{*,TM} v) \hat{f}_{\phi^{-1}(p)}(\phi^{-1}_{*,TM} v)\, dv \\
&\overset{(22)}{=} \phi_{*,\mathcal{A}_{\text{out}}} \int_{T_{\phi^{-1}(p)}M} \mathcal{K}_f(\tilde{v}) \hat{f}_{\phi^{-1}(p)}(\tilde{v})\, d\tilde{v} \\
&\overset{(23)}{=} \phi_{*,\mathcal{A}_{\text{out}}} N_f(\phi^{-1}(p)) \\
&\overset{(24)}{=} [\phi \triangleright N_f](p).
\end{aligned}$$

Step (18) uses equation 13. Step (19) uses the kernel-field transformation law equation 16, and step (20) uses the transporter-pullback identity equation 14. Step (21) cancels the mutually inverse maps on the input fiber. Step (22) makes the measure-preserving change of variables $\tilde{v} = \phi^{-1}_{*,TM} v$ and

moves the fixed linear map $\phi_{*,\mathcal{A}_{\text{out}}}$ outside the integral. Steps (23) and (24) use the definition of N_f and the field action equation 10, respectively.

Since $C_f(\phi^{-1}(p)) \neq 0$ and $\phi_{*,\mathcal{A}_{\text{out}}}$ is linear, the two transformation laws give

$$\begin{aligned}[A(\phi \triangleright_{\rho_{\text{in}}} f)](p) &= \frac{\phi_{*,\mathcal{A}_{\text{out}}} N_f(\phi^{-1}(p))}{C_f(\phi^{-1}(p))} \\ &= \phi_{*,\mathcal{A}_{\text{out}}}\big((Af)(\phi^{-1}(p))\big) \\ &= [\phi \triangleright_{\rho_{\text{out}}} (Af)](p).\end{aligned}$$

This proves the claimed equivariance.

B $\mathrm{SO}(S^2)$ Component Implementation

B.1 The Intertwiner and Value Kernel Constraints

Intertwiner constraint. Let

$$\rho_{\text{in}} = a_{n_1}\rho_{n_1} \oplus \cdots \oplus a_{n_N}\rho_{n_N}, \qquad 0 \leq n_1 < \cdots < n_N, \tag{17}$$

$$\rho_{\text{out}} = b_{m_1}\rho_{m_1} \oplus \cdots \oplus b_{m_M}\rho_{m_M}, \qquad 0 \leq m_1 < \cdots < m_M. \tag{18}$$

The intertwiner constraint that projections $W \in \mathbb{R}^{c_{\text{out}} \times c_{\text{in}}}$ must satisfy is

$$W\rho_{\text{in}}(r_\theta) = \rho_{\text{out}}(r_\theta)W, \qquad \forall\, r_\theta \in \mathrm{SO}(2).$$

With these decompositions of ρ_{in} and ρ_{out} into irreducible representations, W is an $M \times N$ block matrix with blocks $W_{m\leftarrow n}$ for $n = n_1, \ldots, n_N$ and $m = m_1, \ldots, m_M$. Each block $W_{m\leftarrow n}$ is itself an equivariant linear map from $a_n\rho_n$ to $b_m\rho_m$ and can be parameterized as

$$W_{m\leftarrow n} = \begin{cases} C_0, & m = n = 0, \\ C_{1,n} \otimes \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + C_{2,n} \otimes \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}, & m = n \geq 1, \\ 0, & m \neq n, \end{cases}$$

where $C_0 \in \mathbb{R}^{b_0 \times a_0}$ and $C_{i,n} \in \mathbb{R}^{b_n \times a_n}$, $i = 1, 2$, are free, learnable parameter matrices in the corresponding cases (Weiler & Cesa, 2019). Notice that blocks mapping between irreducible representations of different types vanish, but copies of the same type can mix.

Value kernel constraint. Let ρ_{in} and ρ_{out} decompose as in equations 17–18. The steerability constraint that a value kernel $K : \mathbb{R}^2 \to \mathbb{R}^{c_{\text{out}} \times c_{\text{in}}}$ must satisfy is

$$K(r_\theta v) = \rho_{\text{out}}(r_\theta)K(v)\rho_{\text{in}}(r_\theta)^{-1}, \qquad \forall\, r_\theta \in \mathrm{SO}(2),\ v \in \mathbb{R}^2.$$

As in the intertwiner case above, the decompositions of ρ_{in} and ρ_{out} imply that, for each $v \in \mathbb{R}^2$, $K(v)$ is an $M \times N$ block matrix with blocks $K_{m\leftarrow n}(v)$ for $n = n_1, \ldots, n_N$ and $m = m_1, \ldots, m_M$. The blocks $K_{m\leftarrow n}$ are themselves steerable kernels from $a_n\rho_n$ to $b_m\rho_m$ (Weiler & Cesa, 2019). Writing

$$\begin{aligned} c_k(\phi) &:= \cos(k\phi), & s_k(\phi) &:= \sin(k\phi), \\ c_-(\phi) &:= \cos((m-n)\phi), & s_-(\phi) &:= \sin((m-n)\phi), \\ c_+(\phi) &:= \cos((m+n)\phi), & s_+(\phi) &:= \sin((m+n)\phi), \end{aligned}$$

the block $K_{m\leftarrow n}(r,\phi)$ in polar coordinates, for $r>0$, is given by

$$\begin{cases} C_1^{0\leftarrow 0}(r), & m=n=0,\\ C_1^{m\leftarrow 0}(r)\otimes\begin{pmatrix}c_m(\phi)\\ s_m(\phi)\end{pmatrix}+C_2^{m\leftarrow 0}(r)\otimes\begin{pmatrix}-s_m(\phi)\\ c_m(\phi)\end{pmatrix}, & n=0,\ m\geq 1,\\ C_1^{0\leftarrow n}(r)\otimes\begin{pmatrix}c_n(\phi) & s_n(\phi)\end{pmatrix}+C_2^{0\leftarrow n}(r)\otimes\begin{pmatrix}-s_n(\phi) & c_n(\phi)\end{pmatrix}, & n\geq 1,\ m=0,\\ \begin{aligned}&C_1^{m\leftarrow n}(r)\otimes\begin{pmatrix}c_-(\phi) & -s_-(\phi)\\ s_-(\phi) & c_-(\phi)\end{pmatrix}+C_2^{m\leftarrow n}(r)\otimes\begin{pmatrix}-s_-(\phi) & -c_-(\phi)\\ c_-(\phi) & -s_-(\phi)\end{pmatrix}\\ &+C_3^{m\leftarrow n}(r)\otimes\begin{pmatrix}c_+(\phi) & s_+(\phi)\\ s_+(\phi) & -c_+(\phi)\end{pmatrix}+C_4^{m\leftarrow n}(r)\otimes\begin{pmatrix}-s_+(\phi) & c_+(\phi)\\ c_+(\phi) & s_+(\phi)\end{pmatrix}\end{aligned}, & m,n\geq 1. \end{cases}$$

Here $C_i^{m\leftarrow n}:\mathbb{R}_{>0}\to\mathbb{R}^{b_m\times a_n}$ are learnable matrix-valued functions of the radial coordinate. Our implementation parameterizes them using Gaussian radial basis functions. At the origin, $K_{m\leftarrow n}(0)$ must be an intertwiner. In our implementation, we anneal the radial functions to zero as $r\to 0$ and set $K(0)=0$ so that the kernel is continuous at the origin. To allow for self interaction, we use a pointwise linear map outside of the aggregation integral. Unlike intertwiners, steerable value kernels can map nontrivially between fields of different types, which is the mechanism we use in our experiments to lift the scalar input to a latent geometric field.

B.2 $\mathrm{SO}(S^2)$ Transformer Block Components

In this subsection, we assume that our latent field type is

$$\rho=a_{n_1}\rho_{n_1}\oplus\cdots\oplus a_{n_N}\rho_{n_N},\qquad 0\leq n_1<\cdots<n_N.$$

Then, in coordinates, a feature vector of type ρ has the form

$$x=\Big(\big(x_a^{(n_i)}\big)_{a=1}^{a_{n_i}}\Big)_{i=1}^{N},\qquad x_a^{(0)}\in\mathbb{R},\qquad x_a^{(n_i)}\in\mathbb{R}^2\quad(n_i>0). \tag{19}$$

Denote the dimension of ρ_{n_i} by d_{n_i}, so that $d_{n_i}=1$ if $n_i=0$ and $d_{n_i}=2$ if $n_i>0$. Thus, $x\in\mathbb{R}^c$ with $c=\sum_{i=1}^N a_{n_i}d_{n_i}$.

SO2-LayerNorm. We use a gauge-equivariant, pointwise layer normalization. We center scalar channels; if $n_1=0$, define

$$\mu_0(x):=\frac{1}{a_0}\sum_{a=1}^{a_0}x_a^{(0)},$$
$$v_0(x):=\frac{1}{a_0}\sum_{a=1}^{a_0}\big(x_a^{(0)}-\mu_0(x)\big)^2.$$

For $n_i>0$, define

$$v_{n_i}(x):=\frac{1}{2a_{n_i}}\sum_{a=1}^{a_{n_i}}\big\|x_a^{(n_i)}\big\|^2.$$

We obtain the normalization denominator as

$$D(x):=\sqrt{\frac{1}{N}\sum_{i=1}^{N}v_{n_i}(x)+\epsilon},\qquad \epsilon>0,$$

and define the layer normalization by

$$\mathrm{SO2LN}(x)_a^{(0)}=\gamma_a^{(0)}\frac{x_a^{(0)}-\mu_0(x)}{D(x)}+\beta_a^{(0)},$$
$$\mathrm{SO2LN}(x)_a^{(n_i)}=\gamma_a^{(n_i)}\frac{x_a^{(n_i)}}{D(x)},\qquad n_i>0,$$

where the scalar expression applies only if $n_1 = 0$. Here $\gamma_a^{(n_i)} \in \mathbb{R}$ is a learned scalar gain for each irreducible copy, and $\beta_a^{(0)} \in \mathbb{R}$ is a learned bias for each scalar copy. Since $D(x)$ is invariant under rotations, we have

$$\begin{aligned}\mathrm{SO2LN}\big(\rho(r_\theta)x\big)_a^{(n_i)} &= \gamma_a^{(n_i)} \frac{r_{n_i\theta} x_a^{(n_i)}}{D(x)} \\ &= r_{n_i\theta} \mathrm{SO2LN}(x)_a^{(n_i)}\end{aligned}$$

for all $n_i > 0$ and $r_\theta \in \mathrm{SO}(2)$. The scalar outputs are invariant under rotations. Thus, SO2LN is gauge-equivariant and reduces to standard layer normalization for purely scalar fields (Ba et al., 2016).

SO2-Nonlinearities. While several gauge-equivariant nonlinearities exist, we use a scaled sigmoid scalar gate (Weiler et al., 2018). Assume that $n_1 = 0$, and write $x^{(0)} = (x_a^{(0)})_{a=1}^{a_0}$. Then define

$$\begin{aligned}\mathrm{SO2NonLinear}(x)^{(0)} &:= \mathrm{SiLU}\big(x^{(0)}\big), \\ \mathrm{SO2NonLinear}(x)_a^{(n_i)} &:= 2\sigma\Big(\big(A_{n_i} x^{(0)} + b_{n_i}\big)_a\Big) x_a^{(n_i)}, \qquad n_i > 0.\end{aligned}$$

Here SiLU (Elfwing et al., 2018) is applied componentwise, σ denotes the sigmoid function, $A_{n_i} \in \mathbb{R}^{a_{n_i} \times a_0}$ is a learned matrix, and $b_{n_i} \in \mathbb{R}^{a_{n_i}}$ is a learned bias vector.

SO(S^2)-MHSA. Let $H \geq 1$ be an integer that divides a_{n_i} for each $i = 1, \ldots, N$. Define $c_{\mathrm{attn}} := c/H$ and

$$\rho_{\mathrm{attn}} := \frac{a_{n_1}}{H}\rho_{n_1} \oplus \cdots \oplus \frac{a_{n_N}}{H}\rho_{n_N}.$$

For $h = 1, \ldots, H$, let $W_Q^{(h)}, W_K^{(h)} \in \mathbb{R}^{c_{\mathrm{attn}} \times c}$ be intertwiners from ρ to ρ_{attn}, and let $W_V^{(h)} : \mathbb{R}^2 \to \mathbb{R}^{c_{\mathrm{attn}} \times c}$ be a steerable value kernel from ρ to ρ_{attn}. With respect to a discretization $\mathcal{D}$ (see Section 4), $\mathrm{SO}(S^2)$ attention is given by

$$\mathrm{SO}(S^2)\mathrm{Attn}_{\mathcal{D}}^{(h)}(f)_i = \frac{\displaystyle\sum_{j \in \mathcal{N}_i} q_{ij} \exp\left(\frac{\big(W_Q^{(h)} f_i\big)^\top \big(W_K^{(h)} f_{i \leftarrow j}\big)}{\sqrt{c_{\mathrm{attn}}}}\right) W_V^{(h)}(v_{ij}) f_{i \leftarrow j}}{\displaystyle\sum_{j \in \mathcal{N}_i} q_{ij} \exp\left(\frac{\big(W_Q^{(h)} f_i\big)^\top \big(W_K^{(h)} f_{i \leftarrow j}\big)}{\sqrt{c_{\mathrm{attn}}}}\right)}.$$

Then,

$$\mathrm{SO}(S^2)\mathrm{MHSA}_{\mathcal{D}}(f)_i = W_O \left(\bigoplus_{h=1}^{H} \mathrm{SO}(S^2)\mathrm{Attn}_{\mathcal{D}}^{(h)}(f)_i\right),$$

where $W_O \in \mathbb{R}^{c \times c}$ is an intertwiner from $\bigoplus_{h=1}^{H} \rho_{\mathrm{attn}} \cong \rho$ to ρ.

C Shallow Water Experiment

C.1 Losses

The loss used to train a model on a discretization $\mathcal{D}$ with spherical quadrature rule $Q = \{q_i\}_{i=1}^n$ is

$$Q\big[\mathcal{L}_2^2(\hat{z}, z)\big] := \frac{1}{3}\sum_{c=1}^{3} \frac{1}{4\pi} \sum_{i=1}^{n} q_i \, (\hat{z}_{c,i} - z_{c,i})^2, \tag{20}$$

which approximates the average of the continuous squared $\mathcal{L}_2$ loss over the normalized channels,

$$\mathcal{L}_2^2(\hat{z}, z) := \frac{1}{3}\sum_{c=1}^{3} \frac{1}{4\pi} \int_{S^2} \big(\hat{z}_c(p) - z_c(p)\big)^2 \, d\mu(p),$$

where $d\mu$ is the spherical surface measure, with $\mu(S^2) = 4\pi$. This grid-dependent quadrature approximation is what is reported in our tables. When a table contains models evaluated on two different discretizations, we also report a band-limited approximation of the loss on a shared basis of spherical harmonics of degree at most $\ell_{\max}$. In particular, for a fixed model, let $e_{c,i} = \hat{z}_{c,i} - z_{c,i}$ be the error field in channel c on the model's grid. We fit e_c to the real orthonormal spherical harmonics of degree at most $\ell_{\max}$ (Atkinson & Han, 2012) via weighted least squares,

$$\hat{a}_c = \arg\min_{a_c} \sum_i q_i \Big(\sum_j a_{c,j} Y_j(p_i) - e_{c,i} \Big)^2,$$

over the $(\ell_{\max}+1)^2$ harmonics Y_j of degree $\ell_j \le \ell_{\max}$. We then define the band-limited error as

$$E_{\le \ell_{\max}} := \frac{1}{3} \sum_{c=1}^{3} \frac{1}{4\pi} \Big\| \sum_j \hat{a}_{c,j} Y_j \Big\|^2_{L^2(S^2, d\mu)} = \sum_{\ell=0}^{\ell_{\max}} D(\ell), \tag{21}$$

where $D_c(\ell) = \frac{1}{4\pi} \sum_{j:\ell_j=\ell} \hat{a}^2_{c,j}$, $D(\ell) = \frac{1}{3} \sum_{c=1}^{3} D_c(\ell)$, and the second equality holds by the orthonormality of the spherical harmonics with respect to $d\mu$ (Atkinson & Han, 2012). We set $\ell_{\max} = 85$ for the 256×512 solver configuration and $\ell_{\max} = 42$ for the 128×256 benchmark, matching their spectral truncations. Table 4 decomposes the fitted error by degree. We highlight a limitation of the band-limited error. The weighted least squares fit is performed on the evaluation grid, and out-of-band content can alias into the projection.

Table 4: Distribution of error energy on the 256×512 configuration with 1 hour lead and 64 test trajectories. The band-limited $\mathcal{L}^2_2$, then the fraction of it in each spherical-harmonic degree band.

			share by degree band				
Model	horizon	$10^3 \times \mathcal{L}^2_2$	0	1–3	4–23	24–47	48–85
Depth-1 ablation							
D1 $SO(S^2)$ MHSA, $\rho_0^{16} \oplus \rho_1^{16} \oplus \rho_2^{16}$	1 step	4.14	0.05	0.02	0.04	0.12	0.78
D1 $SO(S^2)$ conv, $\rho_0^{16} \oplus \rho_1^{16} \oplus \rho_2^{16}$	1 step	5.75	0.03	0.02	0.04	0.11	0.80
D1 $SO(S^2)$ MHSA, $\rho_0^{26} \oplus \rho_1^{26}$	1 step	5.71	0.03	0.02	0.05	0.11	0.78
D1 $SO(S^2)$ MHSA, ρ_0^{72}	1 step	51.70	0.01	0.00	0.03	0.14	0.82
Zero-shot transfer of the single-level (SL) model, and the reference models							
SL on the L5 icosphere (train)	1 step	0.76	0.18	0.07	0.06	0.15	0.55
	5 h	9.03	0.38	0.38	0.10	0.07	0.07
SL on the L6 icosphere	1 step	2.78	0.05	0.02	0.02	0.04	0.88
	5 h	15.62	0.30	0.37	0.14	0.07	0.11
SL on Legendre–Gauss 128×256	1 step	3.89	0.04	0.01	0.02	0.04	0.90
	5 h	16.91	0.28	0.35	0.15	0.09	0.13
SFNO (256×512, train)	1 step	0.87	0.09	0.02	0.03	0.13	0.73
	5 h	76.39	0.01	0.26	0.27	0.24	0.22
S^2 Transformer, single level (128×256, train)	1 step	5.03	0.01	0.02	0.05	0.14	0.77
	5 h	24923.35	0.00	0.01	0.06	0.44	0.49

C.2 S^2 Transformer Benchmark

The published S^2 Transformer benchmark reports results on a 128×256 solver grid with a lead time of 0.5 hours (Bonev et al., 2025). To fairly represent the S^2 Transformer, we compare our $\mathrm{SO}(S^2)$ Transformer against their released architecture at this resolution. Both models have a matched depth-1 architecture with the same number of transformer blocks on the latent grid. Theirs takes data on a 128×256 grid and restricts to a 64×128 latent grid, while ours takes data on the level-5 icosphere and restricts to the level 4. Though the two models use different grids, the attention neighborhoods and encoder/decoder kernel supports are determined by the same geodesic radii. Table 5 presents the results. We highlight that our $\mathrm{SO}(S^2)$ Transformer with hidden geometric features achieves a 5-8x lower squared error than the scalar S^2 Transformer at each lead time. Moreover, the $\mathrm{SO}(S^2)$ Transformer is within a factor of 1.7 of SFNO at one step on the native grid and 1.4 in the common band, within SFNO's seed spread through 2 hours of rollout, and 40% behind at 2.5 hours. The teacher-forced results indicate that $\mathrm{SO}(S^2)$ Transformer's deficit in rollout is accumulation rather than per-step accuracy.

Table 5: The 128×256 shallow-water benchmark on 64 test trajectories. Metrics reported as mean $\pm$ sd over 3 seeds. $\mathrm{SO}(S^2)$ Transformer is abbreviated $\mathrm{SO}(S^2)$ T. *native*: each model scored on its own grid (Eq. 20). $\ell \leq 42$: the normalized, channel-averaged squared L^2 norm of a weighted least-squares fit to degrees $0, \ldots, 42$ (Eq. 21). *Rollout*: each step is fed the model's own previous prediction. *teacher-forced*: each step is fed the solver's state at the previous lead time. Persistence scores 1255, 1632, 1381, 1378 in the same units at these lead times.

(a) One step (30 min)

		$10^3 \times \mathcal{L}_2^2$		$10^2 \times \mathcal{L}_2$ per channel, native		
Model	Params	native	$\ell \leq 42$	φ	ζ	δ
S^2 Transformer	535,424	3.011 ± 0.142	1.834 ± 0.149	5.23 ± 0.18	4.14 ± 0.15	6.77 ± 0.12
SFNO	541,408	$\mathbf{0.236 \pm 0.047}$	$\mathbf{0.234 \pm 0.046}$	$\mathbf{1.60 \pm 0.23}$	$\mathbf{1.04 \pm 0.07}$	1.84 ± 0.14
$\mathrm{SO}(S^2)$ T. D1 (ours)	491,404	0.402 ± 0.041	0.332 ± 0.031	2.50 ± 0.08	1.54 ± 0.14	$\mathbf{1.84 \pm 0.12}$

(b) Rollout and teacher-forced one step, $10^3 \times \mathcal{L}_2^2$, native

	S^2 Transformer		SFNO		$\mathrm{SO}(S^2)$ T. D1 (ours)	
Lead (h)	rollout	teacher-forced	rollout	teacher-forced	rollout	teacher-forced
1.0	6.823 ± 0.281	3.531 ± 0.097	$\mathbf{1.091 \pm 0.493}$	0.661 ± 0.428	1.274 ± 0.115	$\mathbf{0.425 \pm 0.037}$
1.5	14.668 ± 0.871	3.934 ± 0.276	2.963 ± 1.324	1.203 ± 0.664	$\mathbf{2.924 \pm 0.270}$	$\mathbf{0.550 \pm 0.067}$
2.0	26.407 ± 1.630	3.653 ± 0.209	$\mathbf{4.524 \pm 1.187}$	0.861 ± 0.305	5.453 ± 0.556	$\mathbf{0.661 \pm 0.107}$
2.5	44.779 ± 2.919	3.274 ± 0.130	$\mathbf{6.341 \pm 0.847}$	$\mathbf{0.654 \pm 0.160}$	8.970 ± 1.053	0.738 ± 0.144

Table 6: Per-class recall (%) and accuracy (%) near and away from the class boundaries on the test split. Mean over 3 seeds. Recall is the correctly labeled area of a class over the class's area; "area" is the class's share of the split (%). $\mathrm{SO}(S^2)$, S^2, and $\mathbb{R}^2$ SegFormer columns omit "SegFormer". A boundary pixel has another class in its 4-neighbourhood.

class	area	$\mathrm{SO}(S^2)$	scalar ablation	S^2	$\mathbb{R}^2$	S^2 UNet
wall	28.1	**87.9**	86.5	87.2	87.2	83.5
ceiling	17.9	**93.3**	89.8	92.9	92.2	91.1
floor	16.5	**93.8**	93.0	92.9	92.3	92.7
clutter	12.3	**53.8**	45.4	53.3	48.3	39.7
door	5.3	**84.8**	79.0	83.4	83.2	84.3
bookcase	4.3	**74.8**	71.4	68.2	61.6	74.1
window	3.0	**87.0**	81.5	83.5	73.8	84.1
chair	2.9	**70.9**	63.4	68.4	67.2	66.7
board	2.5	**90.4**	83.2	79.1	78.4	88.0
table	2.4	**65.4**	62.5	63.3	62.8	60.2
unknown	1.7	**31.2**	16.4	26.9	20.5	19.4
column	1.6	**57.0**	19.7	35.8	38.4	7.3
beam	1.1	**68.5**	64.9	63.1	60.0	60.7
sofa	0.4	8.0	0.0	12.2	**20.0**	0.0
accuracy within 2 px of a boundary	31.7	**63.1**	57.9	59.4	57.9	56.2
accuracy elsewhere	68.3	**90.8**	87.1	89.9	88.2	86.6

D STANFORD 2D-3D-S EXPERIMENT

D.1 LOSSES AND METRICS

For an image b, let $z_{b,c} : S^2 \to \mathbb{R}$ denote the logit for class $c \in \{1, \ldots, C\}$, where $C = 14$. These classes comprise the 13 semantic object classes and an additional unknown class, which is included in both training and evaluation. We compute the softmax probabilities and predicted class as

$$p_{b,c}(x) := \frac{\exp z_{b,c}(x)}{\sum_{k=1}^{C} \exp z_{b,k}(x)}, \qquad \hat{y}_b(x) := \operatorname*{arg\,max}_{c \in \{1,\ldots,C\}} p_{b,c}(x).$$

Let $y_b : S^2 \to \{1, \dots, C\}$ denote the target labeling, and let $d\mu$ denote the spherical area measure with $\mu(S^2) = 4\pi$.

We train using the area-normalized cross-entropy loss

$$\mathcal{L}_{\mathrm{CE},b} := -\frac{1}{4\pi} \int_{S^2} \log p_{b,y_b(x)}(x)\, d\mu(x).$$

On a discretization $\mathcal{D}$ with spherical quadrature $Q = \{q_i\}_{i=1}^n$, its quadrature approximation is

$$\mathcal{L}^Q_{\mathrm{CE},b} := -\frac{1}{4\pi} \sum_{i=1}^n q_i \log p_{b,y_b(x_i)}(x_i).$$

The reported metrics are derived from a confusion matrix, which in the continuum is defined as

$$M^{\mathrm{cont}}_{b,cd} := \frac{1}{4\pi} \int_{S^2} \mathbf{1}_{\{y_b(x)=c\}} \mathbf{1}_{\{\hat{y}_b(x)=d\}}\, d\mu(x).$$

Its quadrature approximation, denoted by M_b, is

$$M_{b,cd} := \frac{1}{4\pi} \sum_{i=1}^n q_i \mathbf{1}_{\{y_b(x_i)=c\}} \mathbf{1}_{\{\hat{y}_b(x_i)=d\}}.$$

Rows correspond to target classes and columns to predicted classes. Since every pixel is labeled, $\sum_{c,d} M_{b,cd} = 1$. The quadrature approximations of the normalized correctly and incorrectly classified areas are $T_b := \operatorname{tr} M_b$, $E_b := 1 - T_b$, respectively. The area-weighted pixel accuracy is $P_b := T_b$ and measures the fraction of spherical area that is correctly classified.

For each class c, define the normalized true-positive, false-negative, false-positive, and true-negative areas by

$$\mathrm{TP}_{b,c} := M_{b,cc}, \qquad \mathrm{FN}_{b,c} := \sum_d M_{b,cd} - M_{b,cc},$$

$$\mathrm{FP}_{b,c} := \sum_d M_{b,dc} - M_{b,cc}, \qquad \mathrm{TN}_{b,c} := 1 - \mathrm{TP}_{b,c} - \mathrm{FP}_{b,c} - \mathrm{FN}_{b,c}.$$

The reported intersection-over-union (IoU) pools these areas over classes and then computes the ratio

$$\mathrm{IoU}_b := \frac{\sum_c \mathrm{TP}_{b,c}}{\sum_c \big(\mathrm{TP}_{b,c} + \mathrm{FP}_{b,c} + \mathrm{FN}_{b,c}\big)}. \tag{22}$$

This measures how closely the predicted region overlaps the target region and penalizes both predicting outside the target region and not predicting part of the target region.

To measure class-specific performance, define the class union area

$$U_{b,c} := \sum_d M_{b,cd} + \sum_d M_{b,dc} - M_{b,cc}.$$

For $U_{b,c} > 0$, the class IoU is

$$\mathrm{IoU}_{b,c} := \frac{M_{b,cc}}{U_{b,c}} = \frac{\mathrm{TP}_{b,c}}{\mathrm{TP}_{b,c} + \mathrm{FP}_{b,c} + \mathrm{FN}_{b,c}}.$$

A class absent from both the target and prediction has zero union, so its IoU is undefined and is omitted from the class average. A class present in only the target or only the prediction has positive union and receives IoU zero. Writing

$$\mathcal{C}_b := \{c \in \{1, \dots, C\} : U_{b,c} > 0\},$$

the mean class IoU for image b is

$$\mathrm{mIoU}_b := \frac{1}{|\mathcal{C}_b|} \sum_{c \in \mathcal{C}_b} \mathrm{IoU}_{b,c}. \tag{23}$$

Thus, mIoU_b assigns equal weight to each class present in the target or prediction of that image.

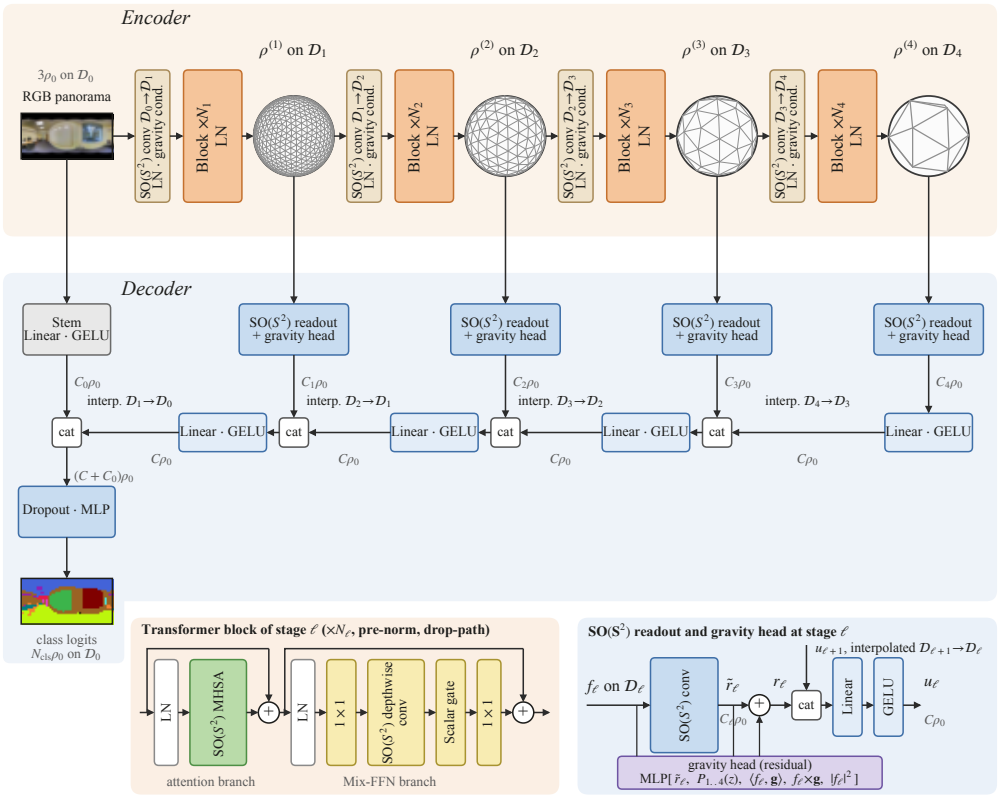

Figure 2: The hierarchical $\mathrm{SO}(S^2)$ SegFormer architecture. For $\ell = 1, \ldots, 4$, $\mathcal{D}_\ell$ is the spherical discretization at stage ℓ and $\rho^{(\ell)}$ is the hidden field type. The discretizations become coarser and the hidden field types become wider as stages progress. The features are scalar after the $\mathrm{SO}(S^2)$ readouts.

D.2 $\mathrm{SO}(S^2)$ SegFormer

Our adaptation of the SegFormer architecture is summarized in Figure 2. The hidden discretizations start at the level 5 icosphere and reduce one level per stage so that stage four is the level 2 icosphere. The field types at each stage are

$$\rho^{(1)} = \rho_0^6 \oplus \rho_1^2 \oplus \rho_2^2 \qquad \rho^{(2)} = \rho_0^{12} \oplus \rho_1^4 \oplus \rho_2^4$$
$$\rho^{(3)} = \rho_0^{18} \oplus \rho_1^6 \oplus \rho_2^6 \qquad \rho^{(4)} = \rho_0^{24} \oplus \rho_1^8 \oplus \rho_2^8.$$

In order, the number of transformer blocks per stage is $3, 4, 6, 3$ with $1, 1, 2, 4$ heads, respectively. The depthwise convolution is an $\mathrm{SO}(S^2)$ convolution whose kernel is such that $C_i^{m \leftarrow n} = 0$ when $m \neq n$ and $C_i^{n \leftarrow n}$ is diagonal, using the parameterized kernel notation from Appendix B.1. The per-stage readout widths are $8, 16, 32, 32$, and the fused decoder width is 32.

The scalar variant uses the same icosphere hierarchy, attention head counts, and decoder structure, with scalar channel widths $[14, 26, 38, 48]$. Its stage depths are $[3, 4, 7, 3]$, compared with $[3, 4, 6, 3]$ for the geometric model, and its drop-path rate is 0.1, compared with 0.2. The scalar variant receives the ρ_1 and ρ_2 gravity features at the input; stagewise latitude conditioning is retained, and its gravity readout heads use the scalar readouts and latitude features. The selected learning rates are 1.5×10^{-3} and 3×10^{-3} for the scalar and geometric models, respectively. Thus, this comparison assesses models with comparable parameter counts but does not isolate hidden field type alone.

The gravity conditioning in our model provides an explicit coordinate-independent reference for "up". Let $e_z := (0, 0, 1)^\top$ be the vertical axis and $p \in S^2$. Define $p_z := \langle p, e_z \rangle \in [-1, 1]$ and $g(p) := e_z - \langle p, e_z \rangle p$. The scalar gravity feature is $G_0(p) := P_{1,\ldots,4}(p_z) := (P_1(p_z), \ldots, P_4(p_z))^\top$, where P_i is the Legendre polynomial of degree i. The ρ_1 gravity feature is the tangent vector

$G_1(p) := g(p)$, which provides the direction of "up", and the ρ_2 gravity feature is $G_2(p) := g(p)^2$, where the square denotes complex squaring in an oriented orthonormal frame. This provides a reference for the vertical axis without distinguishing its two directions. In a gauge w at p, let $G_1^w(p) = (c_1, c_2)^\top$. Then

$$G_2^w(p) = (c_1^2 - c_2^2,\ 2c_1c_2)^\top.$$

Both geometric gravity features are smooth fields that vanish at the poles.

Let x be the latent feature vector after the $\mathrm{SO}(S^2)$ convolution patch merge and layer normalization of stage ℓ. We use the feature vector notation from equation 19 and identify each two-dimensional geometric feature with a complex number. The gravity conditioning of x consists of two updates. First, for each $n = 1, 2$ and $a = 1, \ldots, a_n$,

$$x_a^{(n)}(p) \leftarrow x_a^{(n)}(p) + A_{\ell,n,a} G_n(p),$$

where $A_{\ell,n,a}$ is an intertwiner $\rho_n \to \rho_n$. This update supplies the geometric channels with a reference derived from gravity. Second, the directional features are read against their gravity reference to obtain invariant scalar measurements. Define

$$I(x, G)(p) := \operatorname{concat}_{\substack{n=1,2 \\ a=1,\ldots,a_n}} \begin{pmatrix} \mathrm{Re}\left(x_a^{(n)}(p)\overline{G_n(p)}\right) \\ \mathrm{Im}\left(x_a^{(n)}(p)\overline{G_n(p)}\right) \end{pmatrix}.$$

The scalar channels are updated via

$$x^{(0)}(p) \leftarrow x^{(0)}(p) + B_\ell G_0(p) + C_\ell I(x, G)(p) + b_\ell,$$

where $B_\ell \in \mathbb{R}^{a_0 \times 4}$ and $C_\ell \in \mathbb{R}^{a_0 \times 2(a_1+a_2)}$ are learned linear maps. All conditioning weights and biases are initialized to zero, so the conditioning initially leaves the features unchanged.

Let f_ℓ denote the features coming out of stage ℓ and $\widetilde{r}_\ell$ their convolutional scalar readout. We apply the pointwise residual update

$$r_\ell(p) \leftarrow \widetilde{r}_\ell(p) + \mathrm{MLP}_\ell\left([\widetilde{r}_\ell(p),\ G_0(p),\ I(f_\ell, G)(p),\ N(f_\ell)(p)]\right),$$

where $N(f_\ell)$ concatenates the squared norms of the directional feature copies. The MLP has one GELU hidden layer of twice the readout width and a zero-initialized output layer.