bispectrum: Selective G-Bispectra Made Practical

Johan Mathe
Atmo, Inc.
San Francisco, CA
johan@atmo.ai

Adele Myers Lantow
UC Santa Barbara
Santa Barbara, CA
adele@ucsb.edu

Simon Mataigne
UCLouvain
Louvain-la-Neuve, Belgium
simon.mataigne@uclouvain.be

Nina Miolane
UC Santa Barbara
Santa Barbara, CA
ninamiolane@ucsb.edu

geometric-intelligence/bispectrum

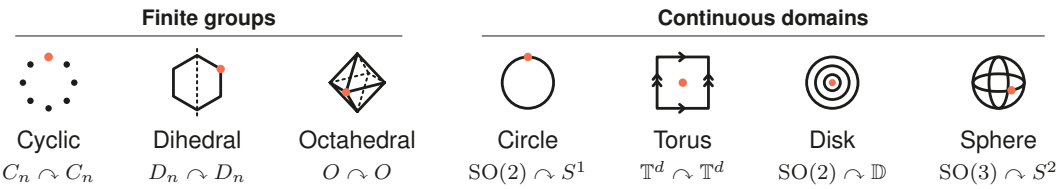

Figure 1: The seven group actions $G \curvearrowright X$ (group G acting on domain X) in bispectrum.

Abstract

Many machine learning tasks are invariant under the action of a group G of transformations. Signal classification can be invariant under translations, image classification under 2D rotations, and spherical-image classification under 3D rotations. One way to build this invariance into a neural network is through G-invariants. The G-bispectrum is a complete one. It determines the signal up to the group action and has served as a pooling layer in deep networks. Yet it is costly to compute, and each existing implementation covers a single group, severely limiting its adoption. We present bispectrum, an open-source, fully unit-tested PyTorch library that implements computationally efficient *selective* G-bispectra for seven group actions as differentiable modules. Selectivity reduces the number of bispectral coefficients from $O(|G|^2)$ to $O(|G|)$ for finite groups G, from $O(m^{5/2})$ to $O(m)$ for planar rotations on the disk with m harmonics, and, with the augmented selective spherical bispectrum we introduce, from $\Theta(L^3)$ to $\Theta(L^2)$ for spherical 3D rotations at band limit L. The library is invariant up to floating-point error and computes selective G-bispectra in sub-millisecond time on GPU. We evaluate them as pooling layers in G-equivariant networks on PatchCamelyon (C_8) and OrganMNIST3D (the octahedral group O) against five baselines. Across 46 comparisons, bispectral pooling remains competitive, matching all baselines on PCam and improving on max pooling by 5.0 pp on OrganMNIST3D. On Spherical MNIST, the augmented selective spherical bispectrum with an MLP head reaches 95.8% on rotated test digits, on par with a matched spherical CNN at $4\times$ lower latency and within 0.3 pp of a tensor-product network at $364\times$ lower latency. Selective G-bispectra are thus fast and competitive invariant layers that keep far more of the signal than standard pooling, for both finite and continuous groups, and we hope bispectrum will broaden their use to new applications.

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

1 Introduction

In many learning problems the label does not change when a group G of transformations acts on the input. A translated signal, a rotated image, or a rotated spherical image keeps its class. Equivariant neural networks [Cohen and Welling, 2016, Weiler and Cesa, 2019] enforce invariance by adding an invariant pooling layer to their equivariant feature maps. Existing choices (norm pooling, gate-based pooling, max/sum aggregation [Weiler and Cesa, 2019]) are simple but *incomplete*. They discard information that a complete group-invariant representation would preserve. For instance, max pooling over G keeps only the largest response across the group, so two feature maps that share their maximum but differ everywhere else are mapped to the same invariant descriptor even though no group element relates them.

The G-bispectrum [Tukey, 1953, Giannakis, 1989, Kakarala, 2012, Sanborn et al., 2023, Mataigne et al., 2024, Myers Lantow and Miolane, 2026], a complex-valued G-invariant descriptor, avoids this. It is *complete* (it preserves all signal information up to the group action), *invertible* (the signal can be recovered from it up to a group action), and *selective* (a small subset of its entries suffices for completeness on generic signals). Selectivity was first observed for translations by Giannakis [1989] and recently extended to finite groups and the disk [Mataigne et al., 2024, Myers Lantow and Miolane, 2026].

Prior implementations. Existing implementations and uses of bispectra are fragmented across works, each tied to one group and its own code. The selective G-bispectrum of Mataigne et al. [2024] covers signals defined over finite groups. The selective disk bispectrum of Myers Lantow and Miolane [2026] covers 2D images defined over the disk, with invariance with respect to planar rotations. The spherical bispectrum of Kakarala [2012] provides a representation of spherical images that is invariant to 3D rotations. Oreiller et al. [2020] use it as a locally rotation-invariant layer in 3D CNNs, keeping all $\Theta(L^3)$ coefficients up to a maximal degree L, and Chevalley et al. [2024] build a bispectral layer into a 3D U-Net for segmentation. The classical translation bispectrum [Tukey, 1953, Giannakis, 1989] is well-known in signal processing but lacks an open, GPU-ready implementation. Sanborn et al. [2023] learn the bispectrum of compact commutative groups from data with bispectral neural networks, a form that does not cover non-commutative groups. No single library offers these invariants across groups behind one tested, GPU-ready interface, so using a bispectrum on a new group or domain still means re-deriving and re-implementing it. `bispectrum` closes this gap.

Contributions. We present `bispectrum`, an open-source PyTorch library of *selective* G-bispectra. Our contributions are:

1. **Library.** We implement selective G-bispectra for *seven group actions* as differentiable modules. We provide an efficient implementation with sub-millisecond GPU forward passes and a fully unit-tested codebase (98.5% line coverage).
2. **Spherical invariant.** We introduce the augmented selective spherical bispectrum for 3D rotations acting on spherical images. At band limit L, it combines a selective subset of the bispectrum with a few Clebsch–Gordan power coefficients, reducing the number of coefficients from $\Theta(L^3)$ to $\Theta(L^2)$. Its per-degree Jacobian has full rank by construction, and a degree-by-degree solver recovers every test signal from it at $L{=}15$. We conjecture that it is generically complete.
3. **Evaluation.** We evaluate bispectral pooling in G-equivariant networks on 2D histopathology (C_8) and 3D organ (O) classification, against norm pooling, gated pooling, Fourier-ELU pooling, max pooling, and a data-augmented CNN. Across 46 comparisons it remains competitive, matching all baselines on PCam and improving on max pooling by 5.0 pp on OrganMNIST3D. On Spherical MNIST, the augmented selective spherical bispectrum with an MLP head is within 0.3 pp of matched learned-equivariant networks on rotated digits at $4\times$ to $364\times$ lower latency. We also find that a ReLU before the first learned projection costs 4–5 pp on OrganMNIST3D.

Overall, these results position selective G-bispectra as a practical, information-preserving alternative to the incomplete pooling layers of equivariant networks.

2 Mathematical Background

A group G *acts* on a domain X, for example rotations acting on the sphere, through a map $(g, x) \mapsto g \cdot x$. This action carries over to signals defined on that domain, $(f \circ g)(x) := f(g \cdot x)$. Two signals are in the same *orbit* if and only if they are related by some $g \in G$, like an image and a rotated copy of it. An *invariant* is a function that is constant on each orbit: it takes the same value on a signal and on all of its transformed copies.

Equivariant networks process signals while keeping track of the group action. A neural network layer Φ is *G-equivariant* if $\Phi(\pi_{\text{in}}(g)\, f) = \pi_{\text{out}}(g)\, \Phi(f)$ for all $g \in G$ [Cohen and Welling, 2016, Weiler and Cesa, 2019], where $\pi_{\text{in}}(g)$ and $\pi_{\text{out}}(g)$ denote how g acts on the layer's input and output feature maps. Transforming the input thus transforms the output in a predictable way. Stacking equivariant layers preserves equivariance, but any task requiring a prediction independent of the group action must eventually map equivariant features to invariant ones. This is the role of the invariant pooling layer, and the G-bispectrum is one choice for it.

The G-bispectrum is built from the Fourier transform on G, which relies on representations. A *representation* ρ of G assigns to each $g \in G$ a unitary matrix $\rho(g)$ in a way that respects composition, $\rho(gh) = \rho(g)\rho(h)$. Representations generalize the complex exponentials $e^{in\phi}$ of classical Fourier analysis, which represent planar rotations by 1×1 matrices. A representation is *irreducible* (an *irrep*) when it cannot be split into smaller ones, that is, when it has no nontrivial invariant subspace. Irreps play the role of frequencies, and the *dual* $\hat{G}$ is the set of equivalence classes of irreps.

Bispectrum on finite groups. The G-bispectrum [Kakarala, 1992, Mataigne et al., 2024] provides a complete invariant for signals on finite groups, usable as a pooling layer. Given a signal $f : G \to \mathbb{R}$ on a finite group G with unitary irreps $\{\rho_i\}_{i \in \hat{G}}$, the *$G$-Fourier transform* [Diaconis and Rockmore, 1990] is

$$\mathcal{F}(f)_{\rho_i} := \sum_{g \in G} f(g)\, \rho_i(g)^\dagger \;\in\; \mathbb{C}^{d_i \times d_i}, \tag{1}$$

where $d_i = \dim(\rho_i)$ and $\dagger$ denotes the conjugate transpose. The *full G-bispectrum* is

$$\beta(f)_{\rho_1,\rho_2} = \left[\mathcal{F}(f)_{\rho_1} \otimes \mathcal{F}(f)_{\rho_2}\right] C_{\rho_1,\rho_2} \left[\bigoplus_{\rho \in \rho_1 \otimes \rho_2} \mathcal{F}(f)_\rho^\dagger\right] C_{\rho_1,\rho_2}^\dagger, \tag{2}$$

where C_{ρ_1,ρ_2} is the unitary Clebsch–Gordan matrix that block-diagonalizes the tensor product $\rho_1 \otimes \rho_2$ into irreducible summands, and $\bigoplus$ forms the block-diagonal matrix with one block $\mathcal{F}(f)_\rho^\dagger$ for each irrep ρ in the decomposition of $\rho_1 \otimes \rho_2$, repeated by multiplicity. Each matrix $\beta(f)_{\rho_1,\rho_2}$ is G-invariant, and the full collection $\{\beta(f)_{\rho_1,\rho_2}\}_{(\rho_1,\rho_2) \in \hat{G}^2}$ is a complete G-invariant for generic signals (i.e., those with nonsingular Fourier coefficients) [Kakarala, 1992]. The full bispectrum has $O(|G|^2)$ scalar entries. Mataigne et al. [2024] show that a breadth-first search over the Kronecker product table of $\hat{G}$, which lists the irreps in each $\rho_i \otimes \rho_j$, selects $O(|G|)$ bispectral coefficients that remain complete. For example, the octahedral group drops from 576 to 172 scalars.

Bispectrum on the disk. The disk bispectrum [Zhao and Singer, 2014, Myers Lantow and Miolane, 2026] is an $\mathrm{SO}(2)$-invariant descriptor for planar images defined on the unit disk $\mathbb{D}$. A signal $f : \mathbb{D} \to \mathbb{R}$ is expanded in disk harmonics (eigenfunctions of the Laplacian on $\mathbb{D}$):

$$\psi_{nk}(r, \theta) = c_{nk}\, J_n(\lambda_{nk}\, r)\, e^{in\theta}, \tag{3}$$

where J_n is the Bessel function of order n, λ_{nk} its k-th positive root, and c_{nk} a normalization constant. The disk harmonic (DH) coefficients $a_{n,k}$ are equivariant to planar rotations: if $f' = f \circ R_\phi$ for the rotation R_ϕ by angle ϕ, then $a_{n,k}^{f'} = e^{in\phi}\, a_{n,k}^{f}$. For a band limit yielding m coefficients with maximum angular frequency N_m, the *full disk bispectrum* [Zhao and Singer, 2014] is

$$b_{j_1,j_2,k_3} = a_{n_{j_1},k_{j_1}}\, a_{n_{j_2},k_{j_2}}\, a^*_{n_{j_1}+n_{j_2},\,k_3}, \tag{4}$$

where $j \mapsto (n_j, k_j)$ enumerates the m coefficients. It has $O(m^3/N_m)$ coefficients, that is $O(m^{5/2})$, since a band limit on the Bessel roots gives $N_m = \Theta(\sqrt{m})$. Myers Lantow and Miolane [2026] introduce the *selective disk bispectrum*, which keeps only the products $a_{0,1}^2\, a_{0,k}^*$ and $a_{1,1}\, a_{n,1}\, a_{n+1,k}^*$, one per coefficient and so $O(m)$ in total, while remaining a complete $\mathrm{SO}(2)$-invariant (under the genericity condition $a_{n,1} \neq 0$ for all angular orders n).

Bispectrum on the sphere. The spherical bispectrum [Kakarala, 2012] is an $\mathrm{SO}(3)$-invariant descriptor for band-limited signals on S^2. A real signal $f : S^2 \to \mathbb{R}$ at band limit L has spherical harmonic coefficients a_ℓ^m with coefficient vectors $\mathbf{F}_\ell = (a_\ell^{-\ell}, \ldots, a_\ell^{\ell})^\top \in \mathbb{C}^{2\ell+1}$, transforming under $g \in \mathrm{SO}(3)$ as $\mathbf{F}_\ell \mapsto D^\ell(g)\,\mathbf{F}_\ell$, where D^ℓ is the $(2\ell{+}1)$-dimensional Wigner D-matrix. For each admissible triple (ℓ_1, ℓ_2, ℓ) satisfying the triangle inequality $|\ell_1 - \ell_2| \le \ell \le \ell_1 + \ell_2$, the *bispectral coefficient* is

$$\beta_{\ell_1,\ell_2,\ell} = \sum_{\substack{m_1,\, m_2,\, m \\ m_1+m_2=m}} C^{\ell,\,m}_{\ell_1,\,m_1;\,\ell_2,\,m_2}\; a_{\ell_1}^{m_1}\, a_{\ell_2}^{m_2}\, \overline{a_\ell^{m}}, \tag{5}$$

where $C^{\ell,\,m}_{\ell_1,\,m_1;\,\ell_2,\,m_2}$ denotes the Clebsch–Gordan coefficient $\langle \ell_1, m_1;\, \ell_2, m_2 \mid \ell, m \rangle$. The full set has $\Theta(L^3)$ scalars. On homogeneous spaces such as S^2, where G can move any point to any other, the Fourier coefficients are rank-deficient (vectors rather than full-rank matrices), so the completeness theorem of Kakarala [2012], which requires nonsingular matrix coefficients, does not directly apply [Kondor, 2007]. In Section 5 we introduce a selective construction that reduces the scalar bispectrum to $\Theta(L^2)$ entries and test its completeness empirically by signal reconstruction.

Strictly, the G-bispectrum [Kakarala, 1992, Mataigne et al., 2024] is defined for signals on the group G itself (Equation (2)), whereas the disk and sphere bispectra (Equations (4) and (5)) are bispectra of signals on a domain acted on by G. For brevity, we extend the term *G-bispectrum* to all three.

3 Related Work

Equivariant network libraries. The `escnn` [Cesa et al., 2022] and `e3nn` [Geiger and Smidt, 2022] libraries provide general-purpose equivariant convolution layers for 2D and 3D groups respectively, and `lie-nn` [Batatia et al., 2023] provides general Lie-group representation primitives (irreducible representations, Clebsch–Gordan decompositions, tensor products) on top of which custom equivariant layers can be built. These libraries implement equivariant *feature maps* but leave the choice of invariant pooling to the user (typically norm or gate). Our modules provide the invariant map and work with any of these libraries.

Bispectral approaches in neural networks. Sanborn et al. [2023] introduced bispectral neural networks (BNNs), which learn a Fourier transform and its full bispectrum from data for compact commutative groups (2D cyclic translations, planar rotations). Oreiller et al. [2022] build a locally rotation-invariant layer for a 2D U-Net from the bispectrum of local circular-harmonic responses, over all frequency pairs up to a maximal degree. Oreiller et al. [2020] do the same in 3D with spherical harmonics, and Chevalley et al. [2024] build such a layer into a 3D U-Net for segmentation. Both Oreiller layers keep every bispectral coefficient up to symmetry, and each of these implementations is tied to a specific architecture. Our library covers seven group/domain pairs, defaults to the $O(|G|)$ selective reduction [Mataigne et al., 2024], and exposes each invariant as a standalone PyTorch module. Outside the equivariant-network setting, `PyBispectra` [Binns et al., 2025] computes bispectrum and bicoherence statistics of one-dimensional time-series signals for electrophysiology applications such as phase-amplitude coupling and time-delay estimation. The underlying 1D bispectrum is translation-invariant by construction, but the toolbox is not designed as a neural-network pooling layer.

Bispectra on homogeneous and non-homogeneous spaces. The selective G-bispectrum construction [Mataigne et al., 2024] reduces the cost of the G-bispectrum from $O(|G|^2)$ to $O(|G|)$ scalars on finite groups. Myers Lantow and Miolane [2026] extend the selective construction to the disk bispectrum of Zhao and Singer [2014], where $\mathrm{SO}(2)$ acts on a non-homogeneous space, since rotations preserve the radius. Kakarala [2012] treats the $\mathrm{SO}(3)$-on-$\mathrm{SO}(3)$ bispectrum and the closely related sphere case via Wigner-D irreps. The matrix (vector) bispectrum on S^2 is the homogeneous-space form on which our scalar contractions and CG power augmentation are built.

Scattering transforms. The wavelet scattering transform [Mallat, 2012] provides translation-invariant representations via iterated modulus operations. McEwen et al. [2022] extend it to the sphere. Unlike the G-bispectrum, scattering representations are stable but generally incomplete. They discard inter-scale phase information.

Table 1: **Selective bispectra are complete invariants with as few coefficients as incomplete ones.** The selective G-bispectrum [Mataigne et al., 2024] and the selective disk bispectrum [Myers Lantow and Miolane, 2026] are complete with $O(|G|)$ and $O(m)$ coefficients, for m disk harmonics. Our augmented selective spherical bispectrum has $\Theta(L^2)$ coefficients and is complete in all our tests (Sec. 5), without a proof. †Completeness holds for almost every signal, all but a negligible set.

Nonlinearity	Complete?	Coefficients	Forward cost	Key limitation
Norm	No	$O(\|G\|)$	$O(\|G\|)$	Discards phase
Gate	No	$O(\|G\|)$	$O(\|G\|)$	Scalar bottleneck
Fourier pointwise	No‡	$O(\|G\|)$	$O(\|G\| \log \|G\|)$	Aliasing
Full bispectrum (finite G)	Yes	$O(\|G\|^2)$	$O(\|G\|^3)$	Quadratic cost
Full bispectrum (SO(2)/disk)	Yes	$O(m^{5/2})$	$O(m^{5/2})$	Superlinear cost
Full bispectrum (SO(3)/S^2)	Yes	$\Theta(L^3)$	$O(L^5)$	Cubic cost
Selective bispectrum (finite G)	Yes	$O(\|G\|)$	$O(\|G\|^2)$	Genericity assumption†
Selective bispectrum (SO(2)/disk)	Yes	$O(m)$	$O(m \log m)$§	Genericity assumption†
Augmented selective spherical bispectrum (SO(3)/S^2, ours)	Empirical	$\Theta(L^2)$	$O(L^4)$	No completeness proof

‡The pointwise nonlinearity is equivariant, not invariant, so an invariant still needs pooling over G (max pooling in our experiments), which is incomplete. Aliasing also breaks exact equivariance. §With the fast disk-harmonic transform of Marshall et al. [2023]. Our implementation uses an exact dense transform, $O(m^2)$.

Table 2: Seven group/domain pairs in the `bispectrum` library. `SO2onS1` is the continuous-n limit of `CnonCn`.

Module	Group / Domain	Output mode	Reference
`CnonCn`	C_n on C_n	selective + full	[Giannakis, 1989]
`SO2onS1`	SO(2) on S^1	selective + full	[Mataigne et al., 2024]
`TorusOnTorus`	$\mathbb{T}^d$ on $\mathbb{T}^d$	selective + full	[Mataigne et al., 2024]
`DnonDn`	D_n on D_n	selective	[Mataigne et al., 2024]
`SO2onDisk`	SO(2) on disk	selective	[Myers Lantow and Miolane, 2026]
`SO3onS2`	SO(3) on S^2	selective + full	[Kakarala, 2012], this paper
`OctaonOcta`	chiral octahedral O	selective	[Mataigne et al., 2024]

Of the layers compared in Table 1, only the selective bispectrum (Section 2) is complete on generic signals with $O(|G|)$ coefficients.

4 The `bispectrum` Library

Design and implementation. Every G/domain pair is a `torch.nn.Module` named `{Group}on{Domain}` (e.g., `SO3onS2`). Table 2 lists all seven modules. Each exposes a uniform API: `forward(f)` returns the selective invariants, `fourier(f)` the group Fourier coefficients, and `invert(beta)` the reconstructed signal up to group-action indeterminacy (where available). Modules default to $O(|G|)$ selective coefficients (`selective=False` restores the full $O(|G|^2)$ set where available, Table 2), precompute CG matrices / DFT kernels / Bessel roots as non-learnable buffers, and depend only on PyTorch [Paszke et al., 2019], NumPy [Harris et al., 2020], and `torch_harmonics` [Bonev et al., 2023] (for `SO3onS2`). Test suite line coverage is 98.5%. A minimal usage example is in Appendix E.

Profiling. We profile the library on a single NVIDIA H100 80 GB GPU (PyTorch 2.11, `bispectrum` 0.4.1, batch = 16 for the forward pass and 4 for inversion, median wall-clock via `torch.utils.benchmark`). Results are in Table 3. With autograd enabled, as in training, the selective forward pass is sub-millisecond for all six modules, from 0.08 ms ($\mathbb{T}^2$) to 0.68 ms (octahedral group). `SO3onS2` at L=16 takes 0.63 ms, or 0.20 ms in inference mode, where it replays a CUDA graph. GPU throughput exceeds 10^7 samples/s for C_n and $\mathbb{T}^2$ (Appendix B). The same configurations stay sub-millisecond on an A100, a consumer RTX 4090, and Apple silicon (MPS). On CPU, all but SO(3) on S^2 do (Appendix B.4). Six modules are presented. The seventh, `SO2onS1`, subclasses `CnonCn` with $n = L+1$ and has identical runtime to the corresponding `CnonCn` row at matched $|G|$. Coefficient-count scaling and additional plots are in Appendix B (Figure 5).

Table 3: **The selective forward pass is sub-millisecond for every module, with up to $512\times$ fewer coefficients than the full bispectrum.** Median wall-clock, NVIDIA H100, PyTorch 2.11, batch = 16, with autograd enabled (Sel.) and in inference mode (Inf.), where `SO3onS2` replays a CUDA graph. "–": not implemented.

Module	G	$\lvert G\rvert/L$	Coefs		Fwd (ms)		
			Sel.	Full	Sel.	Inf.	Full
`CnonCn`	C_{128}	128	128	8,256	0.14	0.13	8.81
`TorusOnTorus`	$C_{32}\times C_{32}$	1,024	1,024	524,800	0.08	0.07	0.31
`DnonDn`	D_{32}	64	245	–	0.54	0.48	–
`SO2onDisk`	$\mathrm{SO}(2)$	L=16	76	–	0.22	0.21	–
`SO3onS2`	$\mathrm{SO}(3)$	L=16	426	1,341	0.63	0.20	53.61
`OctaonOcta`	O	24	172	–	0.68	0.61	–

5 The Augmented Selective Spherical Bispectrum

We construct the augmented selective spherical bispectrum Φ_{sel}, an $\mathrm{SO}(3)$-invariant of size $\Theta(L^2)$ on band-limited spherical signals, and probe its completeness empirically.

Setup and construction. As in Section 2, a real signal $f : S^2 \to \mathbb{R}$ band-limited at degree L has coefficient vectors $\mathbf{F}_\ell \in \mathbb{C}^{2\ell+1}$, $\ell = 0, \dots, L$, that transform as $\mathbf{F}_\ell \mapsto D^\ell(g)\,\mathbf{F}_\ell$ under $g \in \mathrm{SO}(3)$. The reality condition leaves $(L{+}1)^2$ real degrees of freedom (Appendix A.1), and the full scalar bispectrum (5) has $\Theta(L^3)$ entries. The orbit space $\mathbb{R}^{(L+1)^2}/\mathrm{SO}(3)$ has dimension $(L{+}1)^2 - 3$. By a rank argument, any smooth complete $\mathrm{SO}(3)$-invariant must therefore have at least $\Omega(L^2)$ components (Proposition 7).

The augmented selective spherical bispectrum. We define $\Phi_{\mathrm{sel}}(f)$ as the output of Algorithm 1. It has two parts: a *bispectral part* $\mathcal{S}_\beta$, a selected subset of the coefficients (5), and a *CG-power part* $\mathcal{S}_P$ of scalars $P_{\ell_1,\ell_2,\ell} := \|(\mathbf{F}_{\ell_1} \otimes \mathbf{F}_{\ell_2})|_\ell\|^2$, the squared norms of Clebsch–Gordan projections (Definition 4). Both are built degree by degree. Once the lower degrees are known, an entry $\beta_{a,b,\ell}$ with $a, b < \ell$ is linear in $\mathbf{F}_\ell$, so the entries of degree ℓ form a system in the $2\ell{+}1$ real unknowns of $\mathbf{F}_\ell$. Degrees $\ell \le 4$ contribute 22 bispectral entries and degrees $5 \le \ell \le 7$ the fixed lists $\mathcal{T}_\ell^{\mathrm{small}}$ (Table 4). Each degree $\ell \ge 8$ contributes up to $2\ell{+}1$ entries $\beta_{a,b,\ell}$ with $1 \le a \le b < \ell$, one per unordered triple, since on real signals all orderings of a triple are proportional (Remark 17). At $\ell \in \{8, 9\}$, where only 16 such triples are nonzero, the remainder is filled by the power entry $\beta_{0,\ell,\ell} = a_0^0 \|\mathbf{F}_\ell\|^2$ and the self-couplings $\beta_{\ell,\ell,\ell'}$, which are nonlinear in $\mathbf{F}_\ell$. Every degree $\ell \ge 3$ also receives the self-couplings with even $\ell' \le \ell$ and a fixed family of at most seven CG-power scalars. Odd ℓ' is skipped because an entry with odd $\ell_1{+}\ell_2{+}\ell$ and a repeated index vanishes identically on real signals (Proposition 13). With these included, the Jacobian of the degree-ℓ entries with respect to the $2\ell{+}1$ real unknowns of $\mathbf{F}_\ell$ has full rank $2\ell{+}1$ at random signals, so $\mathbf{F}_\ell$ is locally determined by the lower degrees. The test suite checks this for $4 \le \ell \le 20$ and a verification script for $\ell \le 100$. Full rank is a local property and does not exclude the mirror image $T_R f$ of f, which is generally not a rotation of f. Entries with even $\ell_1{+}\ell_2{+}\ell$ and all CG powers take the same value on f and $T_R f$. Entries with odd $\ell_1{+}\ell_2{+}\ell$ change sign, but they vanish identically up to degree 3, so the first that tells f from $T_R f$ is $\beta_{2,3,4}$, which Φ_{sel} includes (Appendix A.4). The total size is $\Theta(L^2)$ (Proposition 10): 380 (resp. 426) real scalars at $L = 15$ (resp. 16), against 1,124 (resp. 1,341) entries of the full bispectrum (Table 5).

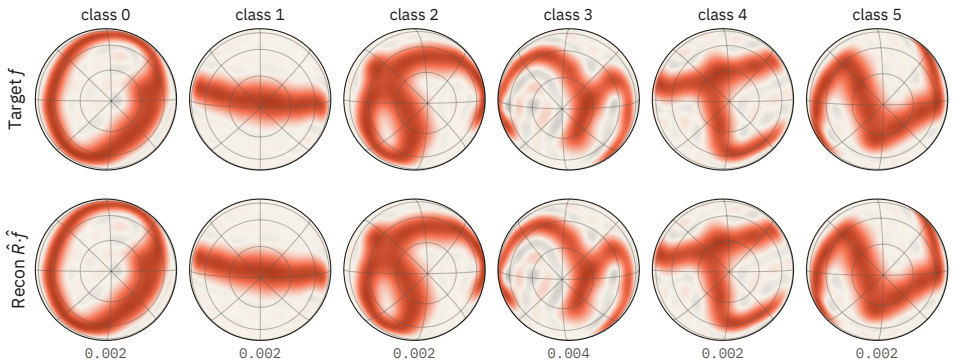

Figure 2: **Spherical signals recovered from their 380 augmented selective spherical bispectrum coefficients alone match the targets up to rotation.** Top: Spherical MNIST digits, band-limited to $L{=}15$. Bottom: reconstructions by a degree-by-degree solver from random starts, rotated onto the target (relative image error below each). All 16 targets (8 digits, each in 2 orientations) are recovered: invariants match to better than 10^{-13} and images to a median 1.8×10^{-3}, the error left by interpolating the alignment rotation on the grid. Gradient descent from noise recovers 9 of 16 (Appendix D.7).

Algorithm 1 Index set of Φ_{sel}, degree by degree. The low-degree and CG-power details are in Algorithm 2 (Appendix A.3).

Input: band limit L
$(\mathcal{S}_\beta, \mathcal{S}_P) \leftarrow$ entries of degree $\leq \min(L, 4)$ {22 bispectral, 10 CG-power at $L \geq 4$}
for $\ell = 5, \ldots, L$ **do**
  $\mathcal{B}_\ell \leftarrow \mathcal{T}_\ell^{\text{small}}$ if $\ell \leq 7$, else (a, b, ℓ) for $1 \leq a \leq b < \ell$, $a{+}b \geq \ell$, by $a{+}b$ then a
  append to $\mathcal{S}_\beta$ the first $2\ell{+}1$ entries of $\mathcal{B}_\ell$, $(0, \ell, \ell)$, (ℓ, ℓ, ℓ') for even $\ell' < \ell$ descending, that do not vanish on real signals
  append (ℓ, ℓ, ℓ') to $\mathcal{S}_\beta$ for even $2 \leq \ell' \leq \ell$, if not present
  append the fixed CG-power triples of degree ℓ to $\mathcal{S}_P$ {5 for even ℓ, 7 for odd}
end for
return $\{\beta_t(f)\}_{t \in \mathcal{S}_\beta} \cup \{P_t(f)\}_{t \in \mathcal{S}_P}$

Empirical evidence of completeness. We call Φ_{sel} a *complete* $\mathrm{SO}(3)$-invariant if $\Phi_{\text{sel}}(f) = \Phi_{\text{sel}}(g)$ implies that f and g lie in the same orbit. Completeness can be probed empirically by attempting to invert Φ_{sel}: if it is complete, any signal $\hat{f}$ matching $\Phi_{\text{sel}}(f)$ must agree with f up to a global rotation, so consistent recovery of the input is evidence that no orbit information is lost. We run this test at the classifier's band limit $L{=}15$, where Φ_{sel} has 380 coefficients, on 8 Spherical MNIST digits (each in its original and one random orientation) and 16 random band-limited signals, sampled on a $128{\times}256$ grid. A degree-by-degree solver works on the 256 real spherical-harmonic coefficients: Levenberg–Marquardt with the exact Jacobian fits degrees 0–4 from random starts, then adds one degree at a time and refits all degrees so far. Each solution is rotated onto its target before the relative image error is measured (Appendix D.7).

Results. All 32 targets are recovered: the augmented selective spherical bispectrum matches to better than 10^{-13}, the full bispectrum to 10^{-12}, and the aligned images to 0.0014–0.0035 (digits) and 0.0056–0.0070 (random signals), the error left by interpolating the alignment rotation on the grid (Figure 2). No run found two signals in different orbits with equal invariants. Gradient descent on the image from noise recovers 9 of 16 digits and none of the random signals, on which it stalls at a 7.5% median invariant error, so the failures come from the optimizer. This does not prove completeness. We conjecture that Φ_{sel} is generically complete, that is, complete outside a measure-zero set of signals.

6 Experiments

We evaluate the selective G-bispectrum as the invariant pooling layer of G-equivariant networks on two benchmarks, 2D histopathology under C_8 and 3D organ classification under the octahedral

group O (Section 6.1), and the augmented selective spherical bispectrum on spherical digit classification (Section 6.2).

6.1 G-bispectral pooling in G-CNNs

Setup. PatchCamelyon [Veeling et al., 2018] (PCam) has 327K 96×96 histopathology patches with binary metastasis labels. Every model uses a C_8-equivariant DenseNet [Huang et al., 2017] backbone following Veeling et al. [2018], and we report test AUC. OrganMNIST3D [Yang et al., 2023] has 1,742 abdominal CT volumes (28^3 voxels, 11 classes, 971 for training). Every model uses an O-equivariant 3D ResNet with exact voxel-permutation kernels, and we report test accuracy. Within a dataset the equivariant models differ only in the terminal invariant map over the group dimension of the feature maps: a maximum (Max pool, and the NormReLU and Gated [Weiler and Cesa, 2019] and Fourier-ELU [Cesa et al., 2022] backbones, named after their nonlinearity, Appendix C), the ℓ_2 norm (Norm pool), or the selective G-bispectrum (128 coefficients for `CnonCn` with n=8 and 172 for `OctaonOcta`). The bispectral head applies only fixed injective maps (real and imaginary parts, signed logarithm) before a learned 1×1 projection and a ReLU. A non-equivariant CNN trained with augmentation over G (Aug. CNN) is the baseline without built-in symmetry. Equivariant models are trained on canonical data only and tested on the test set transformed by every element of G (the *rotated test*). Every model is trained with AdamW [Loshchilov and Hutter, 2019] for a fixed number of epochs that ends with a cosine cycle [Loshchilov and Hutter, 2017], without early stopping, and evaluated at its best-validation checkpoint. We vary the training-set size (100 to 262K examples on PCam and 50 to 971 on OrganMNIST3D) and the width (∼30K to 1.6M parameters on PCam and 0.37M to 6.3M for the equivariant models on OrganMNIST3D), over 3 to 10 seeds (Appendices C.1 and D).

Results. Figure 3 shows the results. Across 46 seed-level comparisons (Welch t-tests on the rotated test against every baseline at every training-set size, and at every width on OrganMNIST3D, detailed in Appendix C.1), bispectral pooling is never significantly worse than any baseline. On OrganMNIST3D it improves on max pooling on the same backbone by 5.0 pp at full data (78.54% vs. 73.56%, $p < 10^{-3}$). Its smallest model (0.46M parameters) beats max pooling's largest (6.0M) by 2.6 pp, and with 500 training examples it beats max pooling trained on all 971 by 3.6 pp. It is 3.6 to 16 pp above norm pooling and the augmented 3D CNN at every training-set size from 100 and at every width. The advantage over max pooling vanishes at the smallest training sets ($n \leq 250$: −0.3 to +1.6 pp, not significant) and is not significant at the largest width (+1.6 pp, p=0.10). On PCam all methods are on par: no model is significantly better than bispectral pooling at any training-set size. With 3 seeds per arm, these tests have 80% power only for gaps of about 0.025 AUC. All equivariant models score the same on the canonical and rotated tests of OrganMNIST3D, where the 24 rotations permute voxels exactly. Placing the ReLU before the projection instead zeroes every negative coefficient and costs 4–5 pp on OrganMNIST3D (Appendix C.3).

6.2 Spherical MNIST: $\mathrm{SO}(3)$ invariance on the sphere

The pooling experiments cover finite groups. For a continuous group, we test `SO3onS2` (the augmented selective spherical bispectrum, Section 5) on Spherical MNIST [Cohen et al., 2018], the standard benchmark for rotation invariance on S^2.

Setup. MNIST digits are stereographically projected onto a 64×128 equiangular grid. We compute Φ_{sel} at L=15 ($N = 380$ real invariants, stored as complex numbers by the library), apply sign-preserving $\operatorname{sign}(x)\log(1+|x|)$ to real and imaginary parts ($2N = 760$ real channels, keeping both channels for simplicity even though one channel per bispectral entry is identically zero by parity), and feed a 3-layer MLP (230K parameters). We compare against two learned $\mathrm{SO}(3)$-equivariant networks run in the same pipeline and matched in size: an S2CNN [Cohen et al., 2018] with BatchNorm (155K parameters) and a Clebsch–Gordan tensor-product network built with `e3nn` [Geiger and Smidt, 2022] (386K). Controls are a power-spectrum MLP (11K parameters, equally invariant but *incomplete*, discarding the relative phases between degrees) and a non-invariant CNN on the equirectangular image (185K). Following Cohen et al. [2018], we evaluate over 5 seeds under the protocols NR/NR, R/R, and NR/R (train unrotated, test rotated), which tests invariance to unseen rotations. The bispectrum and the two equivariant networks are each tuned on validation accuracy at full data, and the selected settings are reused at every training-set size.

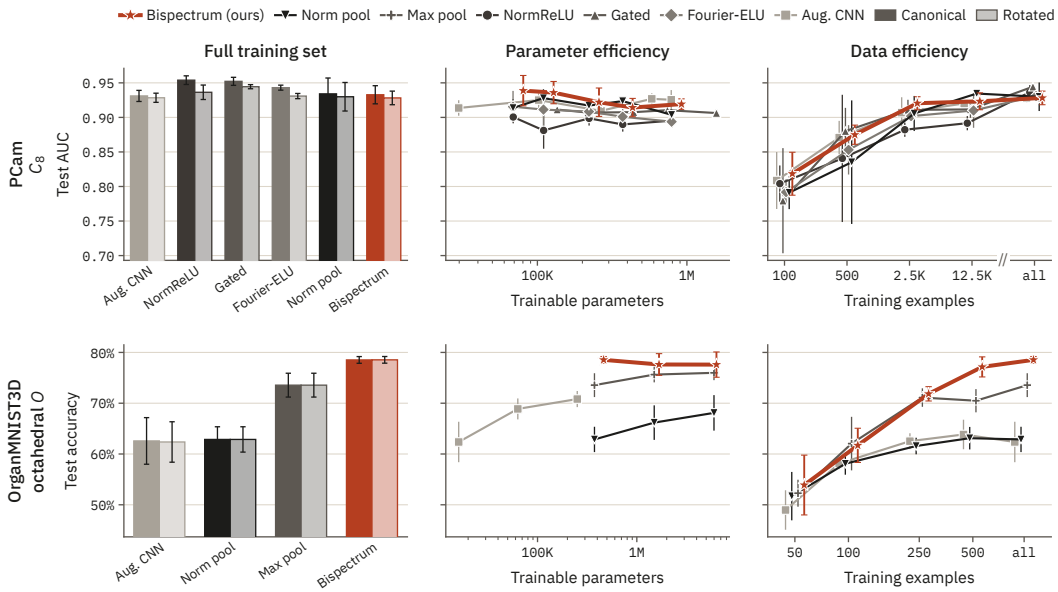

Figure 3: **Selective G-bispectral pooling matches baselines on PCam ($G{=}C_8$) and beats max pooling on OrganMNIST3D ($G{=}O$) at full data.** Top row, PCam (test AUC). Bottom row, OrganMNIST3D (test accuracy). Left column, full training set, with canonical (dark) and rotated (light) test bars. All models are on par on PCam, and on OrganMNIST3D bispectral pooling beats max pooling on the same backbone by 5.0 pp at the smallest width. Middle column, rotated-test metric against trainable parameters. On OrganMNIST3D bispectral pooling stays above max pooling at every width (not significantly at the largest), and its smallest model beats max pooling's largest. Right column, rotated-test metric against training-set size. On PCam no model is significantly better than bispectral pooling at any size. On OrganMNIST3D its lead over max pooling appears from 500 examples, where it already beats max pooling trained on all 971. The rotated test averages the metric over all $|G|$ rotations of the test set. Equivariant models are trained on canonical data only, and Aug. CNN is trained with augmentation over G. Mean $\pm$ std over 3–10 seeds, with details in Appendix C.1.

Results. Figure 4 reports the results. On the rotated test, Bispectrum + MLP reaches 95.8%, on par with the matched S2CNN (96.1%, $p{=}0.14$) and 0.25 pp below the tensor-product network ($p{=}0.006$), at $4\times$ and $364\times$ lower latency. Like the tensor-product network it loses nothing from NR/NR to NR/R, whereas S2CNN, which leads on NR/NR (97.1%) and R/R (96.6%), loses about 1 pp. The planar CNN collapses from 99.1% to 13.4%. The power spectrum is equally invariant but reaches only 77.7%, and scaling its MLP $48\times$ to 550K parameters lifts it only to 80.1%, below the bispectrum with 25K parameters (94.1%, Appendix D.5), so more capacity does not close the gap. With less data, the learned spherical CNN is more sample-efficient: S2CNN leads at every training-set size below the full set (by 13.0 pp at $n{=}100$ and 1.3 pp at 12.5K, Holm $p < 10^{-3}$), while Bispectrum + MLP beats the tensor-product network up to $n{=}2{,}500$ (by 17.3 pp at $n{=}100$) and the power spectrum at every size.

7 Discussion and Conclusion

We have presented `bispectrum`, a PyTorch library that makes selective G-bispectral invariants practical, covering seven group actions through one `nn.Module` interface. We contribute the augmented selective spherical bispectrum, a $\Theta(L^2)$ $\mathrm{SO}(3)$ invariant on S^2 (Section 5), which a degree-by-degree solver inverts on every signal we tested. We conjecture that it is generically complete. The selective forward pass takes under a millisecond on GPU for every module at batch 16. As a pooling layer in G-CNNs, the selective G-bispectrum is never significantly worse than an augmented CNN, than norm or max pooling, or than the NormReLU, Gated, and Fourier-ELU backbones. It is on par on PCam and improves on max pooling by 5.0 pp on OrganMNIST3D. On Spherical MNIST, the augmented selective spherical bispectrum with an MLP head is within 0.3 pp of matched learned-

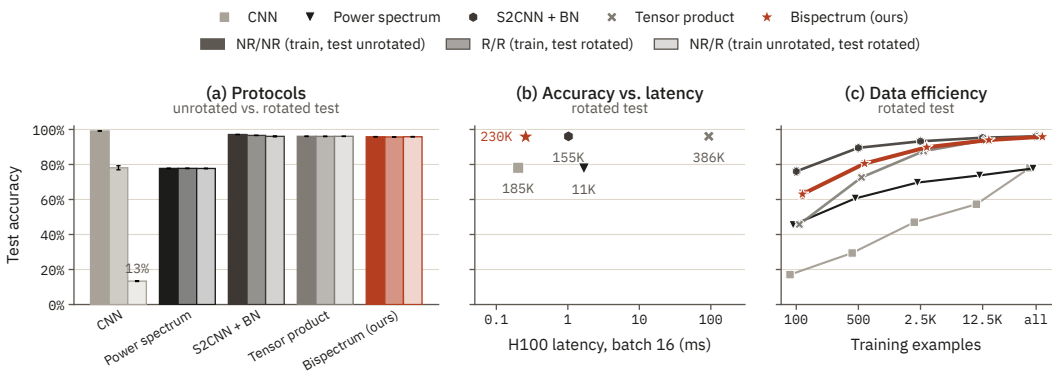

Figure 4: **The augmented selective spherical bispectrum is within 0.3 pp of equivariant networks on Spherical MNIST at $4\times$ to $364\times$ lower latency.** (a) Accuracy under the three train/test protocols (NR: unrotated, R: rotated). The CNN drops to 13% on unseen rotations, while the invariant and equivariant models do not. (b) Rotated-test accuracy against forward latency (spherical grid in, logits out, float32, batch 16, H100, median), with labels giving trainable parameters. (c) Rotated-test accuracy against training-set size. Bispectrum + MLP trails S2CNN + BatchNorm below the full set and beats the tensor-product net up to n=2,500. Mean $\pm$ std over 5 seeds, with settings in Table 11.

equivariant networks on rotated digits, at $4\times$ to $364\times$ lower latency. We speculate that the gain grows with how much information incomplete pooling discards. Per channel, norm pooling keeps a single value, the total energy, and max pooling keeps 1 of $|G|$ values, so both discard more for O ($|G|$=24) than for C_8 ($|G|$=8). Within the pooling head, a ReLU before the first learned projection costs 4–5 pp on OrganMNIST3D (Appendix C.3).

When not to use the bispectrum. Each module is invariant to a *specific* group only. Data augmentation or flexible-symmetry equivariant networks fit better when the relevant symmetry is multi-group or unknown. On the sphere, a fixed invariant followed by an MLP is less sample-efficient than a learned spherical CNN.

Limitations and future work. The degree-by-degree inversion of the augmented selective spherical bispectrum is in our experiment code and not yet in the library, and a formal completeness proof, a characterization of the signals on which completeness fails, and numerical-conditioning analysis at large band limits remain open. On S^2 we compare against one tensor-product network, not against MACE [Batatia et al., 2022] or wavelet scattering, and use the invariant with an MLP head. Bispectral pooling at the end of an equivariant spherical backbone is future work.

Acknowledgments and Disclosure of Funding

A. M. L. and N. M. acknowledge partial support from NSF grant 2313150, NSF CAREER grant 2240158, and the Noyce Foundation. A. M. L. also acknowledges support from an NSF Graduate Research Fellowship (GRFP) and NSF Grant No. DGE-2125644. S. M. is a Research Fellow of the Fonds de la Recherche Scientifique (FNRS). J. M. thanks his Atmo colleagues Giorgio Sarro, Duncan Axisa, Yves Martin, Alexander Sviridenko, Sander Henning, and Deepika Misra, as well as Creon Levit, for their feedback. The authors declare no competing interests.

References

Ilyes Batatia, Dávid Péter Kovács, Gregor N. C. Simm, Christoph Ortner, and Gábor Csányi. MACE: Higher order equivariant message passing neural networks for fast and accurate force fields. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pages 11423–11436, 2022. doi: 10.52202/068431-0830. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/4a36c3c51af11ed9f34615b81edb5bbc-Abstract-Conference.html.

Ilyes Batatia, Mario Geiger, Jose Munoz, Tess Smidt, Lior Silberman, and Christoph Ortner. A general framework for equivariant neural networks on reductive Lie groups. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, pages 55260–55284, 2023. doi: 10.52202/075280-2412.

Tamir Bendory, Dan Edidin, Josh Katz, and Shay Kreymer. Orbit recovery for spherical functions. *arXiv preprint arXiv:2508.02674*, 2025. URL https://arxiv.org/abs/2508.02674.

Thomas S. Binns, Franziska Pellegrini, Tin Jurhar, Tien D. Nguyen, Richard M. Köhler, and Stefan Haufe. PyBispectra: A toolbox for advanced electrophysiological signal processing using the bispectrum. *Journal of Open Source Software*, 10(112):8504, 2025. doi: 10.21105/joss.08504. URL https://doi.org/10.21105/joss.08504.

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 *International Conference on Machine Learning (ICML)*, volume 202 of *Proceedings of Machine Learning Research*, pages 2806–2823. PMLR, 2023. URL https://proceedings.mlr.press/v202/bonev23a.html.

Gabriele Cesa, Leon Lang, and Maurice Weiler. A program to build E(N)-equivariant steerable CNNs. In *International Conference on Learning Representations (ICLR)*, 2022. URL https://openreview.net/forum?id=WE4qe9xlnQw.

Arthur Chevalley, Valentin Oreiller, Julien Fageot, John O. Prior, Vincent Andrearczyk, and Adrien Depeursinge. A bispectral 3D U-Net for rotation robustness in medical segmentation. In *Topology- and Graph-Informed Imaging Informatics (TGI3, MICCAI Workshop)*, volume 15239 of *Lecture Notes in Computer Science*, pages 43–54. Springer Nature Switzerland, 2024. doi: 10.1007/978-3-031-73967-5_5. URL https://doi.org/10.1007/978-3-031-73967-5_5.

Taco S. Cohen and Max Welling. Group equivariant convolutional networks. In *International Conference on Machine Learning (ICML)*, volume 48 of *Proceedings of Machine Learning Research*, pages 2990–2999. PMLR, 2016. URL https://proceedings.mlr.press/v48/cohenc16.html.

Taco S. Cohen, Mario Geiger, Jonas Köhler, and Max Welling. Spherical CNNs. In *International Conference on Learning Representations (ICLR)*, 2018.

Persi Diaconis and Daniel Rockmore. Efficient computation of the Fourier transform on finite groups. *Journal of the American Mathematical Society*, 3(2):297–332, April 1990. doi: 10.1090/S0894-0347-1990-1030655-4. URL https://www.ams.org/journals/jams/1990-03-02/S0894-0347-1990-1030655-4/.

Dan Edidin and Matthew Satriano. Orbit recovery for band-limited functions. *SIAM Journal on Applied Algebra and Geometry*, 8(3):733–755, 2024. doi: 10.1137/23M1577808. URL https://arxiv.org/abs/2306.00155.

Babak Ehteshami Bejnordi, Mitko Veta, Paul Johannes van Diest, Bram van Ginneken, Nico Karssemeijer, Geert Litjens, Jeroen A. W. M. van der Laak, and the CAMELYON16 Consortium. Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer. *JAMA*, 318(22):2199–2210, 2017. doi: 10.1001/jama.2017.14585.

Mario Geiger and Tess Smidt. e3nn: Euclidean neural networks. *arXiv preprint arXiv:2207.09453*, 2022. URL https://arxiv.org/abs/2207.09453.

Georgios B. Giannakis. Signal reconstruction from multiple correlations: frequency- and time-domain approaches. *Journal of the Optical Society of America A*, 6(5):682–697, May 1989. doi: 10.1364/JOSAA.6.000682. URL https://opg.optica.org/josaa/abstract.cfm?URI=josaa-6-5-682.

Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, et al. Array programming with NumPy. *Nature*, 585(7825):357–362, 2020. doi: 10.1038/s41586-020-2649-2.

Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2261–2269, 2017. doi: 10.1109/CVPR.2017.243.

Ramakrishna Kakarala. *Triple Correlation on Groups*. PhD thesis, University of California, Irvine, 1992.

Ramakrishna Kakarala. The bispectrum as a source of phase-sensitive invariants for Fourier descriptors: A group-theoretic approach. *Journal of Mathematical Imaging and Vision*, 44(3):341–353, 2012. doi: 10.1007/s10851-012-0330-6. URL https://arxiv.org/abs/0902.0196.

Risi Kondor. A novel set of rotationally and translationally invariant features for images based on the non-commutative bispectrum. *arXiv preprint cs/0701127*, 2007.

Thijs P. Kuipers and Erik J. Bekkers. Regular SE(3) group convolutions for volumetric medical image analysis. In *Medical Image Computing and Computer Assisted Intervention – MICCAI 2023*, volume 14222 of *Lecture Notes in Computer Science*, pages 252–261. Springer Nature Switzerland, 2023. doi: 10.1007/978-3-031-43898-1_25. URL https://doi.org/10.1007/978-3-031-43898-1_25.

Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. In *International Conference on Learning Representations (ICLR)*, 2017. URL https://openreview.net/forum?id=Skq89Scxx.

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

Stéphane Mallat. Group invariant scattering. *Communications on Pure and Applied Mathematics*, 65(10):1331–1398, 2012. doi: 10.1002/cpa.21413.

Nicholas F. Marshall, Oscar Mickelin, and Amit Singer. Fast expansion into harmonics on the disk: A steerable basis with fast radial convolutions. *SIAM Journal on Scientific Computing*, 45(5): A2431–A2457, 2023. doi: 10.1137/22M1542775.

Simon Mataigne, Johan Mathe, Sophia Sanborn, Christopher Hillar, and Nina Miolane. The selective G-bispectrum and its inversion: Applications to G-invariant networks. In *Advances in Neural Information Processing Systems*, volume 37, pages 115682–115711. Neural Information Processing Systems Foundation, Inc., 2024. doi: 10.52202/079017-3674. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/d1a1e8713fcd5626656553c82f7c3b26-Abstract-Conference.html.

Jason D. McEwen, Christopher G. R. Wallis, and Augustine N. Mavor-Parker. Scattering networks on the sphere for scalable and rotationally equivariant spherical CNNs. In *International Conference on Learning Representations (ICLR)*, 2022. URL https://arxiv.org/abs/2102.02828.

Adele Myers Lantow and Nina Miolane. Selective disk bispectrum: A complete and rotation invariant image descriptor. *arXiv preprint arXiv:2511.19706*, 2026. URL https://arxiv.org/abs/2511.19706.

Valentin Oreiller, Vincent Andrearczyk, Julien Fageot, John O. Prior, and Adrien Depeursinge. 3D solid spherical bispectrum CNNs for biomedical texture analysis. *arXiv preprint arXiv:2004.13371*, 2020. URL https://arxiv.org/abs/2004.13371.

Valentin Oreiller, Julien Fageot, Vincent Andrearczyk, John O. Prior, and Adrien Depeursinge. Robust multi-organ nucleus segmentation using a locally rotation invariant bispectral U-Net. In *Proceedings of The 5th International Conference on Medical Imaging with Deep Learning*, volume 172 of *Proceedings of Machine Learning Research*, pages 929–943. PMLR, 2022. URL https://proceedings.mlr.press/v172/oreiller22a.html.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy,

Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html.

Sophia Sanborn, Christian Shewmake, Bruno Olshausen, and Christopher Hillar. Bispectral neural networks. In *International Conference on Learning Representations (ICLR)*, 2023.

Mateus Sangalli, Samy Blusseau, Santiago Velasco-Forero, and Jesús Angulo. Moving frame net: SE(3)-equivariant network for volumes. In *Proceedings of the 1st NeurIPS Workshop on Symmetry and Geometry in Neural Representations*, volume 197 of *Proceedings of Machine Learning Research*, pages 81–97. PMLR, 2023. URL https://proceedings.mlr.press/v197/sangalli23a.html.

John W. Tukey. The spectral representation and transformation properties of the higher moments of stationary time series. In David R. Brillinger, editor, *The Collected Works of John W. Tukey*, volume 1, pages 165–184. Wadsworth, Belmont, CA, 1953. Unpublished manuscript, reprinted in 1984.

Bastiaan S. Veeling, Jasper Linmans, Jim Winkens, Taco Cohen, and Max Welling. Rotation equivariant CNNs for digital pathology. In *Medical Image Computing and Computer Assisted Intervention (MICCAI)*, volume 11071 of *Lecture Notes in Computer Science*, pages 210–218. Springer, 2018. doi: 10.1007/978-3-030-00934-2_24.

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

Jiancheng Yang, Rui Shi, Donglai Wei, Zequan Liu, Lin Zhao, Bilian Ke, Hanspeter Pfister, and Bingbing Ni. MedMNIST v2 – a large-scale lightweight benchmark for 2D and 3D biomedical image classification. *Scientific Data*, 10(1):41, 2023. doi: 10.1038/s41597-022-01721-8.

Zhizhen Zhao and Amit Singer. Rotationally invariant image representation for viewing direction classification in cryo-EM. *Journal of Structural Biology*, 186(1):153–166, 2014. doi: 10.1016/j.jsb.2014.03.003.

Dmitrii Zhemchuzhnikov and Sergei Grudinin. ILPO-NET: Network for the invariant recognition of arbitrary volumetric patterns in 3D. In *Machine Learning and Knowledge Discovery in Databases. Research Track (ECML PKDD)*, volume 14944 of *Lecture Notes in Computer Science*, pages 352–368. Springer Nature Switzerland, 2024. doi: 10.1007/978-3-031-70359-1_21. URL https://doi.org/10.1007/978-3-031-70359-1_21.

Dmitrii Zhemchuzhnikov and Sergei Grudinin. On the Fourier analysis in the SO(3) space: the EquiLoPO network. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://openreview.net/forum?id=LvTSvdiSwG.

A Augmented Selective Spherical Bispectrum – Structural Background

This appendix gives the structural background for the construction of Section 5. Section A.1 fixes the S^2 band-limited setup and notation. Section A.2 derives the orbit-space dimension lower bound $(L+1)^2 - 3$ that motivates the $\Theta(L^2)$ target. Section A.3 gives the index set as implemented (Algorithm 2) and its size, and Section A.4 the parity structure of the scalar bispectrum that determines which triples vanish identically on real signals. Section A.5 compares the augmented selective spherical bispectrum with prior bispectral constructions, and Section A.6 counts its cost in pixel space. We do not claim a completeness proof. Generic completeness is a conjecture, supported by the full rank of the per-degree Jacobian (Section 5) and by the reconstructions of Figure 2.

A.1 Setup and notation

Definition 1 (Band-limited signal on S^2)**.** A signal $f : S^2 \to \mathbb{R}$ is *band-limited* at degree L if its spherical harmonic expansion

$$f(\theta, \varphi) = \sum_{\ell=0}^{L} \sum_{m=-\ell}^{\ell} a_\ell^m \, Y_\ell^m(\theta, \varphi) \tag{6}$$

has $a_\ell^m = 0$ for all $\ell > L$. We write $\mathbf{F}_\ell = (a_\ell^{-\ell}, \ldots, a_\ell^\ell) \in \mathbb{C}^{2\ell+1}$ for the coefficient vector at degree ℓ.

For real f, the conjugation symmetry $a_\ell^{-m} = (-1)^m \overline{a_\ell^m}$ holds, so $\mathbf{F}_\ell$ has $2\ell + 1$ real degrees of freedom. The total signal dimension is $\dim = \sum_{\ell=0}^{L}(2\ell + 1) = (L + 1)^2$.

Definition 2 (Rotation action)**.** Under $g \in \mathrm{SO}(3)$, the spherical harmonic coefficients transform as $\mathbf{F}_\ell \mapsto D^\ell(g)\, \mathbf{F}_\ell$, where D^ℓ is the $(2\ell + 1)$-dimensional Wigner D-matrix.

In the irrep notation of Mataigne et al. [2024], $\{\rho_\ell\}_{\ell \geq 0}$ are the (unitary) irreducible representations of $\mathrm{SO}(3)$ with V_{ρ_ℓ} the $(2\ell{+}1)$-dimensional space of degree-ℓ spherical harmonics and $\rho_\ell(g) = D^\ell(g)$. The Fourier coefficient of f at ρ_ℓ is $\mathcal{F}(f)_{\rho_\ell} = \mathbf{F}_\ell$. We use $\mathbf{F}_\ell$ as the working symbol throughout.

Definition 3 (Bispectrum on S^2)**.** For each triple (ℓ_1, ℓ_2, ℓ) satisfying the triangle inequality $|\ell_1 - \ell_2| \leq \ell \leq \ell_1 + \ell_2$, the *bispectral coefficient* is

$$\beta_{\ell_1,\ell_2,\ell} = \sum_{\substack{m_1,\, m_2,\, m \\ m_1+m_2=m}} C^{\ell,\, m}_{\ell_1,\, m_1;\, \ell_2,\, m_2} \; a_{\ell_1}^{m_1} \, a_{\ell_2}^{m_2} \, \overline{a_\ell^m}, \tag{7}$$

where $C^{\ell,\, m}_{\ell_1,\, m_1;\, \ell_2,\, m_2}$ denotes the Clebsch–Gordan coefficient $\langle \ell_1, m_1;\, \ell_2, m_2 \mid \ell, m \rangle$. Each $\beta_{\ell_1,\ell_2,\ell}$ is a scalar. For real f, it is real when $\ell_1{+}\ell_2{+}\ell$ is even and purely imaginary when $\ell_1{+}\ell_2{+}\ell$ is odd (Lemma 11).

Definition 3 adapts the G-bispectrum (2) to $\mathrm{SO}(3)$ acting on S^2: the irreps are Wigner-D matrices, the Fourier coefficients are the vectors $\mathbf{F}_\ell$ rather than square matrices, and contracting with the Clebsch–Gordan coefficients leaves one scalar per triple.

Definition 4 (CG power spectrum)**.** For a triple (ℓ_1, ℓ_2, ℓ) satisfying the triangle inequality, write $(\mathbf{F}_{\ell_1} \otimes \mathbf{F}_{\ell_2})\big|_\ell$ for the projection of the Kronecker product $\mathbf{F}_{\ell_1} \otimes \mathbf{F}_{\ell_2}$ onto the component that transforms by D^ℓ, realized by contracting with the CG coefficients $C^{\ell,\, m}_{\ell_1,\, m_1;\, \ell_2,\, m_2}$. The *CG power spectrum entry* is the squared norm of this projection,

$$P_{\ell_1,\ell_2,\ell} := \|(\mathbf{F}_{\ell_1} \otimes \mathbf{F}_{\ell_2})|_\ell\|^2 = \sum_{m=-\ell}^{\ell} \Big| \sum_{\substack{m_1,m_2 \\ m_1+m_2=m}} C^{\ell,\, m}_{\ell_1,\, m_1;\, \ell_2,\, m_2} \; a_{\ell_1}^{m_1} \, a_{\ell_2}^{m_2} \Big|^2. \tag{8}$$

This is a degree-4, real-valued, $\mathrm{SO}(3)$-invariant polynomial in the signal coefficients. The CG-power part $\mathcal{S}_P$ of the augmented selective spherical bispectrum is a fixed family of these scalars, at most seven per degree and $O(L)$ in total (77 at $L = 15$), listed in Algorithm 2. With them included, the per-degree Jacobian of Section 5 has full rank $2\ell{+}1$.

Proposition 5 (Size of the full bispectrum)**.** *The number of admissible triples (ℓ_1, ℓ_2, ℓ) with $0 \leq \ell_1, \ell_2, \ell \leq L$ satisfying the triangle inequality $|\ell_1 - \ell_2| \leq \ell \leq \ell_1 + \ell_2$ is $\Theta(L^3)$.*

Proof. For fixed ℓ_1 and ℓ_2, the triangle inequality admits $\min(\ell_1 + \ell_2, L) - |\ell_1 - \ell_2| + 1$ valid values of ℓ. Summing over $(\ell_1, \ell_2) \in [0, L]^2$ gives $N_{\text{full}} = \Theta(L^3)$. □

Remark 6 (Invariance). Each $\beta_{\ell_1,\ell_2,\ell}$ is $\mathrm{SO}(3)$-invariant: the Wigner matrices satisfy $D^{\ell_1} \otimes D^{\ell_2} = \bigoplus_\ell D^\ell$ under the CG decomposition, so the triple product contracted with CG coefficients is unchanged by $\mathbf{F}_\ell \mapsto D^\ell(g)\mathbf{F}_\ell$.

A.2 Lower bound on the output size

Proposition 7 (Dimensional lower bound)**.** *Any* smooth or polynomial *complete* $\mathrm{SO}(3)$*-invariant for band-limited signals on* S^2 *at band limit* L *must have codomain of dimension at least* $(L+1)^2-3$*.*

Proof. Let $\Phi : \mathbb{R}^{(L+1)^2} \to \mathbb{R}^k$ be a smooth (or polynomial) $\mathrm{SO}(3)$-invariant map that separates orbits on an open dense subset $U \subset \mathbb{R}^{(L+1)^2}$, and let $r := \max_{x\in U} \operatorname{rank} d\Phi_x$. The rank of $d\Phi$ is lower semicontinuous, so $W := \{x \in U : \operatorname{rank} d\Phi_x = r\}$ is open and nonempty. By the constant-rank theorem, for $x \in W$ the level set $\Phi^{-1}(\Phi(x)) \cap W$ is, near x, a submanifold of dimension $(L+1)^2 - r$. Orbit separation on U places it inside the orbit $\mathrm{SO}(3) \cdot x$, a manifold of dimension at most $\dim \mathrm{SO}(3) = 3$. Hence $(L+1)^2 - r \le 3$, and $k \ge r \ge (L+1)^2 - 3 = \Omega(L^2)$. □

Remark 8 (Smoothness assumption). The smoothness assumption is essential. Without continuity, a single real number can separate orbits (e.g., a Borel injection of the orbit space into $\mathbb{R}$), but such an invariant is of no use in a network. The bispectral and Clebsch–Gordan power invariants used here are polynomial in the coefficients, so the proposition applies.

A.3 The selective index set

This subsection describes the index set as implemented in `bispectrum.so3_on_s2`, which Algorithm 2 mirrors. The *degree* of an entry is its largest index. We call $\beta_{0,\ell,\ell} = a_0^0 \|\mathbf{F}_\ell\|^2$ the *power entry* of degree ℓ and $\beta_{\ell,\ell,\ell'}$ with $\ell' \le \ell$ its *self-couplings*.

Definition 9 (Augmented selective spherical bispectrum)**.** The *augmented selective spherical bispectrum* Φ_{sel} at band limit L consists of a bispectral part $\mathcal{S}_\beta$ (entries $\beta_{\ell_1,\ell_2,\ell}$) and a CG-power part $\mathcal{S}_P$ (entries $P_{\ell_1,\ell_2,\ell}$, Definition 4), assembled degree by degree. Degrees $\ell \le \min(L, 7)$ use the fixed entries of Table 4. Each degree $\ell \ge 8$ receives

(i) the first $2\ell{+}1$ entries, among those that do not vanish identically on real signals (Proposition 13), of the list made of the triples (a, b, ℓ) with $1 \le a \le b \le \ell{-}1$ and $a{+}b \ge \ell$, ordered by $a{+}b$ and then by a, followed by the power entry $(0, \ell, \ell)$ and the self-couplings (ℓ, ℓ, ℓ') for even $\ell' < \ell$ in descending order,
(ii) every even self-coupling (ℓ, ℓ, ℓ'), $2 \le \ell' \le \ell$, not already selected, and
(iii) five (even ℓ) or seven (odd ℓ) CG-power triples (ℓ_1, ℓ, ℓ') with $\ell_1 \le 3$ (Algorithm 2).

At $\ell \in \{8, 9\}$ the list in (i) has only 16 nonzero triples (a, b, ℓ), so the power entry and self-couplings make up the remaining entries. From $\ell = 10$ on, the triples alone supply all $2\ell{+}1$. Only one ordering of each triple is listed because, on real signals, all orderings of a triple are proportional (Remark 17).

Mathematically, $\Phi_{\mathrm{sel}} : V_L \to \mathbb{R}^N$, where $V_L \cong \mathbb{R}^{(L+1)^2}$ is the space of real signals band-limited at L, has N real components: by Lemma 11, each bispectral entry contributes one real component on real signals (real for even parity, purely imaginary for odd), and CG-power entries are real-valued degree-4 invariants. The PyTorch implementation stores each scalar in a complex tensor, so downstream layers receive $2N$ real channels after splitting real and imaginary parts. For bispectral entries, exactly one of the two channels is identically zero by parity, and for CG-power entries the imaginary channel is identically zero.

Table 4: Entries of Φ_{sel} at degrees $\ell \leq 7$, as implemented in the library. Each triple (ℓ_1, ℓ_2, ℓ_3) denotes $\beta_{\ell_1,\ell_2,\ell_3}$ in the bispectral columns and P_{ℓ_1,ℓ_2,ℓ_3} in the CG-power column. The capped column holds the entries counted against the budget of $2\ell{+}1$ per degree (10 at $\ell = 4$), and the added column holds the even self-couplings appended beyond it. For $4 \leq \ell \leq 7$ the capped entries are the fixed lists $\mathcal{T}_\ell^{\mathrm{small}}$ of triples (a, b, ℓ) and reorderings (a, ℓ, b) with $a, b < \ell$, without the identically-zero triples $(4, 5, 4)$ and $(6, 7, 6)$, padded with the power entry $(0, \ell, \ell)$ when the list has fewer than $2\ell{+}1$ entries. In total, there are 22 bispectral and 10 CG-power entries at $\ell \leq 4$, and 69 and 29 at $\ell \leq 7$.

ℓ	Capped bispectral	Added self-couplings	CG-power
0	$(0,0,0)$	–	–
1	$(0,1,1)$	–	–
2	$(1,1,2)$, $(0,2,2)$, $(2,2,2)$	–	$(1,2,1)$
3	$(1,2,3)$, $(1,3,2)$, $(2,3,1)$, $(0,3,3)$, $(3,3,2)$	–	$(1,3,2)$, $(2,3,1)$, $(2,3,2)$, $(3,3,2)$
4	$(1,3,4)$, $(2,2,4)$, $(2,3,4)$, $(3,3,4)$, $(1,4,3)$, $(2,4,2)$, $(3,4,1)$, $(2,4,3)$, $(3,4,2)$, $(3,4,3)$	$(4,4,2)$, $(4,4,4)$	$(1,4,3)$, $(1,4,4)$, $(2,4,2)$, $(2,4,3)$, $(3,4,1)$
5	$(1,4,5)$, $(2,3,5)$, $(2,4,5)$, $(3,4,5)$, $(1,5,4)$, $(2,5,3)$, $(3,5,2)$, $(4,5,1)$, $(2,5,4)$, $(3,5,4)$, $(0,5,5)$	$(5,5,2)$, $(5,5,4)$	$(1,5,4)$, $(1,5,5)$, $(2,5,3)$, $(2,5,4)$, $(3,5,2)$, $(3,5,3)$, $(3,5,4)$
6	$(1,5,6)$, $(2,4,6)$, $(3,3,6)$, $(3,4,6)$, $(1,6,5)$, $(2,6,4)$, $(3,6,3)$, $(4,6,2)$, $(5,6,1)$, $(2,6,5)$, $(3,6,5)$, $(4,6,5)$, $(5,6,5)$	$(6,6,2)$, $(6,6,4)$, $(6,6,6)$	$(1,6,5)$, $(1,6,6)$, $(2,6,4)$, $(2,6,5)$, $(3,6,3)$
7	$(1,6,7)$, $(2,5,7)$, $(3,4,7)$, $(4,5,7)$, $(1,7,6)$, $(2,7,5)$, $(3,7,4)$, $(4,7,3)$, $(5,7,2)$, $(6,7,1)$, $(2,7,6)$, $(3,7,6)$, $(4,7,6)$, $(5,7,6)$, $(0,7,7)$	$(7,7,2)$, $(7,7,4)$, $(7,7,6)$	$(1,7,6)$, $(1,7,7)$, $(2,7,5)$, $(2,7,6)$, $(3,7,4)$, $(3,7,5)$, $(3,7,6)$

Table 5: Coefficient counts of Φ_{sel} at representative band limits. "Bispectral" counts the triples in $\mathcal{S}_\beta$ (including the even self-couplings $\beta_{\ell,\ell,\ell'}$ that every degree $\ell \geq 3$ receives). "CG-power" counts entries in $\mathcal{S}_P$, and "Total N" is $|\Phi_{\mathrm{sel}}|$ (real components) and matches the library's `output_size`. "Stored channels" $(2N)$ counts the real and imaginary channels seen by the Spherical MNIST classifier. One channel per bispectral entry is identically zero by parity, and the imaginary channel of each CG-power entry is identically zero. "Full" counts the entries of the full bispectrum computed by `SO3onS2(selective=False)`: every admissible triple with $\ell_1 \leq \ell_2 \leq L$ and $\ell \leq L$, including those that vanish identically on real signals.

L	Bispectral	CG-power	Total $N = \lvert\Phi_{\mathrm{sel}}\rvert$	Stored channels $(2N)$	Full
4	22	10	32	64	42
5	35	17	52	104	69
15	303	77	380	760	1,124
16	344	82	426	852	1,341

Algorithm 2 Index set of the augmented selective spherical bispectrum at band limit L, as implemented. Returns the bispectral triples $\mathcal{S}_\beta$ and the CG-power triples $\mathcal{S}_P$.

Input: band limit L
$\mathcal{S}_\beta \leftarrow$ bispectral entries of degree $\leq \min(L, 7)$ {Table 4}
$\mathcal{S}_P \leftarrow$ CG-power entries of degree $\leq \min(L, 7)$ {Table 4}
for $\ell = 8, \ldots, L$ **do**
  $C \leftarrow [(a, b, \ell) : 1 \leq a \leq b \leq \ell{-}1,\ a{+}b \geq \ell]$, ordered by $a{+}b$, then a
  append $(0, \ell, \ell)$, then (ℓ, ℓ, ℓ') for even $\ell' = \ell{-}1, \ldots, 2$, to C
  remove from C the entries with odd $\ell_1{+}\ell_2{+}\ell_3$ and a repeated index {$\equiv 0$ on real signals, Prop. 13}
  $\mathcal{S}_\beta \leftarrow \mathcal{S}_\beta \cup$ first $\min(2\ell{+}1, |C|)$ entries of C
  $\mathcal{S}_\beta \leftarrow \mathcal{S}_\beta \cup \{(\ell, \ell, \ell') : \ell'$ even, $2 \leq \ell' \leq \ell\}$ {even self-couplings}
  $\mathcal{S}_P \leftarrow \mathcal{S}_P \cup \{(1, \ell, \ell{-}1),\ (1, \ell, \ell),\ (2, \ell, \ell{-}2),\ (2, \ell, \ell{-}1),\ (3, \ell, \ell{-}3)\}$
  if ℓ is odd **then**
    $\mathcal{S}_P \leftarrow \mathcal{S}_P \cup \{(3, \ell, \ell{-}2),\ (3, \ell, \ell{-}1)\}$
  end if
end for
return $(\mathcal{S}_\beta, \mathcal{S}_P)$

Proposition 10 (Output size). *As $L \to \infty$, $|\Phi_{\mathrm{sel}}| = \Theta(L^2)$. Each degree $\ell \geq 8$ contributes at most $2\ell{+}1$ bispectral entries, at most $\lfloor \ell/2 \rfloor$ further even self-couplings, and at most seven CG-power*

entries. The degrees $\ell \leq 7$ contribute a constant number (Table 4), and the first $2\ell{+}1$ entries of each degree alone already number $\Theta(L^2)$. Per-L counts are in Table 5.

A.4 Parity structure of the bispectrum

The selective construction of Section 5 exploits two structural facts about the scalar bispectrum $\beta_{\ell_1,\ell_2,\ell}$: a parity rule that fixes whether each entry is real or purely imaginary on real signals (Lemma 11) and the resulting identical vanishing of certain odd-parity entries with a repeated index (Proposition 13). Together with the proportionality of the orderings of a triple, the lemma fixes how many independent equations in $\mathbf{F}_\ell$ the bispectral entries of each degree supply (Remark 17). The proposition fixes which candidate entries Algorithm 2 discards.

Let T_R be the reflection $(T_R \cdot f)(\theta,\varphi) := f(\theta,-\varphi)$ through the xz-plane, an element of $\mathrm{O}(3) \setminus \mathrm{SO}(3)$. In the Condon–Shortley convention behind the reality condition $a_\ell^{-m} = (-1)^m \overline{a_\ell^m}$, it acts on coefficients as $a_\ell^m \mapsto (-1)^m a_\ell^{-m}$. The signal $T_R \cdot f$ is the mirror image of f, which is generally not a rotation of f, so a complete $\mathrm{SO}(3)$-invariant must tell them apart.

Lemma 11 (Parity of the bispectrum)**.** *For any signal (real or complex):*

$$\beta_{\ell_1,\ell_2,\ell}(T_R \cdot f) = (-1)^{\ell_1+\ell_2+\ell}\, \beta_{\ell_1,\ell_2,\ell}(f). \tag{9}$$

For real *signals, there is an independent constraint:*

$$\overline{\beta_{\ell_1,\ell_2,\ell}(f)} = (-1)^{\ell_1+\ell_2+\ell}\, \beta_{\ell_1,\ell_2,\ell}(f). \tag{10}$$

Together:

- *(a) If $\ell_1+\ell_2+\ell$ is even: β is real-valued (by (10)) and T_R-invariant (by (9)).*
- *(b) If $\ell_1+\ell_2+\ell$ is odd: β is purely imaginary (by (10)) and $\beta(T_R \cdot f) = -\beta(f)$, so $\operatorname{Im}\beta$ flips sign under T_R.*

Proof. Under T_R, each $a_\ell^m \mapsto (-1)^m a_\ell^{-m}$. Substituting into $\beta = \sum_m (\sum_{m_1+m_2=m} C\, a^{m_1} a^{m_2}) \overline{a^m}$, the three sign factors $(-1)^{m_1+m_2+m} = (-1)^{2m} = 1$ cancel. Re-indexing $m \mapsto -m$ and applying the CG symmetry $C^{\ell,\,-m}_{\ell_1,\,-m_1;\,\ell_2,\,-m_2} = (-1)^{\ell_1+\ell_2-\ell}\, C^{\ell,\,m}_{\ell_1,\,m_1;\,\ell_2,\,m_2}$ yields (9) (with $(-1)^{\ell_1+\ell_2-\ell} = (-1)^{\ell_1+\ell_2+\ell}$).

For real signals, $\overline{a_\ell^m} = (-1)^m a_\ell^{-m}$. Taking the complex conjugate of β and re-indexing similarly yields (10). □

Lemma 12 (CG power is T_R-invariant)**.** *For all triples (ℓ_1,ℓ_2,ℓ): $P_{\ell_1,\ell_2,\ell}(T_R \cdot f) = P_{\ell_1,\ell_2,\ell}(f)$.*

Proof. Under T_R, the CG-projected component transforms as $(\mathbf{F}_{\ell_1} \otimes \mathbf{F}_{\ell_2})|_\ell^M \mapsto (-1)^{\ell_1+\ell_2+\ell+M} (\mathbf{F}_{\ell_1} \otimes \mathbf{F}_{\ell_2})|_\ell^{-M}$, where the superscript M selects the $m = M$ component. Taking the squared norm and re-indexing the sum over M leaves P unchanged. □

Proposition 13 (Vanishing of odd-parity entries with a repeated index)**.** *Let $\beta_{\ell_1,\ell_2,\ell}$ be the scalar bispectrum entry of a real signal f (so $\ell_1+\ell_2+\ell$ odd). If the index multiset $\{\ell_1,\ell_2,\ell\}$ contains a repeated value, then $\beta_{\ell_1,\ell_2,\ell}(f) \equiv 0$ as a polynomial in the SH coefficients of f.*

Proof. Define the symmetric scalar 3j contraction

$$T_{\ell_1,\ell_2,\ell_3}(f) := \sum_{m_1,m_2,m_3} \begin{pmatrix} \ell_1 & \ell_2 & \ell_3 \\ m_1 & m_2 & m_3 \end{pmatrix} a_{\ell_1}^{m_1}\, a_{\ell_2}^{m_2}\, a_{\ell_3}^{m_3}.$$

Converting the Wigner 3j into a Clebsch–Gordan coefficient via $\left(\begin{smallmatrix} j_1 & j_2 & j_3 \\ m_1 & m_2 & m_3 \end{smallmatrix}\right) = \frac{(-1)^{j_1-j_2-m_3}}{\sqrt{2j_3+1}}\, C^{j_3,\,-m_3}_{j_1,\,m_1;\,j_2,\,m_2}$ and applying the reality identity $a_{\ell_3}^{m_3} = (-1)^{m_3}\, \overline{a_{\ell_3}^{-m_3}}$ shows

$$\beta_{\ell_1,\ell_2,\ell}(f) = (-1)^{\ell_1+\ell_2} \sqrt{2\ell+1}\, T_{\ell_1,\ell_2,\ell}(f),$$

so β and T have the same vanishing locus. The 3j symbol is invariant under even column permutations and picks up $(-1)^{\ell_1+\ell_2+\ell_3}$ under odd ones. If two columns i,j satisfy $\ell_i = \ell_j$, the inserted vectors a_{ℓ_i}, a_{ℓ_j} are literally the same coefficient vector of f, so swapping those two columns leaves the contracted sum T unchanged while multiplying the 3j prefactor by $(-1)^{\ell_1+\ell_2+\ell_3}$. For odd parity this forces $T = -T$, hence $T \equiv 0$ and $\beta \equiv 0$. □

Remark 14 (Identically-zero candidates). Proposition 13 rules out, for example, (ℓ, ℓ, ℓ') with odd ℓ', $(2, 3, 2)$, $(4, 5, 4)$, and $(6, 7, 6)$. Algorithm 2 discards every such candidate, including the triples $(4, 5, 4)$ and $(6, 7, 6)$ of the lists $\mathcal{T}_\ell^{\text{small}}$ (Table 4).

Remark 15 (First nonzero odd-parity entries). The first odd-parity entries with all-distinct indices, and hence not covered by Proposition 13, occur at degree 4: $\beta_{2,3,4}$ and its orderings $\beta_{2,4,3}$ and $\beta_{3,4,2}$. They are not identically zero (a random real signal gives a nonzero value), so by Lemma 11 they change sign under T_R.

Remark 16 (No odd-parity entries at $\ell \leq 3$). For degrees ≤ 3, every odd-parity triple (ℓ_1, ℓ_2, ℓ) has at least two indices equal (since $\max(\ell_1, \ell_2, \ell) \leq 3$ and $\ell_1 + \ell_2 + \ell$ odd forces at least one pair to be equal in this range). By Proposition 13, all odd-parity entries vanish. Therefore, together with Lemma 12, every entry of degree ≤ 3 takes the same value on f and on its mirror image $T_R \cdot f$.

Remark 17 (Distinct equations per degree). On real signals each bispectral entry is one real number (Lemma 11), and all orderings of a triple are proportional: $\beta_{\ell_1,\ell_2,\ell} = (-1)^{\ell_1+\ell_2+\ell}\beta_{\ell_2,\ell_1,\ell}$ by the CG exchange symmetry, and $\beta_{\ell_1,\ell_2,\ell} = (-1)^{\ell_1}\sqrt{(2\ell+1)/(2\ell_2+1)}\,\beta_{\ell_1,\ell,\ell_2}$ by the 3j form in the proof of Proposition 13. Only distinct unordered triples therefore count toward the $2\ell{+}1$ real unknowns of $\mathbf{F}_\ell$, and for $\ell \geq 8$ Algorithm 2 lists one ordering per triple. At $\ell \in \{8, 9\}$ only 16 such triples with $a, b < \ell$ are nonzero, so the power entry and self-couplings make up the remaining entries. The lists $\mathcal{T}_\ell^{\text{small}}$ at $4 \leq \ell \leq 7$ (Table 4) contain several orderings of some triples and supply 4, 4, 8, and 8 distinct bispectral equations from their 10, 10, 13, and 14 nonzero entries.

A.5 Comparison with existing results

Table 6: Comparison of bispectral invariants for signals on S^2 or finite groups. *constr.* marks a constructive proof. Bendory et al. [2025] recover signals on $\mathbb{R}^3$ sampled on concentric spherical shells by solving linear systems one degree at a time (frequency marching), and three shells suffice.

Result	Invariant	Size	Complete?
Kakarala [2012]	Full bispec. (on SO(3))	$O(L^3)$	Yes (constr.)
Edidin and Satriano [2024]	Full bispec. (band-lim.)	$O(L^3)$	Yes (generic)
Bendory et al. [2025]	Freq. marching	$O(L^2)$/shell	Yes (3 shells)
Mataigne et al. [2024]	Sel. G-bispec. (finite)	$O(\lvert G\rvert)$	Yes (generic)
This work	**Augmented selective bispectrum**	$\boldsymbol{\Theta(L^2)}$	**Empirical (Section 5)**

A.6 Complexity in pixel space

A signal band-limited at degree L is sampled on an equiangular grid of $P = \Theta(L^2)$ pixels (e.g., 34×68 at $L = 16$ in Table 3), so $L = \Theta(\sqrt{P})$. The augmented selective bispectrum has $\Theta(L^2) = \Theta(P)$ entries, linear in the number of pixels, against $\Theta(L^3) = \Theta(P^{3/2})$ for the full bispectrum. Each entry contracts up to $O(L^2)$ CG-weighted products (Eq. (7)), so the forward contraction as implemented costs $O(L^4) = O(P^2)$ for Φ_{sel} and $O(L^5) = O(P^{5/2})$ for the full bispectrum. Counting the terms of the contraction at $L = 64$ (128×256 grid): the full bispectrum has 70,257 entries and 2.67×10^8 terms, and Φ_{sel} has 5,610 entries and 1.08×10^7 terms, a $25\times$ reduction. Between $L = 32$ and $L = 64$ the term counts grow as $L^{4.86}$ (full) and $L^{3.71}$ (Φ_{sel}), measured as log-log slopes.

B Extended Benchmark Results

The scaling plots in this appendix extend Table 3 and were produced by `benchmarks/benchmark.py` on a single NVIDIA H100 80 GB GPU (PyTorch 2.11, `bispectrum` 0.4.1), with autograd enabled as during training. Section B.4 covers other devices.

B.1 Coefficient-count scaling

B.2 Forward-pass timing

The selective forward pass is dominated by kernel-launch overhead rather than arithmetic. The full bispectrum grows with $|G|$ and dominates the cost at large group sizes.

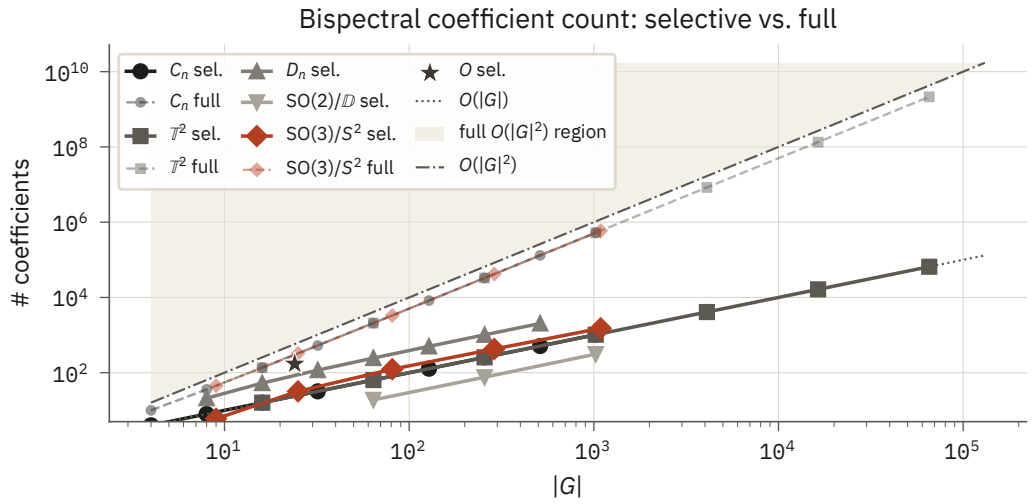

Figure 5: **The number of selective G-bispectral coefficients grows linearly with the group size. For full G-bispectra, it grows quadratically.** Invariant coefficients per signal against $|G|$: selective (solid) and full (dashed). For $\mathrm{SO}(2)$ on the disk and $\mathrm{SO}(3)$ on S^2, $|G|$ is the band-limited signal dimension. At $|G|$=65,536 the selective $\mathbb{T}^2$-bispectrum has 32,768× fewer coefficients than the full $\mathbb{T}^2$-bispectrum (6.6×10^4 vs. 2.1×10^9). The full bispectra for D_n, the disk $\mathbb{D}$ and O are not implemented.

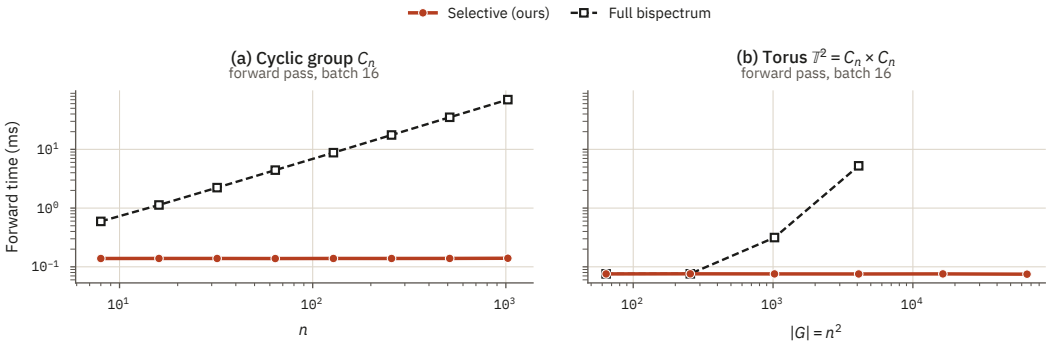

Figure 6: **The selective forward pass costs the same at every group size. The full bispectrum's cost grows with it.** Median forward time, H100, batch 16, autograd enabled. (a) C_n: the selective pass takes 0.14 ms at every n from 8 to 1,024. At n=1,024, the full bispectrum takes 70 ms, 502× longer. (b) $\mathbb{T}^2$: 0.075–0.076 ms up to $|G|$=65,536. The full bispectrum, run up to $|G|$=4,096 (8,390,656 coefficients), takes 5.2 ms there, 69× longer.

B.3 GPU throughput and inversion

Throughput grows linearly with batch size up to 4,096 for C_n, D_n, and O and up to 1,024 for $\mathbb{T}^2$: per-call cost stays nearly constant, consistent with the launch-dominated forward pass above. $\mathrm{SO}(3)$ on S^2 stops scaling at batch ≈256, where its spherical harmonic transform and Clebsch–Gordan contractions, rather than launch overhead, set the cost.

B.4 Portability across devices and backends

Table 7 times the selective configurations of Table 3 on four GPUs and four host CPUs with a single script (`experiments/tooling/benchmark_devices.py`): batch = 16, float32 input, `torch.inference_mode()`, median over 30 repetitions of 20 forward passes after 10 warm-up passes, PyTorch 2.11, `bispectrum` 0.4.1 (0.4.0 for this table's H100 and A100 rows, a release

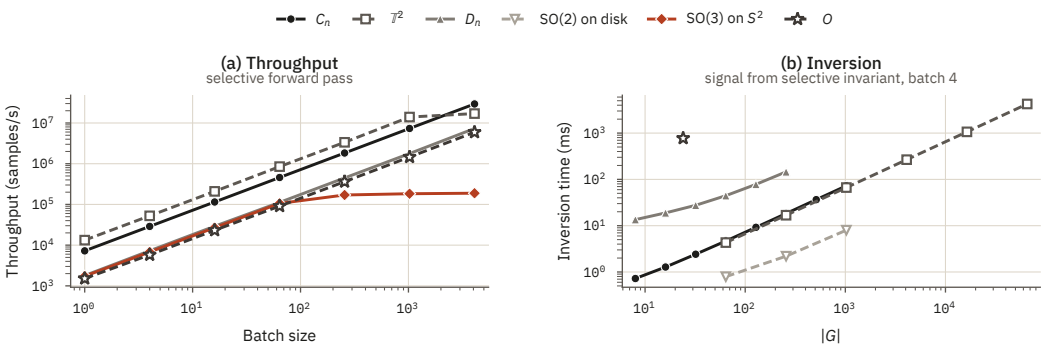

Figure 7: **Selective G-bispectra on finite groups reach millions of samples/s on one H100, and the library inverts every module except $\mathrm{SO}(3)$ on S^2.** (a) Selective forward throughput against batch size, H100, autograd enabled (C_{128}, $C_{32}\times C_{32}$, D_{32}, S^2 at $L{=}16$, O): C_n and $\mathbb{T}^2$ exceed 10^7 samples/s, O reaches 5.9×10^6, and $\mathrm{SO}(3)$ on S^2 saturates near 1.9×10^5. (b) Time to recover a signal from its selective invariant (batch 4). Log–log slopes against $|G|$ are 1.0 for C_n, 1.0 for $\mathbb{T}^2$, 0.7 for D_n and 0.8 for the disk. O (775 ms) uses iterative Levenberg–Marquardt corrections. The $\mathrm{SO}(3)$ module on S^2 has no closed-form inversion.

that differs from 0.4.1 only on MPS). Every module runs on every backend. All modules are sub-millisecond on the four GPUs, which span data-center and consumer NVIDIA parts, two CUDA versions (12.8 and 13.0) and Apple MPS. On the consumer RTX 4090, `SO3onS2`, which replays a CUDA graph, is faster than on the H100, while the other modules are 1.5–1.7× slower, a gap comparable to that between the two hosts' single-thread CPU timings: at batch = 16 these modules are bound by host-side kernel launches rather than by the GPU. On CPU, every module except `SO3onS2` is sub-millisecond, and `SO3onS2` takes 2.2–45 ms depending on host and thread count. CPU rows fix the number of intra-op threads: with PyTorch's default of one thread per core, a shared 128-core host took 402 ms for `SO3onS2`. MPS has no float64, so modules are cast to float32 there (`bispectrum` $\geq$ 0.4.1). On CPU and CUDA, the precomputed tables keep the dtype they are built with, and `SO2onDisk` computes its disk harmonic transform in float64, so the MPS row is not precision-matched with the others.

Table 7: **Every module runs in under a millisecond on every device tested, except $\mathrm{SO}(3)$ on S^2 on CPU.** Selective forward pass (ms, median), batch = 16, inference mode. Groups and sizes as in Table 3 (C_{128}, $C_{32}\times C_{32}$, D_{32}, O, and the disk and S^2 at $L{=}16$), except that S^2 is sampled on the 64×128 Spherical MNIST grid instead of 34×68. On the H100, both grids take 0.20 ms. This script and its timing loop differ from those of Table 3, so the H100 row differs from that table's Inf. column by up to 0.01 ms.

Device	Backend	Threads	C_n	$\mathbb{T}^2$	D_n	Disk	S^2	O
H100 80 GB	CUDA 12.8	–	0.12	0.07	0.48	0.21	0.20	0.60
A100 80 GB PCIe	CUDA 13.0	–	0.16	0.10	0.64	0.31	0.26	0.82
RTX 4090 24 GB	CUDA 12.8	–	0.20	0.12	0.76	0.34	0.18	0.92
Apple M5 Pro	MPS (fp32)	–	0.08	0.07	0.30	0.69	0.82	0.41
Intel Xeon 8480+	CPU	1	0.05	0.18	0.29	0.16	27.8	0.25
		8	0.08	0.14	0.32	0.13	7.36	0.28
AMD EPYC 7763	CPU	1	0.06	0.22	0.37	0.16	32.2	0.35
		8	0.08	0.17	0.45	0.19	5.46	0.45
AMD EPYC 7K62	CPU	1	0.08	0.26	0.44	0.18	45.4	0.44
		8	0.10	0.30	0.51	0.23	13.9	0.55
Apple M5 Pro	CPU	1	0.02	0.10	0.10	0.04	5.75	0.11
		8	0.02	0.15	0.14	0.07	2.22	0.12

C G-Bispectral Pooling: Extended Results

C.1 Invariant maps in G-CNNs (Figure 3)

Protocol. The rotated test averages the metric over the test set transformed by every element of G, identity included. On OrganMNIST3D the 24 octahedral rotations permute voxels exactly, so G-invariant models score identically on the canonical and rotated test sets. On PCam, the 45° rotations are bilinearly interpolated, so invariance is exact only at multiples of 90°. Aug. CNN is a non-equivariant CNN (3D for OrganMNIST3D) trained with augmentation over G, so the rotated test is in-distribution for it. All other models are G-equivariant, trained on canonical data only, and end in an invariant map over the group dimension: a maximum (Max pool, and the NormReLU, Gated, and Fourier-ELU models, named after their backbone nonlinearity), the ℓ_2 norm (Norm pool, an incomplete invariant), or the selective G-bispectrum (complete). Max pool and Bispectrum on OrganMNIST3D, and Norm pool and Bispectrum on PCam, share a ReLU backbone and differ only in the invariant map. OrganMNIST3D's Norm pool uses NormReLU.

Held fixed. Left: full training set, at the smallest width (OrganMNIST3D) or $\sim$100K-parameter configurations (PCam). Middle: full training set (OrganMNIST3D) or $n = 12.5$K (PCam). Right: smallest width (OrganMNIST3D) or $\sim$100K parameters (PCam). The label "all" is the full training set, and the PCam axis is broken before it (262K examples). The panels of a row share the y-axis.

Statistics. Mean $\pm$ sample standard deviation over 10 seeds (OrganMNIST3D) and 3 seeds (PCam, 5 at $n \leq 2.5$K). p-values are two-sided Welch t-tests, uncorrected for multiplicity. The 46 comparisons are 25 on PCam (5 training-set sizes, 5 baselines) and 21 on OrganMNIST3D (4 training subsets at the smallest width and the full training set at 3 widths, 3 baselines). The PCam width sweep is not tested, because the methods' parameter counts differ at each growth rate. On OrganMNIST3D, bispectral pooling beats max pooling by 5.0 pp ($p < 10^{-3}$). Its smallest model (463K parameters) beats max pooling's largest (6.0M) by 2.6 pp ($p < 10^{-3}$), and at $n = 500$ it beats max pooling at $n = 971$ by 3.6 pp ($p = 0.0017$). On PCam, no model is significantly better than bispectral pooling at any training-set size. The closest is Gated on the full training set ($+0.016$ AUC, $p = 0.093$). With 3 seeds per arm and PCam's seed spread, these tests have 80% power only for gaps of about 0.025 AUC, so "on par" means no difference at that resolution.

C.2 Model variants

PCam. All equivariant models share a C_8-equivariant DenseNet [Huang et al., 2017] backbone with block configuration $(4, 4, 4)$, 24 initial channels, and compression factor 0.5, following the architecture of Veeling et al. [2018] but replacing the final invariant map. Six model variants are evaluated:

- **Aug. CNN**: non-equivariant DenseNet trained with augmentation over C_8 and global average pooling. No group dimension, so $\sim 8\times$ fewer parameters at matched growth rate.
- **NormReLU**, **Gated**, **Fourier-ELU**: C_8-equivariant backbones with norm-based, gated (doubling internal channels) or Fourier-space ELU nonlinearities, each followed by max pooling over the group dimension.
- **Norm pool**: ReLU backbone, ℓ_2 norm over the group dimension.
- **Bispectrum**: ReLU backbone, selective G-bispectrum via `CnonCn`$(n{=}8)$ as the terminal invariant map.

The data-efficiency and full-data panels use growth rates matched to $\sim$100K parameters. The parameter-efficiency panel sweeps the DenseNet growth rate $k \in \{3, 4, 6, 8, 12\}$ for equivariant models and $k \in \{6, 12, 20, 30, 35\}$ for Aug. CNN at $n = 12.5$K, from $\sim$30K to $\sim$1.6M parameters.

OrganMNIST3D. All equivariant models share an O-equivariant 3D ResNet whose kernel rotations are exact voxel permutations on the 3^3 grid. The channel widths $(4, 8)$, $(8, 16)$, and $(16, 32)$ give 463K, 1.7M, and 6.3M parameters with the bispectral head (172 `OctaonOcta` coefficients per channel) and 374K, 1.5M, and 6.0M with max pooling. Aug. CNN is a non-equivariant 3D CNN (16K, 63K, and 251K parameters) trained with one of the 24 octahedral rotations per sample. Norm pool uses a NormReLU backbone. Training-set sizes are $n \in \{50, 100, 250, 500, 971\}$ at width $(4, 8)$.

C.3 Pooling head

The bispectral head computes $\mathrm{ReLU}(\mathrm{proj}(\mathrm{signlog}(\beta)))$, with signlog the signed logarithm of the real and imaginary parts and proj the learned 1×1 projection. Placing the ReLU first, $\mathrm{proj}(\mathrm{ReLU}(\mathrm{signlog}(\beta)))$, zeroes every negative G-bispectral coefficient before the projection. Table 8 compares the two heads and two variants that append the power spectrum. The head was chosen on validation AUC alone. Spherical MNIST does not use this head. On O the truncated invariant keeps full Jacobian rank (200 of 200 random inputs) yet costs $\approx$5 pp, while on C_8 it is rank-deficient (24 of 200) yet costs nothing measurable, so Jacobian rank does not explain the gap. We suspect that the surviving subset changes with every input, so the fixed projection reads different coordinates at every voxel, and the zeroed entries pass no gradient. The rule we adopt is to apply only fixed injective maps (real–imaginary split, signed log) before the first learned layer. Because we compared four heads for the bispectrum but only one per baseline, Table 8 also reports the ReLU-first head. Norm pooling has no truncated variant, because its outputs are non-negative and a ReLU leaves them unchanged.

Table 8: **Moving the ReLU after the projection adds 4–5 pp on OrganMNIST3D at full data.** Test accuracy (OrganMNIST3D, %) or AUC (PCam) at the best-validation checkpoint, mean $\pm$ std over 3 seeds. The ReLU-first head applies the ReLU to the signed-log bispectrum before the 1×1 projection, which zeroes every negative coefficient. The head we use (bold) was chosen on validation AUC alone, by mean rank over the five settings. It makes no significant difference at n=100 or on PCam. p: Welch test against the ReLU-first head.

	OrganMNIST3D (%)			PCam (AUC)	
Head	6.3M	463K	463K, n=100	n=100	n=12.5K
ReLU, then projection	73.9±1.5	73.6±2.0	61.0±0.9	.812±.024	.939±.011
Projection, then ReLU	**77.9±1.9**	**78.6±0.3**	**61.1±2.6**	**.824±.021**	**.936±.007**
p vs. ReLU first	0.045	0.049	0.93	0.55	0.73
+ power spectrum	78.1±1.2	76.9±2.4	62.4±2.7	.813±.031	.940±.008
+ power spectrum, cube root	77.5±0.7	79.0±3.7	62.7±2.8	.821±.003	.936±.005
Max pool	76.0±2.1	73.1±0.6	63.1±1.8	–	–
Norm pool	68.2±3.7	63.3±2.7	57.9±3.0	.802±.038	.933±.011
Aug. CNN	–	–	–	.813±.042	.928±.009

C.4 Published baselines

Table 9 places our results in the context of published methods on PCam. Our models are trained under one fixed protocol for every method rather than tuned per method. At full data and $\sim$100K parameters, bispectral pooling reaches a test AUC of 0.933.

Table 9: Published baselines on PatchCamelyon from Veeling et al. [2018], trained on 100% data.

Method	Reference	Params	ACC	AUC
P4M-DenseNet (D_4)	Veeling et al. [2018]	119K	0.898	0.963
P4M-DenseNet M	Veeling et al. [2018]	19K	0.893	0.958
DenseNet (augmented)	Veeling et al. [2018]	128K	0.881	0.951
DenseNet (no aug.)	Veeling et al. [2018]	128K	0.876	0.955

Table 10 situates our OrganMNIST3D results against published methods on the same benchmark. Our equivariant models (0.37M–6.3M parameters) share one plain backbone to isolate the pooling mechanism, and bispectral pooling reaches 78.5% at 0.46M parameters. Equivariant models of similar or smaller size (ILPOResNet-50, EquiLoPO) reach 86.6–87.9%, so our absolute accuracy is not competitive. The comparison our experiments support is between pooling layers on a shared backbone.

Table 10: Published baselines on OrganMNIST3D for context.

Method	Reference	Params	ACC	AUC
ResNet-18 + 3D	Yang et al. [2023]	33M	0.907	0.996
ResNet-18 + ACS	Yang et al. [2023]	11M	0.900	0.994
ILPOResNet-50	Zhemchuzhnikov and Grudinin [2024]	38K	0.879	0.992
EquiLoPO (local train.)	Zhemchuzhnikov and Grudinin [2025]	418K	0.866	0.991
SE3MovFrNet	Sangalli et al. [2023]	—	0.745	—
Regular SE(3) conv	Kuipers and Bekkers [2023]	172K	0.698	—

D Experimental Details

D.1 Training protocol

Every run trains for a fixed number of epochs that ends a cosine cycle of SGDR [Loshchilov and Hutter, 2017] ($T_0 = 10$ epochs, $T_{\text{mult}} = 2$, so cycles end at epochs 10, 30, 70, ...) and is evaluated at its best-validation checkpoint. There is no early stopping. An epoch draws at least 10,000 training examples, cycling small training sets, so low-data runs take as many optimizer steps per epoch as larger ones. Table 11 lists the remaining settings.

Table 11: Training hyperparameters. All experiments use AdamW [Loshchilov and Hutter, 2019] with weight decay 10^{-4} and the SGDR schedule above.

	PCam	OrganMNIST3D	Sph. MNIST
Learning rate	10^{-3}	10^{-3}	tuned†
Batch size	64	16	tuned†
Epochs	30 (10 at full data)	70	30
Checkpoint	best val. AUC	best val. accuracy	best val. accuracy
Min. examples per epoch	10,000	10,000	10,000
Seeds	3 (5 at $n \leq 2.5$K)	10	5
Mixed precision	Yes	No	No

†Selected on full-data validation accuracy and reused at every n. Bispectrum uses batch 32, lr 10^{-3} (from batch $\{32, 256\}\times$ lr $\{1, 3, 5\} \times 10^{-3}$). S2CNN+BN uses batch 32, lr 5×10^{-3} (from batch $\{32, 256\}\times$ lr $\{1, 5\} \times 10^{-3}$). The tensor product uses 24 channels, $\ell_{\max} = 6$, lr 3×10^{-3} (from 8 configurations of channels, $\ell_{\max}$ and lr). The CNN and power spectrum use batch 256, lr 10^{-3}.

D.2 Hardware

Training ran on three machines: $8\times$ A100 (40 GB), $2\times$ A100 (80 GB), and one H100 (80 GB). Spherical MNIST latencies in Figure 4 are measured on an idle H100.

D.3 Data preprocessing

PCam. PatchCamelyon [Veeling et al., 2018] consists of 327,680 RGB patches of size 96×96 pixels extracted from whole-slide images of lymph node sections [Ehteshami Bejnordi et al., 2017]. We use the standard train/val/test split. Images are loaded from HDF5 files and normalized to $[0, 1]$. Aug. CNN is trained with augmentation over C_8. All equivariant models are trained without augmentation. Training-set sizes $n \in \{100, 500, 2.5\text{K}, 12.5\text{K}\}$ are uniform random subsets of the 262K training patches, drawn per seed.

OrganMNIST3D. OrganMNIST3D [Yang et al., 2023] contains 1,742 abdominal CT organ volumes of size $28 \times 28 \times 28$ voxels with 11 organ classes (train/val/test = 971/161/610, as distributed by `medmnist` 3.0.2). Volumes are normalized to $[0, 1]$. Aug. CNN is trained with one of the 24 octahedral rotations per sample. Equivariant models are trained without augmentation. Kernel rotations are implemented as exact voxel permutations on the 3^3 grid, avoiding interpolation artifacts. Subsets with $n < 971$ are uniform random subsets, drawn per seed.

Spherical MNIST. MNIST digits are projected onto the sphere via stereographic projection following Cohen et al. [2018] and sampled on a 64×128 equiangular grid. The augmented selective spherical bispectrum is computed at L=15, producing 380 invariants, stored as complex numbers.

The model input is obtained by applying $\operatorname{sign}(x)\log(1+|x|)$ separately to the real and imaginary parts of each coefficient, giving 760 real features. These features are fed to a 3-layer MLP: Linear(760, 256)–BN–ReLU–Linear(256, 128)–BN–ReLU–Linear(128, 10). The power spectrum MLP uses the same structure with hidden width 128: Linear(16, 128)–BN–ReLU–Linear(128, 64)–BN–ReLU–Linear(64, 10). The standard CNN has 4 convolutional layers (16–32–64–64 channels) followed by adaptive average pooling and a Linear(1024, 128)–ReLU–Linear(128, 10) classifier. S2CNN+BN is the spherical CNN of Cohen et al. [2018] (features 20 and 40, bandwidths 10 and 6) with batch normalization after each convolution, 155K parameters. The tensor-product network is an $\mathrm{SO}(3)$-equivariant Clebsch–Gordan network with 24 channels per degree up to $\ell_{\max}=6$, 386K parameters. Latency is the median end-to-end forward time, from the spherical grid to the logits, in float32 at batch 16 on an idle H100. With the published settings (no BatchNorm), S2CNN reaches 80.7% NR/R in our pipeline, against 94% reported by Cohen et al. [2018].

D.4 Statistics

All results report the mean $\pm$ sample standard deviation over seeds, evaluated at the best-validation checkpoint. Comparisons are two-sided Welch t-tests. The pooling comparisons (Figure 3) are reported uncorrected, and the 10 Spherical MNIST data-efficiency comparisons (Figure 4) are Holm-corrected jointly. The rotated test for C_8 and O averages over every group element. For $\mathrm{SO}(3)$, it applies one random rotation per test digit.

D.5 Spherical MNIST: capacity of the power spectrum

The power spectrum is a weak baseline at any capacity. Table 12 sweeps the MLP width on both invariants at the default training settings (batch 256, lr 10^{-3}, 30 epochs): the power spectrum saturates near 80% at 550K parameters, below the bispectrum at 25K.

Table 12: Rotated-test accuracy on Spherical MNIST (NR/R, %) against MLP size, mean $\pm$ std over 5 seeds at the default training settings.

Power spectrum (16 features)		Bispectrum (760 features)	
Params	NR/R	Params	NR/R
3.7K	75.79 ± 0.19	25K	94.05 ± 0.23
11K	77.72 ± 0.17	51K	94.90 ± 0.29
39K	78.93 ± 0.17	107K	95.29 ± 0.14
144K	79.51 ± 0.17	230K	95.52 ± 0.08
550K	80.08 ± 0.20	525K	95.74 ± 0.13

D.6 Spherical MNIST: ablation of the augmented entries

Table 13 builds the augmented selective spherical bispectrum up entry type by entry type. Only the Clebsch–Gordan power spectra measurably help accuracy. The self-couplings do not change accuracy and are included for the Jacobian rank. The classifier uses all 760 real features.

Table 13: **Of the augmentation steps, only the Clebsch–Gordan power spectra measurably help.** Spherical MNIST accuracy (%) as the augmented selective spherical bispectrum at $L{=}15$ is built up. Each row adds entries to the one above, with the MLP width fixed and the other entries masked to zero. Mean $\pm$ std over 5 seeds, at the default training settings (batch 256, learning rate 10^{-3}, 30 epochs), whereas Figure 4 uses tuned settings. p: Welch test of NR/R against the row above.

Entries	Real features	NR/NR	NR/R	p
Base bispectral entries	480	95.20±0.09	95.18±0.11	–
+ even self-couplings $\beta_{\ell,\ell,\ell'}$	584	95.23±0.10	95.19±0.08	0.87
+ Clebsch–Gordan power spectra	738	95.50±0.05	95.51±0.05	$< 10^{-3}$
+ degree power, other self-couplings (full)	760	95.58±0.10	95.58±0.08	0.14

D.7 Recovering signals from the augmented selective spherical bispectrum (Figure 2)

Protocol. Signals are band-limited to $L{=}15$ and sampled on a $128{\times}256$ grid. The augmented selective spherical bispectrum has 380 coefficients. Targets are 8 Spherical MNIST digits, each

in its original and one random orientation, and 16 random band-limited signals. Gradient descent on the image starts from noise (4 restarts, 8,000 Adam steps, projected onto the band limit). The degree-by-degree solver works on the 256 real spherical-harmonic coefficients in double precision: Levenberg–Marquardt with the exact Jacobian fits degrees 0–4 from random starts, then adds one degree at a time and refits all degrees so far, keeping up to 12 candidates. Each result is rotated onto the target before measuring the relative image error. The solver lives in `experiments/spherical_mnist_reconstruction/degree_marching.py`.

Table 14: Reconstruction from the augmented selective spherical bispectrum at $L{=}15$, by method and target set.

	Spherical MNIST	Random signals
Gradient descent from noise		
invariant matched to $< 1\%$	14 / 16	0 / 16
image within 0.3 after alignment	9 / 16	0 / 16
Degree-by-degree solver (Figure 2)		
full bispectrum matched to 10^{-12}	16 / 16	16 / 16
image error after alignment	0.0014–0.0035	0.0056–0.0070
Different-orbit signals with equal invariants found	0	0

Results. Gradient descent recovers no random signal: it stalls at a 7.5% median invariant error. The degree-by-degree solver recovers all 32 targets up to the interpolation error of the alignment. Refitting all degrees after each new one is necessary: holding the lower degrees fixed instead lets the residual drift just past the 10^{-10} acceptance threshold, which lost a "1" at $\ell{=}15$ and a "0" at $\ell{=}10$. The remaining image error comes from the alignment, which rotates the reconstruction by interpolating on the grid. As a check, we rotated each target exactly, through its spherical-harmonic expansion, and aligned it back in the same way. This leaves 0.0014–0.0034 (digits) and 0.0060–0.0076 (random signals), the same range as the reconstructions. No run found two signals in different orbits with equal invariants, consistent with conjectured generic completeness.

D.8 Asset licenses

- **PCam**: CC0 1.0 Universal (public domain).
- **OrganMNIST3D**: CC BY 4.0, loaded with `medmnist` 3.0.2 (Apache-2.0).
- **Spherical MNIST**: generated from MNIST, which is released under CC BY-SA 3.0.
- **Code**: PyTorch 2.11 (BSD-3-Clause), NumPy (BSD-3-Clause), `torch_harmonics` 0.9.2 (BSD-3-Clause), and `e3nn` 0.6.0 (MIT).

E API Example

```
import torch
from bispectrum import CnonCn, OctaonOcta, SO2onDisk

f = torch.randn(16, 8) # 16 signals on C_8
bsp = CnonCn(n=8) # selective by default
beta = bsp(f) # (16, 8) invariant coefficients
f_rec = bsp.invert(beta) # signal recovered from beta

# The same interface for other group actions
beta_o = OctaonOcta()(torch.randn(16, 24)) # signals on O
beta_d = SO2onDisk(L=32)(torch.randn(16, 32, 32)) # images on the disk
```

NeurIPS Paper Checklist

1. **Claims**
   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?
   Answer: [Yes]
   Justification: The abstract and introduction list three contributions (a PyTorch library for seven group actions, the augmented selective spherical bispectrum with $\Theta(L^2)$ coefficients, and an evaluation on PCam, OrganMNIST3D, and Spherical MNIST) that are substantiated in Sections 4–6 and Appendices A–D. Empirical claims are stated at the resolution of the Welch tests in Section 6, and completeness of the augmented selective spherical bispectrum is stated as a conjecture.
   Guidelines:
   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.
2. **Limitations**
   Question: Does the paper discuss the limitations of the work performed by the authors?
   Answer: [Yes]
   Justification: Section 7 has "When not to use the bispectrum" and "Limitations and future work" paragraphs: invariance to one specific group, lower sample efficiency than a learned spherical CNN on S^2, inversion of the augmented selective spherical bispectrum living in experiment code, no formal completeness proof, open numerical-conditioning analysis at large band limits, a single tensor-product baseline (no MACE or wavelet scattering), and an MLP head only on S^2. Section 6.1 and Appendix C.1 state the statistical power of the PCam comparisons. Sections 2 and 3 note the genericity assumption for selective bispectra on finite groups. Appendix C.3 discloses that we compared four pooling heads for the bispectrum but one per baseline.
   Guidelines:
   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an impor-

tant role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**
   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?
   Answer: [Yes]
   Justification: All theoretical results (Propositions 5, 7, 10, and 13, Lemmas 11–12) state their assumptions explicitly. Section 5 gives the construction and Algorithm 1. Appendix A provides the structural background with proofs. Generic completeness is stated as a conjecture. The empirical evidence (full per-degree Jacobian rank and recovery of every test signal at $L{=}15$) is in Section 5 and Appendix D.7.
   Guidelines:
   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**
   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?
   Answer: [Yes]
   Justification: The training protocol and all hyperparameters, including the tuning grids, are in Appendix D (Table 11). Model variants are in Appendix C, and seed counts and statistical tests in Appendix D. The library, training scripts, and figure generators are public at `https://github.com/geometric-intelligence/bispectrum/`.
   Guidelines:
   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to re-

produce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

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

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

   Answer: [Yes]

   Justification: The library and all experiment code (training, sweep scripts, and the scripts that generate every figure and table) are released at https://github.com/geometric-intelligence/bispectrum/. All three datasets (PCam, OrganMNIST3D, Spherical MNIST) are publicly available.

   Guidelines:

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

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: Section 6 describes the experimental setups. Full hyperparameters are in Table 11 (Appendix D), model variants in Appendix C, and data preprocessing and splits in Appendix D.

   Guidelines:

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

7. **Experiment statistical significance**

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

   Answer: [Yes]

   Justification: Figures 3 and 4 report mean $\pm$ sample standard deviation over 10 seeds (OrganMNIST3D), 3–5 seeds (PCam) and 5 seeds (Spherical MNIST). The seed sets the weight initialization, data order, augmentation draws, and, below the full training set, the training subset. Splits and test sets are fixed. Differences are tested with two-sided Welch

t-tests. The Spherical MNIST data-efficiency tests are Holm-corrected, and Appendix C.1 states the power of the PCam tests. The procedure is described in Appendix D.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: Appendix D lists the training hardware (8× A100 40 GB, 2× A100 80 GB and one H100 80 GB). Benchmark timings are in Table 3 and Appendix B, and Spherical MNIST inference latencies in Figure 4.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: The work develops mathematical invariants and a software library evaluated on publicly available datasets. No human subjects, private data, or dual-use concerns are involved.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [N/A]

Justification: This is foundational mathematical and software research on group-theoretic invariants. We do not foresee direct negative societal impacts.

Guidelines:

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

11. **Safeguards**

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

Answer: [N/A]

Justification: The released asset is a deterministic mathematical library for computing group-theoretic invariants. We see no specific misuse risk.

Guidelines:

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

12. **Licenses for existing assets**

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

Answer: [Yes]

Justification: All datasets and code libraries are cited. Datasets: PCam (CC0 1.0), OrganMNIST3D (CC BY 4.0, via `medmnist` 3.0.2), and MNIST (CC BY-SA 3.0), from which Spherical MNIST is generated. Code: PyTorch, NumPy, and `torch_harmonics` (BSD-3-Clause) and `e3nn` (MIT). Licenses and versions are listed in Appendix D.

Guidelines:

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

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

13. **New assets**
    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?
    Answer: [Yes]
    Justification: The library is released under the MIT license with documentation, API reference, and usage examples. Section 4 describes the design and interface. Appendix E provides a minimal code example.
    Guidelines:
    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**
    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?
    Answer: [N/A]
    Justification: This work does not involve crowdsourcing or human subjects.
    Guidelines:
    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**
    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?
    Answer: [N/A]
    Justification: This work does not involve human subjects.
    Guidelines:
    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.

- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

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

    Answer: [N/A]

    Justification: LLMs are not part of the core methodology of this work.

    Guidelines:

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