Tunable Soft Equivariance with Guarantees

Md Ashiqur Rahman[1] Lim Jun Hao[2] Jeremiah Jiang[2] Teck-Yian Lim[2] Raymond A. Yeh[1]
[1]Purdue University [2]DSO National Laboratories

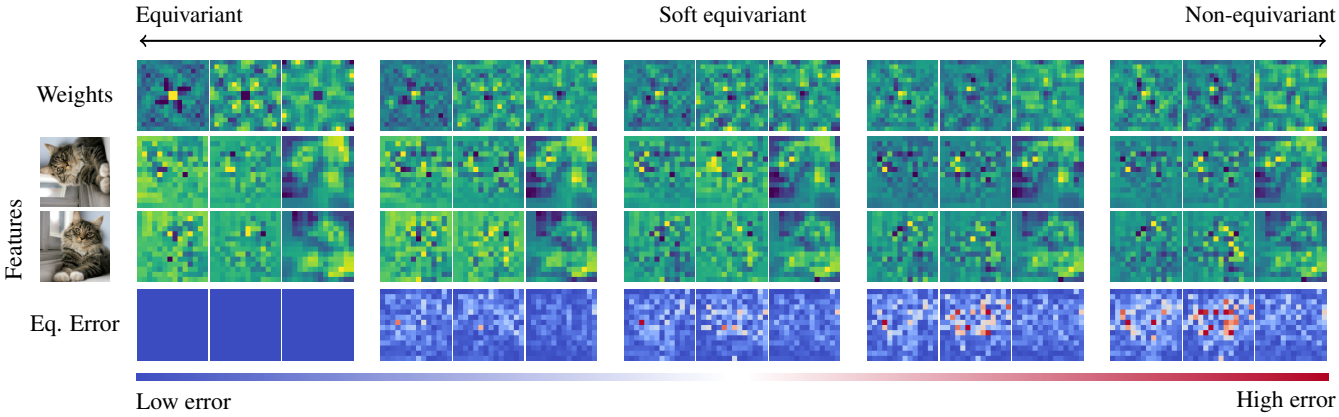

Figure 1. Visualization of the ViT [12] weights with our soft equivariance layer (w.r.t. 90° rotation) under different softness levels, along with the corresponding extracted features and the equivariance errors. Our tunable design allows the layers' weights to transition smoothly from perfectly equivariant to fully non-equivariant behavior in a controlled manner.

Abstract

Equivariance is a fundamental property in computer vision models, yet strict equivariance is rarely satisfied in real-world data, which can limit a model's performance. Controlling the degree of equivariance is therefore desirable. We propose a general framework for constructing soft equivariant models by projecting the model weights into a designed subspace. The method applies to any pre-trained architecture and provides theoretical bounds on the induced equivariance error. Empirically, we demonstrate the effectiveness of our method across multiple pre-trained backbones, including ViT and ResNet, for image classification, semantic segmentation, and human-trajectory prediction. Notably, our approach improves the performance while simultaneously reducing equivariance error on the competitive ImageNet benchmark.

1. Introduction

A model is equivariant to a transformation if applying that transformation to the input results in a predictable transformation at the output. Consider image segmentation: when an object in an image is shifted, the predicted mask is expected to shift by the same amount; this is known as shift equivariance. Although designing models with built-in equivariance has been well studied and shown to be effective in various applications [8, 29, 45, 46, 49, 55, 63, 69], such architectures remain uncommon in mainstream vision systems. In practice, real-world data only approximately satisfies equivariance, and strictly enforcing it can reduce a model's expressiveness.

This led to the development of soft equivariant models, *i.e.*, models that are only approximately equivariant. Common approaches include augmentation [3, 57, 59] and regularization-based methods [14, 28, 56]. However, these techniques do not offer guarantees on a model's equivariance properties after training. Another direction [16, 53, 61, 62] achieves soft equivariance by adding non-equivariant components into equivariant models, providing a way to trade off between expressiveness and equivariance. Nonetheless, these methods still lack guarantees on the resulting equivariance and rely on specialized architectural designs that cannot be easily adapted from off-the-shelf models.

To address these challenges, we propose to construct soft-equivariant models through a generalized notion of "blurring" filters, which can be applied to any pre-trained model. This is inspired by the special case of shift-invariance in convolutional neural networks (CNNs) by Zhang [76], where anti-aliasing (blurring) filters are used to make CNNs more invariant. Our approach extends this idea beyond shift equivariance to other groups and further provides a bound on the equivariance error. This allows us to systematically tune the trade-off between equivariance and expressiveness in a principled manner; see illustration in Fig. 1.

In our experiments, we first demonstrate the tunability of the proposed soft equivariant models on small-scale image classification. We then incorporate the proposed layer into various pre-trained backbones, including ViT [12], DINOv2 [43], ResNet [21], and Segformer [71] for image classification (CIFAR10/100 [31], and ImageNet [11]) and segmentation task (PASCAL VOC [15]). We demonstrate that utilizing our soft equivariance layer further improves the model's performance and reduces equivariance errors. Finally, we go beyond image tasks and evaluate our layers on trajectory prediction [19] and a synthetic $O(5)$-invariant regression problem [17]. **Our main contributions:**

- We introduce a novel framework for constructing soft equivariant layers by restricting the parameters via projections, applicable to any pre-trained model.
- We derive bounds on the equivariance error, which guides the design of the tunable soft equivariant layers, allowing a controllable expressiveness-equivariance trade-off.
- Extensive experiments on three applications (classification, segmentation, and trajectory prediction) and four backbones demonstrate the practicality and effectiveness of the proposed approach.

2. Related Work

Group equivariant architectures. Early work focused on group convolutions [2, 8, 9], while subsequent research extended equivariance to broader architectures, including transformers [4, 32, 60] and graph neural networks [13, 35, 36, 39, 42, 73]. Later developments generalized equivariance beyond rotation and translation [5, 51, 52, 76] to include permutations [20, 50, 75], scaling [48, 58], and reflections [72]. Recent works have explored ways to finetune pre-trained vision models to be equivariant to specific transformations [1, 27]. Despite strong theoretical foundations, strict equivariance assumptions are often misaligned with real-world data and tasks, leading to suboptimal performance [66].

Soft equivariance. Several works address misalignment by relaxing strict equivariance constraints. One line of research softens the rigid structure of group convolutions by introducing additional parameters or input-dependent modulation of the convolution operation [53, 54, 61, 64, 67]. Another line employs loss-based regularization to encourage approximate equivariance [14, 28]. Residual mixing between equivariant and non-equivariant layers has also been explored, where the mixing weights are learned from data [16, 61, 62].

Recently, there have been works aiming to learn the underlying group representations directly from data [40], relating to broader efforts on discovering symmetries in data [74]. These techniques have been adopted to various applications [23, 26, 70]. However, these approaches are dependent on group equivariant architectures [53, 61, 64, 67] or rely on kernels defined directly on the symmetry group [54], thus limiting their applicability to modern large-scale foundation models. In contrast, our framework is architecture-agnostic. While our approach shares conceptual similarities with Finzi et al. [17], it generalizes the idea beyond exact equivariance, integrates seamlessly with modern pre-trained vision models, and provides explicit control over the level of equivariance.

Signal processing (SP). Traditional signal processing establishes a tight connection between low-pass filters, bandlimited subspaces, and shift-invariance: a bandlimited subspace remains bandlimited under shifts [65]. Equivalently, a low-pass (anti-aliasing) filter can be interpreted as a *projection operator* onto a shift-invariant bandlimited space. This projection viewpoint has been generalized to graph signal processing via the graph Laplacian [6, 7], extended to arbitrary discrete groups [47], and recently adapted to image generative models [78].

3. Preliminaries

For readers needing a refresher on the concept of groups, we provide a review in Appendix Sec. B. Here, we will only discuss the most essential background information.

A *group* G is a set with a binary operation that is closed, associative, has an identity element e, and every element has an inverse. The *group representation* $\rho : G \to \mathrm{GL}(V)$ maps group elements to linear transformations on a vector space V, which describes the *action* of the group on V.

Lie algebra and Lie group. A *Lie group* G is a smooth manifold whose multiplication and inversion maps are smooth. The associated *Lie algebra* $\mathfrak{g}$ is the tangent space at the identity, equipped with the Lie bracket $[\cdot,\cdot] : \mathfrak{g} \times \mathfrak{g} \to \mathfrak{g}$ (bilinear, antisymmetric, and satisfying the Jacobi identity). Let $\{A_1, A_2, \ldots, A_N\}$ denote a basis of $\mathfrak{g}$, then any element $A \in \mathfrak{g}$ can be written as $A = \sum_{i=1}^{N} a_i A_i$ with real coefficients a_i.

The *exponential map* $\exp : \mathfrak{g} \to G$ connects the algebra to the group. For a matrix Lie group and $A \in \mathfrak{g}$, this is given by the matrix exponential

$$\exp(tA) = \sum_{n=0}^{\infty} \frac{(tA)^n}{n!}, \tag{1}$$

where the real parameter $t \in \mathbb{R}$ controls the magnitude.

A *group representation* ρ induces an *Lie algebra representation* $d\rho : \mathfrak{g} \to \mathrm{End}(V)$ by differentiation at the identity:

$$d\rho(A) = \frac{d}{dt}\rho(\exp(tA))\Big|_{t=0}, \qquad A \in \mathfrak{g}. \tag{2}$$

Here, $d\rho(A)$ is the *infinitesimal generator* describing the first-order derivative action near the identity on V.

Generalized Taylor expansion. The Taylor approximation of a smooth function $h : \mathbb{R} \to \mathbb{R}$ around $x_0 = 0$ is given as $h(x) = h(0) + h'(0)x + O(x^2)$. This can be extended to functions on a compact, connected Lie group $f : G \to \mathbb{R}$.

Any $g \in G$ can be expressed as $g = \exp(A)$ for $A \in \mathfrak{g}$. The first-order Taylor approximation of f around the identity element e, $f(\exp(\sum_{i=1}^{k} t_i A_i))$ is:

$$f(e) + \sum_{i=1}^{k} t_i \left. \frac{\partial f(\exp(\sum_i t_i A_i))}{\partial t_i} \right|_{t=0} + O(\|A\|_{\mathfrak{g}}^2), \quad (3)$$

where $\|\cdot\|_{\mathfrak{g}}$ denotes the norm on the Lie algebra.

Equivariance. A function $F : \mathcal{X} \to \mathcal{Y}$ is equivariant with respect to a group G if

$$F(\rho_{\mathcal{X}}(g)\boldsymbol{x}) = \rho_{\mathcal{Y}}(g)F(\boldsymbol{x}) \quad \forall g \in G,\ \boldsymbol{x} \in \mathcal{X}, \quad (4)$$

where $\rho_{\mathcal{Y}}$ and $\rho_{\mathcal{X}}$ are representations (group actions) of G on $\mathcal{X}$ and $\mathcal{Y}$, respectively. When $\rho_{\mathcal{Y}}(g) = \rho_{\mathcal{Y}}(e) = \mathbf{I}\ \ \forall g \in G$, then this is a special case called invariance.

4. Towards Soft Equivariant Networks

Soft equivariance can be viewed as a relaxation of the equality constraints (Eq. (4)) into inequality constraints, *e.g.*, the amount of violation measured in norm difference (*a.k.a.* equivariance error):

$$\|F(\rho_{\mathcal{X}}(g)\boldsymbol{x}) - \rho_{\mathcal{Y}}(g)F(\boldsymbol{x})\| \leq \delta. \quad (5)$$

This formulation has been considered in prior work [37, 40]. However, such a definition is sensitive to the scale of the output $F(\boldsymbol{x})$, making it difficult to interpret the significance of the equivariance error δ. Hence, we propose a relative notion of soft equivariance as follows:

Definition 1 (η-Soft Equivariant)**.** *A function F is η-soft equivariant with respect to a group G if it satisfies:*

$$\frac{\|F(\rho_{\mathcal{X}}(g)\boldsymbol{x}) - \rho_{\mathcal{Y}}(g)F(\boldsymbol{x})\|}{\|\boldsymbol{J}_F(\boldsymbol{x})\|_{\mathtt{F}}\|\boldsymbol{x}\|} \leq \eta, \quad \forall g \in G, \boldsymbol{x} \in X. \quad (6)$$

Here, $\boldsymbol{J}_F(\boldsymbol{x})$ is the Jacobian of F at $\boldsymbol{x}$, and $\eta \in \mathbb{R}^+$ is the soft equivariance constant. When $\rho_{\mathcal{Y}}$ is the identity, then we say F is η-soft invariant.

Intuitively, the Jacobian $\|\boldsymbol{J}_F\|$ represents the scaling locally around $\boldsymbol{x}$. Hence, Eq. (6) is measuring equivariance relative to F's own local output variation, making η easier to interpret. To avoid degeneracy, we assume $\|\boldsymbol{J}_F(\boldsymbol{x})\|_{\mathtt{F}} > 0$ and $\|\boldsymbol{x}\| > 0$ for all $\boldsymbol{x}$ in the domain (see Sec. D.1 for details). In the following, we will develop soft equivariant linear layers for both continuous and discrete groups. This is followed by practical considerations and guidance on incorporating them into pre-trained models.

4.1. Soft equivariance for continuous groups

We describe our proposed soft invariant/equivariance linear layers for continuous groups. For readability, we present the layer for a single generator and a single output channel.

Soft invariant fully connected layer. A fully connected layer is defined as $\boldsymbol{y} = F_{\mathtt{FC}}(\boldsymbol{x}; \boldsymbol{w}) \triangleq \boldsymbol{w}^\top \boldsymbol{x}$; here, we consider $\boldsymbol{x}, \boldsymbol{w} \in \mathbb{R}^d$. To achieve η-soft invariance, we impose a structure on $\boldsymbol{w}$ via a "blurring"/ projection operator $\boldsymbol{B}_{\mathtt{inv}} \in \mathbb{R}^{d \times d}$, *i.e.*,

$$\boldsymbol{y} \triangleq (\boldsymbol{B}_{\mathtt{inv}}\theta)^\top \boldsymbol{x}, \text{ with } \boldsymbol{w} \triangleq \boldsymbol{B}_{\mathtt{inv}}\theta \quad (7)$$

where $\theta \in \mathbb{R}^d$ is the learnable parameter of the layer. This projection operation is derived from the Lie algebra representation for a given group G. Let's denote the singular value decomposition (SVD) of the Lie algebra representation as

$$\boldsymbol{A} \triangleq d\rho_{\mathcal{X}}(A) = \boldsymbol{U}\Sigma\boldsymbol{V}^\top, \quad (8)$$

where singular values are sorted, *i.e.*, $0 \leq \sigma_1 \leq \sigma_2 \ldots$ with corresponding left and right singular vectors $\boldsymbol{u}_i$ and $\boldsymbol{v}_i$, respectively. We design the projection operator

$$\boldsymbol{B}_{\mathtt{inv}} \triangleq \sum_{i:\sigma_i < b} \boldsymbol{u}_i \boldsymbol{u}_i^\top, \quad (9)$$

where it filters out the components with a cut-off value $b \in \mathbb{R}^+$. Intuitively, this projection operation removes the directions that are highly affected by the group action.

We now formalize the soft invariance property of the proposed layer in the following claim.

Claim 1. *For any compact and connected Lie group G with injective radius r_G and n_G number of generators, the function $F_{\mathtt{FC}}(\boldsymbol{x}, \boldsymbol{B}_{\mathtt{inv}}\theta)$ is η_b-soft invariant, i.e.,*

$$\frac{\|(\boldsymbol{B}_{\mathtt{inv}}\theta)^\top \boldsymbol{x} - (\boldsymbol{B}_{\mathtt{inv}}\theta)^\top \rho_{\mathcal{X}}(g)\boldsymbol{x}\|}{\|\boldsymbol{J}_{F_{\mathtt{FC}}(\cdot;\boldsymbol{w})}(\boldsymbol{x})\|_{\mathtt{F}}\|\boldsymbol{x}\|} \leq \eta_b, \forall g \in G \quad (10)$$

where $\eta_b = b\sqrt{n_G} r_G + \varepsilon_G$, $b \in \mathbb{R}^+$ is the cut-off value for the projection operator , and ε_G is the residual from the first-order Taylor approximation.

Proof. We use the Taylor expansion in Eq. (3) to express the invariance error in terms of the Lie algebra representation $d\rho$. Next, we relate the contribution of each singular vector of $d\rho(A)$ to the overall invariance error. Finally, we show that $\boldsymbol{B}_{\mathtt{inv}}$ bounds this error by constraining $\boldsymbol{w}$ to lie within a subspace spanned by a selected subset of singular vectors. See proof in Appendix E.1. □

Claim 1 states that the invariance error is dependent on the cut-off value b and other group properties r_G and n_G characterizing the group's size and complexity, *e.g.*, for continuous $2D$ rotation, $r_G = \pi$ and $n_G = 1$.

Remarks. In the case of shift invariance, our construction of $\boldsymbol{B}_{\mathtt{inv}}$ yields a blurring filter whose cut-off frequency is determined by the degree of invariance. While Zhang [76] empirically showed that anti-aliasing (blurring filters) improves shift invariance in CNNs, our general framework

provides a mathematical justification linking bandlimited signals to shift invariance.

Multiple generators. For groups with multiple generators $\{A_i\}_{i=1}^k$, we construct the combined projection operator $\boldsymbol{B}_{\texttt{inv}}$ by calculating the left singular vectors of the concatenated generators

$$\boldsymbol{A} \triangleq [d\rho(A_1) \mid d\rho(A_2) \mid \ \ldots \ \mid d\rho(A_k)]\,. \tag{11}$$

The rest of the design remains the same following Eq. (8) and Eq. (9). See Appendix E.2 for proof.

Soft equivariant fully connected layer. A vector-valued fully connected layer is defined as $\boldsymbol{y} = F_{\texttt{FC}}(\boldsymbol{x}; \boldsymbol{W}) \triangleq \boldsymbol{W}\boldsymbol{x}$. Here, we consider $\boldsymbol{x} \in \mathbb{R}^d$ and $\boldsymbol{W} \in \mathbb{R}^{d' \times d}$. Similar to the invariant layer, to achieve η-soft equivariance, we impose a structure on $\boldsymbol{W}$ via a projection operator $\boldsymbol{B}_{\texttt{eq}} \in \mathbb{R}^{d \cdot d' \times d \cdot d'}$ as follows:

$$\texttt{vec}(\boldsymbol{W}) \triangleq \boldsymbol{B}_{\texttt{eq}}\theta, \tag{12}$$

where $\theta \in \mathbb{R}^{d \cdot d'}$ is the learnable parameter of the layer and $\texttt{vec}$ is the vectorization operator that stacks the columns of a matrix into a vector.

Next, the equivariance constraint in Eq. (4) involving the input and output representations can be consolidated using the Kronecker product [17, 24]. Similarly, we design a matrix $\boldsymbol{L}$ using the Kronecker product of Lie algebra representations, which quantifies deviation from exact equivariance:

$$\boldsymbol{L} \triangleq (d\rho_{\mathcal{X}}(A)^\top \otimes \mathbf{I}_{d'} - \mathbf{I}_d \otimes d\rho_{\mathcal{Y}}(A)) \in \mathbb{R}^{d \cdot d' \times d \cdot d'}. \tag{13}$$

Let the SVD of $\boldsymbol{L}$ be denoted as

$$\boldsymbol{L} = \boldsymbol{U}^{\texttt{L}} \Sigma^{\texttt{L}} {\boldsymbol{V}^{\texttt{L}}}^\top, \tag{14}$$

where the singular values are sorted, *i.e.*, $0 \leq \sigma_1 \leq \sigma_2 \ldots$ with corresponding left and right singular vectors $\boldsymbol{u}_i^{\texttt{L}}$ and $\boldsymbol{v}_i^{\texttt{L}}$, respectively. We propose the projection operator for equivariance as

$$\boldsymbol{B}_{\texttt{eq}} \triangleq \sum_{i:\sigma_i < b} \boldsymbol{v}_i^{\texttt{L}} {\boldsymbol{v}_i^{\texttt{L}}}^\top, \tag{15}$$

where it filters out the components with a cut-off value $b \in \mathbb{R}^+$. We now formally state the soft equivariance property of this layer.

Claim 2. *For any compact and connected Lie group G with injective radius r_G and n_G generators, let $\boldsymbol{W}$ be defined as in (12). Then $F_{\texttt{FC}}(\boldsymbol{x}, \boldsymbol{W})$ is η_b-soft equivariant, i.e.,*

$$\frac{\|\boldsymbol{W}\rho_{\mathcal{X}}(g)\boldsymbol{x} - \rho_{\mathcal{Y}}(g)\boldsymbol{W}\boldsymbol{x}\|}{\|\boldsymbol{J}_{F_{\texttt{FC}}(\cdot;\boldsymbol{W})}(\boldsymbol{x})\|_{\texttt{F}}\,\|\boldsymbol{x}\|} \leq \eta_b, \quad \forall\, g \in G, \tag{16}$$

where $\eta_b = b\sqrt{n_G\, d'}\, r_G + \varepsilon_G$, $b \in \mathbb{R}^+$ is the cut-off value of the projection operator $\boldsymbol{B}_{\texttt{eq}}$, d' is the output dimension, ε_G is the residual from the first-order Taylor approximation.

Proof. We use the Taylor expansion in Eq. (3) to express the equivariance error in terms of the Lie algebra representation $d\rho$. Using properties of the Kronecker product and singular value decomposition, we separate the contribution of each of the singular vectors in the equivariance error. Complete proof in Appendix E.3. □

Efficient design of soft equivariant layer. The computational complexity of performing the SVD on matrix $\boldsymbol{L}$ in Eq. (14) is $O((d \cdot d')^3)$ for a parameter $\theta \in \mathbb{R}^{d \cdot d'}$. While the SVD is precomputed only once per group G before training, it remains computationally expensive for large values of $d \cdot d'$. To address this, we further propose an alternative design using the Schur decomposition [24] with time complexity of $O(\max(d, d')^3)$ for a group whose Lie algebra representations are normal matrices (commute with conjugate transpose), *e.g.*, $2D$ or $3D$ rotations.

Denote the real Schur decomposition of $d\rho_{\mathcal{X}}$ and $d\rho_{\mathcal{Y}}$ as

$$d\rho_{\mathcal{X}} = \boldsymbol{U}_{\mathcal{X}} \boldsymbol{\Sigma}_{\mathcal{X}} \boldsymbol{U}_{\mathcal{X}}^\top \text{ and } d\rho_{\mathcal{Y}} = \boldsymbol{U}_{\mathcal{Y}} \boldsymbol{\Sigma}_{\mathcal{Y}} \boldsymbol{U}_{\mathcal{Y}}^\top. \tag{17}$$

As $d\rho_{\mathcal{X}}$ and $d\rho_{\mathcal{Y}}$ are normal matrices, the $\boldsymbol{\Sigma}_{\mathcal{X}}$ and $\boldsymbol{\Sigma}_{\mathcal{Y}}$ are block diagonal, *i.e.*, $\boldsymbol{\Sigma}_{\mathcal{X}} = \text{diag}(\{\boldsymbol{S}_k\}_{k=1}^p)$ and $\boldsymbol{\Sigma}_{\mathcal{Y}} = \text{diag}(\{\boldsymbol{T}_l\}_{l=1}^q)$, where $\{\boldsymbol{S}_k\}_{k=1}^p$ and $\{\boldsymbol{T}_l\}_{l=1}^q$ are sets of 1×1 or 2×2 Schur form blocks. The $\boldsymbol{U}_{\mathcal{X}}$ and $\boldsymbol{U}_{\mathcal{Y}}$ are orthogonal matrices.

Let θ be the learnable parameter and $\Theta \in \mathbb{R}^{d' \times d} : \mathcal{X} \to \mathcal{Y}$ with $\texttt{vec}(\Theta) \triangleq \theta$. We transform Θ into the Schur basis:

$$\Theta' = \boldsymbol{U}_{\mathcal{Y}}^\top \Theta \boldsymbol{U}_{\mathcal{X}} \tag{18}$$

Lemma 1. *The weight matrix Θ is equivariant if it satisfies the condition*

$$\boldsymbol{\Sigma}_{\mathcal{Y}}\Theta' - \Theta'\boldsymbol{\Sigma}_{\mathcal{X}} = \mathbf{0} \iff \boldsymbol{T}_l\Theta'_{lk} = \Theta'_{lk}\boldsymbol{S}_k \quad \forall l, k, \tag{19}$$

where Θ'_{lk} are blocks of Θ' corresponding to the blocks of $\boldsymbol{\Sigma}_{\mathcal{Y}}$ and $\boldsymbol{\Sigma}_{\mathcal{X}}$ of dimensions $\dim(\boldsymbol{T}_l) \times \dim(\boldsymbol{S}_k)$.

Proof. We apply the change of basis using $\boldsymbol{U}_{\mathcal{X}}$ and $\boldsymbol{U}_{\mathcal{Y}}$ to the equivariance condition, which results in block diagonalization of the Lie algebra representations. Complete proof in Appendix E.4. □

From Lemma 1, we are motivated to leverage the block-wise diagonal structure in our design. Let the magnitude of the maximum eigenvalue of $\boldsymbol{S}_k$ and $\boldsymbol{T}_l$ be $\lambda_{\boldsymbol{S}_k}$ and $\lambda_{\boldsymbol{T}_l}$ respectively. We defined the Schur equivariance projection $\boldsymbol{B}_{\texttt{Schur}}$ as $\boldsymbol{W}' = \boldsymbol{B}_{\texttt{Schur}}[\Theta']$ with block-wise operation as

$$\boldsymbol{W}'_{lk} = \begin{cases} \mathbf{0} & \text{if } \boldsymbol{T}_l \not\simeq \boldsymbol{S}_k, \lambda_{\boldsymbol{S}_k} + \lambda_{\boldsymbol{T}_l} > b, \\ \texttt{Sym}(\Theta'_{lk}) & \text{if } \boldsymbol{T}_l \simeq \boldsymbol{S}_k, \lambda_{\boldsymbol{S}_k} + \lambda_{\boldsymbol{T}_l} > b, \\ \Theta'_{lk} & \text{otherwise} \end{cases} \tag{20}$$

where b is the cut-off value, $\boldsymbol{T}_l \simeq \boldsymbol{S}_k$ implies that $\boldsymbol{T}_l$ and $\boldsymbol{S}_k$ have common eigenvalues ($\boldsymbol{T}_l \not\simeq \boldsymbol{S}_k$ otherwise). For any 2×2 matrix $\Theta'_{lk} = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$, we define

$$\texttt{Sym}(\Theta'_{lk}) = \begin{pmatrix} \frac{a+d}{2} & \frac{b-c}{2} \\ -\frac{b-c}{2} & \frac{a+d}{2} \end{pmatrix}. \tag{21}$$

This form is based on the observation from the condition in Lemma 1 that each block Θ'_{lk} solves a Sylvester equation. When $\boldsymbol{T}_l \not\simeq \boldsymbol{S}_k$ only zero matrix satisfies the condition [24]. When $\boldsymbol{T}_l \simeq \boldsymbol{S}_k$, by Schur's Lemma [18], for 1×1 blocks, any unconstrained scalar satisfies the condition. For 2×2 blocks, the solutions have the form

$$\begin{pmatrix} \alpha & \beta \\ -\beta & \alpha \end{pmatrix}, \quad \alpha, \beta \in \mathbb{R}, \tag{22}$$

i.e., it has the weight symmetric form imposed by $\texttt{Sym}(\cdot)$. Our proposed Schur projection eliminates the weight components that break the equivariance condition, depending on the cut-off value b. We now state the soft equivariance bound.

Claim 3. *For any Lie group G with normal Lie algebra representations with injective radius r_G and n_G generators. Let $\boldsymbol{W} = \boldsymbol{U}_{\mathcal{Y}} \boldsymbol{B}_{\texttt{Schur}}[\Theta'] \boldsymbol{U}_{\mathcal{X}}^{\top}$, then the function $F_{\texttt{FC}}(\boldsymbol{x}, \boldsymbol{W})$ is η_b-soft equivariant*

$$\frac{\|\boldsymbol{W}\rho_{\mathcal{X}}(g)\boldsymbol{x} - \rho_{\mathcal{Y}}(g)\boldsymbol{W}\boldsymbol{x}\|}{\|\boldsymbol{J}_{F_{\texttt{FC}}(\cdot;\boldsymbol{W})}(\boldsymbol{x})\|_{\texttt{F}}\|\boldsymbol{x}\|} \leq \eta_b, \forall g \in G \tag{23}$$

with $\eta_b = b\sqrt{n_G} r_G + \varepsilon_G$, b is the cut-off value of the Schur filter $\boldsymbol{B}_{\texttt{Schur}}$ and ε_G is the residual from the first-order Taylor approximation.

Proof. Complete proof in Appendix E.5. The proof follows the same overall structure as the previous claims. □

4.2. Soft equivariance for discrete groups

The projection design in Sec. 4.1 builds on the Lie group Taylor expansion in Eq. (3) and its Lie algebra representation, which do not apply to discrete groups. We extend this formulation by introducing a *group forward-difference* operator, the discrete analogue of the Lie-algebra representation, and use it to derive the first-order Taylor expansion for discrete groups.

Taylor approximation for discrete groups. For a finite discrete group G with generating set $\mathbb{S}$ (reviewed in Appendix B), we defined the forward difference operator Δ_s as action of function $f : G \to \mathbb{R}$ along generator $s \in \mathbb{S}$ as

$$\Delta_s f(g) = f(sg) - f(g), \forall g \in G. \tag{24}$$

Using the forward difference operator, we can state the first-order Taylor approximation for discrete groups as follows:

Lemma 2. *Let G be a finite discrete group with generating set $\mathbb{S} = \{s_1, \ldots, s_k\}$, and $f : G \to \mathbb{R}$ be h-Lipschitz with respect to the word metric $d_{\mathbb{S}}$, then the first-order Taylor approximation of f at the identity element e defined as $\hat{f}(g) \triangleq f(e) + \sum_{i=1}^{k} n_{s_i} \Delta_{s_i} f(e)$ satisfies the following point-wise error bound:*

$$\left| f(g) - \hat{f}(g) \right| \leq 2h \cdot d_{\mathbb{S}}(e, g), \tag{25}$$

where n_{s_i} number of occurrence of s_i in the canonical word representation of g.

Proof. The error $|f(g) - \hat{f}(g)|$ decomposes into two terms via the triangle inequality: the global displacement $|f(g) - f(e)|$ and contribution of each generators $\sum_i n_i |\Delta_{s_i} f(e)|$. Each term is then bounded using the h-Lipschitz condition. The complete proof is in Appendix E.6. □

With Taylor approximations for discrete groups, the previously introduced designs can be extended to a discrete group by replacing the Lie algebra representation $d\rho$ with the forward difference operator Δ_s.

4.3. Soft equivariant layers in practice

The proposed soft equivariant/invariant layers can be seamlessly integrated into existing architectures, whether pretrained or trained from scratch, without introducing additional learnable parameters.

Integration into existing architectures. For vision models, we incorporate soft equivariant and invariant layers to improve consistency under spatial transformations. These layers are applied to operations defined on structured grids, *e.g.*, convolution, patch embedding, positional encoding, and fully connected layers over flattened grids. The point-wise non-linearities (*e.g.*, ReLU) are equivariant by design. For other modalities, such as 2D/3D point clouds or geometric inputs (*e.g.*, velocity or flow direction), we apply the soft equivariant and invariant layers to the fully connected layers operating on point features.

Softness control. The degree of softness is controlled by the cut-off value b in the projection operator $\boldsymbol{B}_{\texttt{inv}}$ and $\boldsymbol{B}_{\texttt{eq}}$. A smaller b results in a stronger bias towards equivariance, while a larger b allows for more flexibility at the cost of increased equivariance error. In practice, we treat b as a hyperparameter that can be tuned based on validation performance or specific requirements of the task at hand.

We defer the multi-generator equivariant layer design, smooth cut-off with soft threshold, and additional implementation details to Appendix C.

5. Experiments

We empirically validate the effectiveness of our soft equivariance method. In Sec. 5.1, we demonstrate the ability

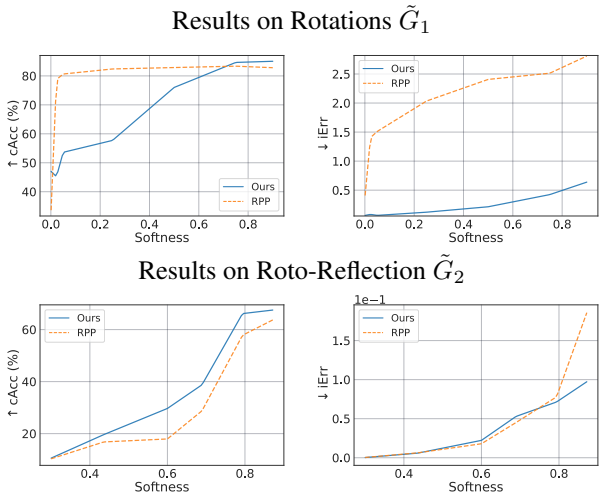

Figure 2. Tunable softness results (cAcc & iErr). Compared to RPP, ours achieves comparable performance with better iErr across two groups.

to control a model's equivariance error. Next, we show that our method can adapt pre-trained vision models to improve their equivariance consistency and performance in image classification (Sec. 5.2) and semantic segmentation (Sec. 5.3). We also showcase an application on human trajectory prediction (Sec. 5.4). For experiments in Sec. 5.2, 5.3 and 5.4, we focus on $2D$ rotation equivariance following prior works [16, 27, 41].

5.1. Validating tunable softness level

Experiment setup. We aim to demonstrate the tunability of our soft equivariance approach. We evaluate on the MNIST classification dataset [33]. All the models here are trained from scratch. Our model and RPP are trained with varying rotation equivariance *'softness'* $\in [0, 1)$, where 'softness' of 1 corresponds to a non-equivariant model and 0 to a fully equivariant model. The 'softness' value controls the fraction of the total number of basis vectors in the projection operator by adjusting the cutoff value b. See Appendix G for details.

Evaluation metric. We report the following metrics:

- Test accuracy (Acc): The standard test set's accuracy.
- Augmented test accuracy (aAcc): Accuracy evaluated on augmented test samples, where augmentations are uniformly drawn from an augmented set $\tilde{G}$, a subset of the group G to which the model is designed to be equivariant.
- Combined accuracy (cAcc): To have a single metric, we report the geometric mean between Acc and aAcc.

$$\text{cAcc} \triangleq (\text{Acc} \cdot \text{aAcc})^{1/2}. \tag{26}$$

- Invariance error (iErr): As the classifiers output probabilities, we use the KL divergence to measure the invariance error, *i.e.*,

$$\text{iErr} \triangleq \mathbb{E}_{\boldsymbol{x}\sim\mathcal{D},g\sim\tilde{G}}\text{KL}\big[F(\boldsymbol{x}) \,\|\, F(\rho(g)\boldsymbol{x})\big]. \tag{27}$$

In this experiment, we consider two groups: rotations and roto-reflections to design the models. For evaluation, the augmented set for rotations is $\tilde{G}_1$, which consists of rotations within $\pm 60°$, while for the roto-reflection group it is $\tilde{G}_2$, which includes the same range of rotations plus reflections.

Baseline. We compare with the residual-based soft equivariance method (RPP) [16]. This approach starts from an equivariant model and introduces a non-equivariant branch through residual connections, roughly doubling the model size. For tunability, we incorporate a scalar parameter on their residual path to balance between equivariance and non-equivariance.

Results. In Fig. 2, we observe that for both the rotation and the roto-reflection group, on different degrees of softness of equivariance, our method has lower invariance error (iErr) while maintaining higher combined accuracy (cAcc). We also observe that RPP has a worse trade-off between accuracy and invariance error compared to ours.

5.2. Image classification

We demonstrate that our proposed method can be effectively incorporated into pre-trained models through fine-tuning.

Setup. We evaluate on CIFAR10/100 [31] and ImageNet-1K [11]. For pre-trained backbones, we use models from PyTorch Image Models [68], including ViT-B/16 [12], DINOv2-Base [25, 43], and ResNet-50 [21]. For CIFAR-10/100, the classification head is replaced to match the number of classes, and the image is interpolated to match the expected input size of the pre-trained model. While for ImageNet-1K, we fine-tune the released models without modification.

Evaluation metric. We follow the same metrics of Acc, aAcc, cAcc, and iErr as described in Sec. 5.1. The main results are reported with the augmentation set $\tilde{G}$ that consists of rotations between $\pm 30°$. We also report on $\pm 15°, 45°$, and $60°$.

Baselines. Alongside the original model (Base) for which we applied our method, we also compare against the canonicalizer (Canon.) baseline [41]. An equivariant network is used to predict the group element that standardizes the image before passing it to the base model. Building on prior work, we employ discrete rotations to construct the models [9, 27, 41]. Note that ResEq [16]'s method builds on top of an equivariant model, so we are unable to adapt a non-equivariant backbone.

CIFAR10/100 results. In Tab. 1, we present the results on CIFAR10 and CIFAR100. We observe that Ours achieves the best overall performance across CIFAR10/100 and all backbones (ViT, DINOv2, ResNet-50). In every setting, Ours consistently improves both aAcc and cAcc, and reduces iErr relative to the Base model. While Canon. achieved a smaller iErr for DINOv2 or ResNet-50, it does so with a

Table 1. **Performance on CIFAR10 and CIFAR100 across various backbones.** Acc(Δ) is top-1 accuracy with Δ vs. Base; aAcc is accuracy under augmentation; cAcc is the combined accuracy; iErr is invariance error ($\times 10^{-2}$, lower is better). We observe that Ours offers a strong performance while maintaining low invariance error.

	Arch.	**ViT** [12]				**DINOv2** [43]				**ResNet-50** [21]			
		Acc(Δ)↑	aAcc↑	cAcc↑	iErr↓	Acc(Δ)↑	aAcc↑	cAcc↑	iErr↓	Acc(Δ)↑	aAcc↑	cAcc↑	iErr↓
CIFAR10	Base	**97.79(00.00)**	96.62	97.20	07.21	98.81(00.00)	97.27	98.04	08.16	96.32(00.00)	91.15	93.70	27.29
	Canon.	93.23(-04.56)	92.62	92.92	10.97	97.16(-01.65)	96.88	97.02	**05.58**	91.11(-05.21)	90.66	90.89	**10.32**
	Ours	**97.79(00.00)**	**96.69**	**97.24**	**07.02**	**98.82(00.01)**	**97.43**	**98.12**	07.03	**96.61(00.29)**	**91.28**	**93.91**	24.20
CIFAR100	Base	86.36(00.00)	84.09	85.22	23.81	91.01(00.00)	85.52	88.22	43.02	83.10(00.00)	73.73	78.28	59.36
	Canon.	78.42(-07.94)	76.83	77.62	24.55	84.39(-06.62)	83.66	84.02	**20.86**	72.90(-10.20)	72.11	72.50	**19.07**
	Ours	**86.60(00.24)**	**84.13**	**85.36**	**23.80**	**91.03(00.02)**	**86.81**	**88.90**	35.17	**83.30(00.20)**	**74.27**	**78.66**	58.50

Table 2. **ImageNet results with various backbones.** Acc(Δ) is top-1 accuracy with Δ vs. Base; aAcc is accuracy under augmentation; cAcc is the combined accuracy; iErr is invariance error ($\times 10^{-2}$, lower is better).

Arch.	**ViT** [12]				**DINOv2** [43]				**ResNet-50** [21]			
	Acc(Δ)↑	aAcc↑	cAcc↑	iErr↓	Acc(Δ)↑	aAcc↑	cAcc↑	iErr↓	Acc(Δ)↑	aAcc↑	cAcc↑	iErr↓
Base	81.67(00.00)	77.29	79.40	00.36	84.27(00.00)	82.82	83.52	00.13	77.91(00.00)	75.12	76.48	00.24
Canon.	76.51(-05.16)	75.81	76.15	**00.15**	83.22(-01.06)	82.54	82.87	00.07	72.07(-05.84)	70.89	71.46	00.24
Ours	**82.28(00.61)**	**80.56**	**81.40**	**00.15**	**85.31(01.04)**	**84.44**	**84.87**	**00.05**	**77.96(00.06)**	**75.52**	**76.72**	**00.11**

Table 3. **Aug. accuracy (aAcc) with various angles on ImageNet.**

Arch.	**ViT** [12]			**DINOv2** [43]			**ResNet-50** [21]		
	15°	45°	60°	15°	45°	60°	15°	45°	60°
Base	79.04	74.83	72.66	83.63	81.78	80.37	75.77	73.42	70.70
Canon.	75.91	76.01	75.92	82.57	82.57	82.42	70.93	71.01	70.78
Ours	**81.23**	**79.36**	**77.55**	**84.80**	**84.08**	**82.74**	**76.47**	**73.70**	**71.27**

substantial drop, roughly 5 to 9%, in performance. We note that Canon. does not achieve zero invariance error due to boundary effects under rotation. When rotating an image, corner pixels move outside the field of view, and these missing pixels can change the canonicalizer prediction.

ImageNet results. Tab. 2 shows results on ImageNet. As in the CIFAR experiments, we observe that Ours is best across all backbones on ImageNet, both in terms of accuracy and invariance error. Ours achieved the best aAcc and cAcc in every case. At the same time, it matches or improves the invariance error (iErr), tying the best with ViT and outperforming DINOv2 and ResNet. In this case, there is no trade-off between performance and invariance, *i.e.*, Ours improves both simultaneously. Finally, in Tab. 3 we report additional experiments on different ranges of rotation angles. As expected, the performance generally drops when a larger degree of rotation is used. In general, the same trend in Tab. 2 holds for Ours.

5.3. Semantic segmentation

Setup. We report on the PASCAL VOC (2012) dataset [15], which comprises 21 object classes for semantic segmentation. For the pretrained backbones, we choose ViT [12], DINOv2 [43] and SegFormer [71] (trained on the ADE20K dataset [77]). Following the linear probing experiment in DINOv2, we add a linear classification head on top of features extracted from ViT and DINOv2. For SegFormer, we replace the output head to match the number of classes.

Evaluation metrics. We report the standard mean Intersection over Union (mIoU). Analogous to the image classification setting, we report mIoU on the augmented set (aIoU) and the combined mIoU (cIoU). As segmentation models F are equivariant, we report the equivariance error (eErr):

$$\mathbb{E}_{(u,v),\boldsymbol{x},g\sim\tilde{G}}\mathrm{KL}\Big[F(\rho_{\mathcal{X}}(g)\boldsymbol{x})[u,v]\parallel\rho_{\mathcal{Y}}(g)F(\boldsymbol{x})[u,v]\Big], \quad (28)$$

where (u, v) indexes the set of valid pixels, excluding those that go out of bounds after rotation. A lower eErr. indicates the model is more consistent in its prediction under the augmentation. We study $\tilde{G}$ consisting of $\pm 30°$ degree rotations. Note, larger rotation leads to significant boundary issues for segmentation, which are less meaningful to study.

Baselines. Same as in the classification experiment, we compare with the Base model, and with the Canonicalization-based approach [41].

Results. Tab. 4 shows segmentation results. Our method improves aIoU, cIoU, and reduces eErr across all settings. For ViT and SegFormer, there is an increase in mIoU, and a mild drop for DINOv2. In contrast, the Canon. baseline reduces mIoU significantly and worsens consistency. We found that the canonicalizer was unable to effectively predict the rotation. In summary, our method offers a better, or even no, trade-off between performance and equivariance error.

5.4. Human trajectory prediction

Setup. Beyond image-based tasks, we evaluate our method on human trajectory prediction using the ETH [44] and UCY [34] datasets, which together comprise five scenes (ETH, Hotel, Zara1, Zara2, and Univ) with varying crowd

Table 4. **Segmentation Performance on PASCAL VOC [15].** mIoU (Δ) is the mean intersection over union with Δ vs. Base; aIoU is the augmented mIoU; cIoU is the combined mIoU; eErr is equivariance error ($\times 10^{-2}$, lower is better.)

Arch.	**ViT** [12]				**DINOv2** [43]				**SegFormer** [71]			
	mIoU(Δ)↑	aIoU↑	cIoU↑	eErr↓	mIoU(Δ)↑	aIoU↑	cIoU↑	eErr↓	mIoU(Δ)↑	aIoU↑	cIoU↑	eErr↓
Base	73.40(00.00)	70.09	71.73	12.31	**89.57(00.00)**	88.10	88.83	04.06	65.34(00.00)	61.17	63.22	11.03
Canon.	65.36(-08.03)	61.93	63.62	20.39	83.48(-06.09)	82.73	83.11	44.93	57.50(-07.84)	55.23	56.36	24.82
Ours	**74.78(01.38)**	**71.61**	**73.18**	**11.12**	89.48(-00.09)	**88.70**	**89.09**	**03.70**	**66.34(00.99)**	**62.52**	**64.40**	**10.64**

Table 5. **Prediction Error on Human Trajectory Datasets.** cADE and cFDE are combined metrics. eErr is equivariance error in $\times 10^{-2}$.

Scenes	**ETH**			**UNIV**			**ZARA1**			**ZARA2**			**HOTEL**		
	cADE↓	cFDE↓	eErr↓	cADE↓	cFDE↓	eErr↓	cADE↓	cFDE↓	eErr↓	cADE↓	cFDE↓	eErr↓	cADE↓	cFDE↓	eErr↓
Base [19]	4.73	**6.15**	1.68	7.91	8.16	0.73	3.61	4.68	1.14	3.17	3.79	1.17	5.84	6.67	2.50
EqAuto [10]	5.40	7.33	**0.00**	8.16	8.33	**0.00**	3.66	4.98	**0.00**	2.94	3.63	**0.00**	6.16	6.78	**0.00**
Ours	**4.58**	6.23	1.42	**7.85**	**8.07**	0.69	**3.40**	**4.67**	0.39	**2.91**	**3.60**	0.24	**5.69**	**6.26**	0.41

Table 6. **Runtime (s) of Schur vs. SVD to compute the projection operator across input sizes.**

Size	4×4	6×6	8×8	10×10	12×12	14×14
SVD	$1.7{\cdot}10^{-1}$	$5.0{\cdot}10^{-1}$	$1.0{\cdot}10^{1}$	$4.5{\cdot}10^{1}$	$2.8{\cdot}10^{2}$	$8.9{\cdot}10^{2}$
Schur	$\mathbf{4.0{\cdot}10^{-3}}$	$\mathbf{1.0{\cdot}10^{-2}}$	$\mathbf{2{\cdot}10^{-2}}$	$\mathbf{5{\cdot}10^{-2}}$	$\mathbf{1.3{\cdot}10^{-1}}$	$\mathbf{2.5{\cdot}10^{-1}}$

Table 7. **Hard vs. Soft threshold performance for image segmentation on PASCAL VOC [15].**

Hard Threshold				**Soft Threshold**			
mIoU↑	aIoU↑	cIoU↑	eErr↓	mIoU↑	aIoU↑	cIoU↑	eErr↓
73.92	69.70	71.78	11.74	**74.78**	**71.61**	**73.18**	**11.12**

densities. The task predicts the next 10 time steps of pedestrian positions $\boldsymbol{y}$ given the past 10 time steps $\boldsymbol{x}$, where each position is represented by 2D coordinates.

Evaluation metrics. Following prior work [19], we report standard performance metrics of the Average Displacement Error (ADE) and Final Displacement Error (FDE) to quantify performance. As with the other tasks, we report the augmented metrics (aADE, aFDE). For the combined metrics (cADE, cFDE), as these are no longer probabilities, we directly combine the standard and augmented metrics with their average.

Next, for equivariance error, as the output $\boldsymbol{y}$ is 2D coordinates, we use the ℓ_2-loss to capture differences, *i.e.*, eErr is defined as:

$$\mathbb{E}_{\boldsymbol{x}\sim\mathcal{D},g\sim\tilde{G}} \left\| F(\rho_{\mathcal{X}}(g)\boldsymbol{x}) - \rho_{\mathcal{Y}}(g)F(\boldsymbol{x}) \right\|_2, \tag{29}$$

where $\mathcal{D}$ is the test set and $\tilde{G}$ consists of $\pm 30°$ rotations.

Baselines. We adopt the autoregressive transformer model by Giuliari et al. [19] as our base architecture (Base) and apply our proposed method (Ours) on top of it. We also compare against an equivariant baseline (EqAuto), an equivariant autoregressive transformer based on vector neurons [10]. We adopt continuous rotation to design the models.

Results. In Tab. 5, we report the quantitative metrics. Our method outperforms both the standard and equivariant transformer baselines in terms of cADE and cFDE across most scenes. We observe that EqAuto with full equivariance does not necessarily lead to better performance. Our approach achieves the best cADE and cFDE on four of the scenes.

5.5. Ablation studies

Schur decomposition vs. SVD. The runtime comparison between Schur decomposition and SVD is reported in Tab. 6. We observe that the cost of constructing the projection operator via SVD grows rapidly, reaching nearly 15 minutes for an input size of 14×14. In contrast, the Schur-based operator requires less than one second.

Hard threshold vs. smooth cut-off in projection. In Tab. 7, we provide the ablation results comparing the hard vs. smooth thresholding method. The experiment follows the same setup as in the segmentation experiment using the ViT backbone. We observe that using the smooth cut-off in projection leads to improved performance (mIoU, aIoU, cIoU) and lower equivariance error (eErr). We defer additional results and ablations to the Appendix F.

6. Conclusion

We propose a principled framework for designing soft invariant and equivariant layers with tunable and theoretically bounded equivariance error. This enables the creation of a family of layers with controllable softness levels. Our layer can be used to adapt existing non-equivariant pre-trained models to soft-equivariant ones. Extensive experiments on image classification, segmentation, trajectory prediction, and synthetic $O(5)$ invariant tasks demonstrate that our approach improves task performance while reducing equivariance error. Notably, our method achieves gains on the competitive ImageNet benchmark. Overall, our framework provides a practical solution for incorporating soft equivariance into modern vision systems with guarantees.

References

[1] Sourya Basu, Prasanna Sattigeri, Karthikeyan Natesan Ramamurthy, Vijil Chenthamarakshan, Kush R Varshney, Lav R Varshney, and Payel Das. Equi-tuning: Group equivariant fine-tuning of pretrained models. In *Proc. AAAI*, 2023. 2

[2] Erik J Bekkers, Maxime W Lafarge, Mitko Veta, Koen AJ Eppenhof, Josien PW Pluim, and Remco Duits. Roto-translation covariant convolutional networks for medical image analysis. In *Proc. MICCAI*, 2018. 2

[3] Gregory Benton, Marc Finzi, Pavel Izmailov, and Andrew G Wilson. Learning invariances in neural networks from training data. In *Proc. NeurIPS*, 2020. 1

[4] Georg Bökman, David Nordström, and Fredrik Kahl. Flopping for flops: Leveraging equivariance for computational efficiency. *arXiv preprint arXiv:2502.05169*, 2025. 2

[5] Anadi Chaman and Ivan Dokmanic. Truly shift-invariant convolutional neural networks. In *Proc. CVPR*, 2021. 2

[6] Siheng Chen, Aliaksei Sandryhaila, and Jelena Kovačević. Sampling theory for graph signals. In *Proc. ICASSP*, 2015. 2

[7] Siheng Chen, Rohan Varma, Aliaksei Sandryhaila, and Jelena Kovačević. Discrete signal processing on graphs: Sampling theory. *IEEE TSP*, 2015. 2

[8] Taco Cohen and Max Welling. Group equivariant convolutional networks. In *International conference on machine learning*, pages 2990–2999. PMLR, 2016. 1, 2

[9] Taco S Cohen and Max Welling. Steerable CNNs. In *Proc. ICLR*, 2017. 2, 6, 30

[10] Congyue Deng, Or Litany, Yueqi Duan, Adrien Poulenard, Andrea Tagliasacchi, and Leonidas J Guibas. Vector neurons: A general framework for SO(3)-equivariant networks. In *Proc. ICCV*, 2021. 8, 29, 32

[11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *Proc. CVPR*, 2009. 2, 6

[12] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *Proc. ICLR*, 2021. 1, 2, 6, 7, 8, 12, 30

[13] Yuanqi Du, Limei Wang, Dieqiao Feng, Guifeng Wang, Shuiwang Ji, Carla P Gomes, Zhi-Ming Ma, et al. A new perspective on building efficient and expressive 3D equivariant graph neural networks. In *Proc. NeurIPS*, 2023. 2

[14] Ahmed A Elhag, T Konstantin Rusch, Francesco Di Giovanni, and Michael Bronstein. Relaxed equivariance via multitask learning. In *Proc. LOG*, 2025. 1, 2

[15] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The PASCAL visual object classes (VOC) challenge. *IJCV*, 2010. 2, 7, 8, 30

[16] Marc Finzi, Gregory Benton, and Andrew G Wilson. Residual pathway priors for soft equivariance constraints. In *Proc. NeurIPS*, 2021. 1, 2, 6, 29

[17] Marc Finzi, Max Welling, and Andrew Gordon Wilson. A practical method for constructing equivariant multilayer perceptrons for arbitrary matrix groups. In *Proc. ICML*, 2021. 2, 4, 29

[18] William Fulton and Joe Harris. *Representation theory: a first course*. Springer Science & Business Media, 2013. 5

[19] Francesco Giuliari, Irtiza Hasan, Marco Cristani, and Fabio Galasso. Transformer networks for trajectory forecasting. In *Proc. ICPR*, 2021. 2, 8, 29

[20] Jason Hartford, Devon Graham, Kevin Leyton-Brown, and Siamak Ravanbakhsh. Deep models of interactions across sets. In *Proc. ICML*, 2018. 2

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. CVPR*, 2016. 2, 6, 7, 30

[22] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proc. CVPR*, 2022. 30

[23] Elyssa Hofgard, Rui Wang, Robin Walters, and Tess Smidt. Relaxed equivariant graph neural networks. *arXiv preprint arXiv:2407.20471*, 2024. 2

[24] Roger A Horn and Charles R Johnson. *Matrix analysis*. Cambridge university press, 2012. 4, 5, 15

[25] Hugging Face Model Hub. facebook/dinov2-base-imagenet1k-1-layer. https://huggingface.co/facebook/dinov2-base-imagenet1k-1-layer, 2025. Accessed: 2025-11-09. 6, 30

[26] Royina Karegoudra Jayanth, Yinshuang Xu, Ziyun Wang, Evangelos Chatzipantazis, Daniel Gehrig, and Kostas Daniilidis. Eqnio: Subequivariant neural inertial odometry. *arXiv preprint arXiv:2408.06321*, 2024. 2

[27] Sékou-Oumar Kaba, Arnab Kumar Mondal, Yan Zhang, Yoshua Bengio, and Siamak Ravanbakhsh. Equivariance with learned canonicalization functions. In *Proc. ICML*, 2023. 2, 6

[28] Hyunsu Kim, Hyungi Lee, Hongseok Yang, and Juho Lee. Regularizing towards soft equivariance under mixed symmetries. In *Proc. ICML*, 2023. 1, 2

[29] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *Proc. ICLR*, 2017. 1

[30] Wilhelm Klingenberg. *Riemannian geometry*. Walter de Gruyter, 1995. 21

[31] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 2, 6

[32] Soumyabrata Kundu and Risi Kondor. Steerable transformers for volumetric data. *arXiv preprint arXiv:2405.15932*, 2024. 2

[33] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 2002. 6, 30

[34] Alon Lerner, Yiorgos Chrysanthou, and Dani Lischinski. Crowds by example. In *Computer graphics forum*, 2007. 7, 32

[35] Yi-Lun Liao and Tess Smidt. Equiformer: Equivariant graph attention transformer for 3d atomistic graphs. In *Proc. ICLR*, 2023. 2

[36] Iou-Jen Liu, Raymond A Yeh, and Alexander G Schwing. PIC: permutation invariant critic for multi-agent deep reinforcement learning. In *Proc. CORL*, 2020. 2

[37] Andrei Manolache, Luiz FO Chamon, and Mathias Niepert. Learning (approximately) equivariant networks via constrained optimization. *arXiv preprint arXiv:2505.13631*, 2025. 3

[38] Dimitris G Manolakis and Vinay K Ingle. *Applied digital signal processing: theory and practice*. Cambridge university press, 2011. 16

[39] Haggai Maron, Heli Ben-Hamu, Nadav Shamir, and Yaron Lipman. Invariant and equivariant graph networks. In *Proc. ICLR*, 2019. 2

[40] Daniel McNeela. Almost equivariance via lie algebra convolutions. *arXiv preprint arXiv:2310.13164*, 2023. 2, 3

[41] Arnab Kumar Mondal, Siba Smarak Panigrahi, Oumar Kaba, Sai Rajeswar Mudumba, and Siamak Ravanbakhsh. Equivariant adaptation of large pretrained models. In *Proc. NeurIPS*, 2023. 6, 7, 30

[42] Christopher Morris, Gaurav Rattan, Sandra Kiefer, and Siamak Ravanbakhsh. SpeqNets: Sparsity-aware permutation-equivariant graph networks. In *Proc. ICML*, 2022. 2

[43] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. DINOv2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 2, 6, 7, 8, 30

[44] Stefano Pellegrini, Andreas Ess, Konrad Schindler, and Luc Van Gool. You'll never walk alone: Modeling social behavior for multi-target tracking. In *Proc. ICCV*, 2009. 7, 32

[45] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proc. CVPR*, 2017. 1

[46] Md Ashiqur Rahman and Raymond A Yeh. Truly scale-equivariant deep nets with Fourier layers. In *Proc. NeurIPS*, 2024. 1

[47] Md Ashiqur Rahman and Raymond A Yeh. Group downsampling with equivariant anti-aliasing. In *Proc. ICLR*, 2025. 2

[48] Md Ashiqur Rahman, Robert Joseph George, Mogab Elleithy, Daniel Leibovici, Zongyi Li, Boris Bonev, Colin White, Julius Berner, Raymond A. Yeh, Jean Kossaifi, Kamyar Azizzadenesheli, and Anima Anandkumar. Pretraining codomain attention neural operators for solving multiphysics PDEs. In *Proc. NeurIPS*, 2024. 2

[49] Md Ashiqur Rahman, Chiao-An Yang, Michael N Cheng, Lim Jun Hao, Jeremiah Jiang, Teck-Yian Lim, and Raymond A Yeh. Local scale equivariance with latent deep equilibrium canonicalizer. In *Proc. ICCV*, 2025. 1

[50] Siamak Ravanbakhsh, Jeff Schneider, and Barnabas Poczos. Deep learning with sets and point clouds. In *Proc. ICLR workshop*, 2017. 2

[51] Renan A Rojas-Gomez, Teck-Yian Lim, Alex Schwing, Minh Do, and Raymond A Yeh. Learnable polyphase sampling for shift invariant and equivariant convolutional networks. In *Proc. NeurIPS*, 2022. 2

[52] Renan A Rojas-Gomez, Teck-Yian Lim, Minh N Do, and Raymond A Yeh. Making vision transformers truly shift-equivariant. In *Proc. CVPR*, 2024. 2

[53] David W Romero and Suhas Lohit. Learning partial equivariances from data. In *Proc. NeurIPS*, 2022. 1, 2

[54] Ashwin Samudre, Mircea Petrache, Brian D Nord, and Shubhendu Trivedi. Symmetry-based structured matrices for efficient approximately equivariant networks. *arXiv preprint arXiv:2409.11772*, 2024. 2

[55] Arne Schneuing, Charles Harris, Yuanqi Du, Kieran Didi, Arian Jamasb, Ilia Igashov, Weitao Du, Carla Gomes, Tom L Blundell, Pietro Lio, et al. Structure-based drug design with equivariant diffusion models. *Nature Computational Science*, 2024. 1

[56] Patrice Simard, Bernard Victorri, Yann LeCun, and John Denker. Tangent prop-a formalism for specifying selected invariances in an adaptive network. In *Proc. NeurIPS*, 1991. 1

[57] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *Proc. ICLR*, 2015. 1

[58] Ivan Sosnovik, Michał Szmaja, and Arnold Smeulders. Scale-equivariant steerable networks. In *Proc. ICLR*, 2020. 2

[59] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proc. CVPR*, 2015. 1

[60] Nathaniel Thomas, Tess Smidt, Steven Kearnes, Lusann Yang, Li Li, Kai Kohlhoff, and Patrick Riley. Tensor field networks: Rotation-and translation-equivariant neural networks for 3d point clouds. *arXiv preprint arXiv:1802.08219*, 2018. 2

[61] Tycho van der Ouderaa, David W Romero, and Mark van der Wilk. Relaxing equivariance constraints with non-stationary continuous filters. In *Proc. NeurIPS*, 2022. 1, 2

[62] Tycho van der Ouderaa, Alexander Immer, and Mark van der Wilk. Learning layer-wise equivariances automatically using gradients. In *Proc. NeurIPS*, 2023. 1, 2

[63] Elise Van der Pol, Daniel Worrall, Herke van Hoof, Frans Oliehoek, and Max Welling. MDP homomorphic networks: Group symmetries in reinforcement learning. In *Proc. NeurIPS*, 2020. 1

[64] Lars Veefkind and Gabriele Cesa. A probabilistic approach to learning the degree of equivariance in steerable CNNs. *arXiv preprint arXiv:2406.03946*, 2024. 2

[65] Martin Vetterli, Jelena Kovačević, and Vivek K Goyal. *Foundations of signal processing*. Cambridge University Press, 2014. 2, 16

[66] Dian Wang, Xupeng Zhu, Jung Yeon Park, Mingxi Jia, Guanang Su, Robert Platt, and Robin Walters. A general theory of correct, incorrect, and extrinsic equivariance. 2023. 2

[67] Rui Wang, Robin Walters, and Rose Yu. Approximately equivariant networks for imperfectly symmetric dynamics. In *Proc. ICML*, 2022. 2, 29

[68] Ross Wightman. Pytorch image models. `https://github.com/huggingface/pytorch-image-models`, 2019. 6, 30

[69] Daniel Worrall and Max Welling. Deep scale-spaces: Equivariance over scale. In *Proc. NeurIPS*, 2019. 1

[70] Zhiqiang Wu, Yingjie Liu, Hanlin Dong, Xuan Tang, Jian Yang, Bo Jin, Mingsong Chen, and Xian Wei. R2det: Exploring relaxed rotation equivariance in 2d object detection. *arXiv preprint arXiv:2408.11760*, 2024. 2

[71] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. SegFormer: Simple and efficient design for semantic segmentation with transformers. In *Proc. NeurIPS*, 2021. 2, 7, 8, 30

[72] Raymond A Yeh, Yuan-Ting Hu, and Alexander Schwing. Chirality nets for human pose regression. In *Proc. NeurIPS*, 2019. 2

[73] Raymond A Yeh, Alexander G Schwing, Jonathan Huang, and Kevin Murphy. Diverse generation for multi-agent sports games. In *Proc. CVPR*, 2019. 2

[74] Raymond A Yeh, Yuan-Ting Hu, Mark Hasegawa-Johnson, and Alexander Schwing. Equivariance discovery by learned parameter-sharing. In *Proc. AISTATS*, 2022. 2

[75] Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Ruslan R Salakhutdinov, and Alexander J Smola. Deep sets. In *Proc. NeurIPS*, 2017. 2

[76] Richard Zhang. Making convolutional networks shift-invariant again. In *Proc. ICML*, 2019. 1, 2, 3

[77] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ADE20K dataset. *IJCV*, 2019. 7

[78] Yifan Zhou, Zeqi Xiao, Shuai Yang, and Xingang Pan. Alias-free latent diffusion models: Improving fractional shift equivariance of diffusion latent space. In *Proc. CVPR*, 2025. 2