LEARNING COLOR EQUIVARIANT REPRESENTATIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

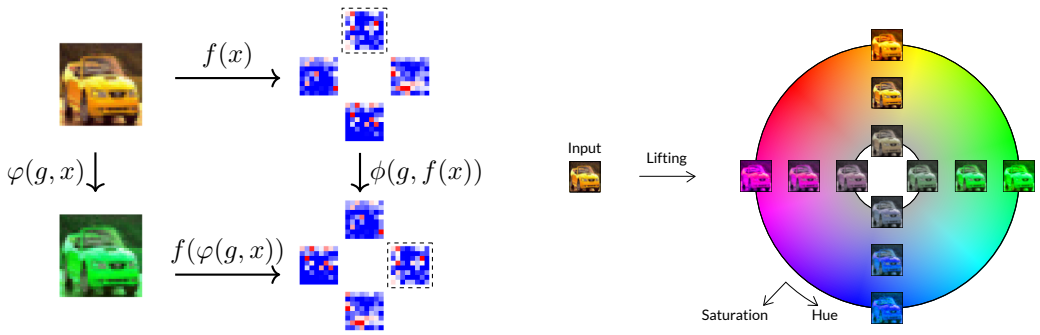
In this paper, we introduce group convolutional neural networks (GCNNs) equivariant to color variation. GCNNs have been designed for a variety of geometric transformations from 2D and 3D rotation groups, to semi-groups such as scale. Despite the improved interpretability, accuracy and generalizability of these architectures, GCNNs have seen limited application in the context of perceptual quantities. Notably, the recent CEConv network uses a GCNN to achieve equivariance to hue transformations by convolving input images with a hue rotated RGB filter. However, this approach leads to invalid RGB values which break equivariance and degrade performance. We resolve these issues with a lifting layer that transforms the input image directly, thereby circumventing the issue of invalid RGB values and improving equivariance error by over three orders of magnitude. Moreover, we extend the notion of color equivariance to include equivariance to saturation shift. Our hue-, saturation-, and color-equivariant networks achieve strong generalization to out-of-distribution perceptual variations and improved sample efficiency over conventional architectures. We demonstrate the utility of our approach on synthetic and real world datasets where we consistently outperform competitive baselines.

1 INTRODUCTION

The tremendous progress of image classification in the last decade can be readily attributed to the development of deep convolutional neural networks [1, 2, 3]. The highly nonlinear mapping and large parameter space does not lend itself to interpretation easily, but even in early networks, representations of geometry and color were observed and recognized for their importance [1, 4]. While a relatively large body of literature has worked to improve the robustness of convolutional neural networks to geometric transformations [5, 6, 7, 8, 9, 10], improving their robustness to perceptual variation has garnered considerably less attention.

A commonly used heuristic for improving network robustness to color variation is to perform mean subtraction and normalization on training set examples. This approach can work well when the training and testing datasets are drawn from the same distribution; however, for data that are collected at different points in time, or with different sensors, this is not likely to be the case. Consider, for example, the case of medical imaging where images of tissue samples collected from different labs (or from the same lab at different points in time) may have different characteristics due to variability in data collection protocols or imaging processes [11]. This variability presents a significant challenge for convolutional neural networks which have been found to be sensitive to color variations even on the level of individual blocks [12]. Moreover, when presented with color perturbations, conventional networks exhibit a significant drop in classification performance [13]. One approach to mitigate the effect of color variation is to ignore color entirely by enforcing color invariance. This can be done by converting input images to grayscale, or enforcing representation similarity across color as is done in [14]. This approach has its own challenges, however, since in many domains, color is an important cue for classification. Another approach is to use dataset augmentation, a technique that can improve robustness in the presence of known transformations [15]. This approach, however, requires extended training times and does not provide improved interpretability.

In this work, we address the challenge of neural network sensitivity to color variation by leveraging the geometric structure of color in a group convolutional neural network. Geometric deep learning has gained strong interest in recent years due to its ability to capture information in interpretable and generalizable representations; by using a group convolutional neural network, we inherit these



(a) Hue equivariance

(b) Lifting layer

Figure 1: **Color equivariant network. (a)** The equivariance of our hue equivariant model is illustrated by the commutativity of the (hue) rotation and neural network mapping. A hue rotation of 90° in the input image space (top-left to bottom-left), results in a feature map rotation at each layer of the network (top-right to bottom-right). Corresponding feature maps are highlighted with a blue border. **(b)** An input image (left) is lifted to the hue-saturation group (right) by shifting its hue and saturation values. For comparison, we illustrate the CEConv [16] lifting layer in Appendix B.1.

characteristics for the context of color variation. We represent color in the hue saturation luminance (HSL) color space, and leverage the insight that the hue, and saturation can be modeled with geometric group structure. Using this approach, our network retains color information in a structured representation throughout the processing pipeline, thereby allowing color information to be used and/or discarded intentionally.

Recently, *Color Equivariant Convolutional Networks* [16], proposed a group convolutional neural network architecture for hue equivariant representation learning. Most notably, CEConv identifies hue transformations with the 2D rotation group. However, the proposed lifting layer introduces invalid RGB values which break equivariance, and degrade performance. To resolve this issue, our lifting layer operates on the input image instead of the network filters, which produces an equivariant descriptor without projection induced artifacts. This seemingly minor change improves equivariance error by more than three orders of magnitude, and stabilizes network performance across discretizations. We also expand the notion of color equivariance to capture variation in saturation. Specifically, we identify variations in saturation with the 1D translation group. With this identification, we propose a group equivariant network where variations in hue and saturation are represented as a geometric transformations.

2 Related Work

Group Convolutional Neural Networks The strong generalization performance of convolutional neural networks on image processing tasks is attributed, in part, to the equivariance of planar convolution to 2D translations. This insight has garnered considerable attention and led to a strong interest in the development of convolutional neural networks equivariant to other symmetry groups [17, 18]. Previous works introduce group convolutional networks for finite and continuous groups. A framework for finite group convolution was developed in [6], and demonstrated for the 2D rotation, and reflection groups. The authors achieve equivariance to specific symmetry groups by convolving input images and feature maps with the group orbit of a learned filter bank. In [19], the authors design for equivariance to the continuous 2D rotation group by constraining filter representations to have circular harmonic structure.

Group convolutional networks have also been designed for use in the context of scale symmetry. In [20], the authors represent input images in log-polar coordinates where 2D rotation and scale transformations present as 2D translations. To navigate the semi-group structure of scale transformations, the authors in [21] approximate the scale space as finite and use dilated convolutions in a group equivariant architecture. The SREN network proposed in [22] designs for equivariance to

the continuous 2D rotation and isotropic scaling group by constraining filter representations to be expressed as the linear combination of a windowed Fourier basis.

Group convolutional networks have also been designed for groups acting in higher dimensions. For 3D rotational symmetry, [23] and [24] use a spherical harmonics based representation with 3D point clouds and spherical image inputs respectively; and [25] introduce an equivariant network for transformations of the Euclidean group in 3D. In [26], the authors introduce a framework for general Lie groups whose exponential map is surjective, and [27] introduces a framework for more general manifolds. By designing for known symmetries in the task, these works provide improved interpretability, training efficiency and generalizability over conventional convolutional networks.

Other group equivariance methods encourage transformation equivariance using a soft penalty term on representation dissimilarity [4, 28]. These methods are separate from the group convolutional neural network literature, and different from what we propose. Our model leverages the geometric structure of hue and saturation to design a hue-saturation equivariant network bringing the benefits of group equivariant networks to perceptual transformations.

Color Invariance Several prior works attempt to mitigate the effect of color variation in image processing tasks, for example, in [29], the authors represent images in a color space where pixel values are invariant to changes in luminance for luminance invariant image segmentation, and in [14], the authors penalize changes in their latent representations due to color variation for color invariant skin lesion identification. While color invariance resolves the effect of color variation, color has been identified as an important cue in neural network representation learning [12, 13] which is discordant with the goal of color invariance. In contrast to these approaches, our work leverages the geometric structure of hue, and saturation to construct a convolutional neural network equivariant to variations in these quantities by design.

Most similar to ours is the model proposed in [16], a hue equivariant network for improved network performance in the presence of color variation. Our color equivariant networks differ from the work proposed in [16] in two important ways. First, we lift the input image instead of the filters of the first layer which circumvents the issue of invalid hue rotations suffered by the network in [16]; and second, we expand the notion of color equivariance by introducing networks that are equivariant to both hue and saturation.

3 PRELIMINARIES

In this section we describe our notation, and review definitions of the group action, equivariance, and group convolution.

Notation. We use f^l to denote the l-th feature map (f^0 to denote an input image), and f^l_j to denote the j-th channel of that feature map. We use ψ^l_i to denote the i-th filter of the l-th layer, and $\psi^l_{i,j}$ to denote the j-th channel of that filter.

Group action. Adapted from [30]. Given a set X and a group G, the action of the group G on X is a function $\varphi : G \times X \to X$ satisfying the following:

1. For all $g, h \in G$ and all $x \in X$
$$\varphi(g, \varphi(h, x)) = \varphi(gh, x) \tag{1}$$
2. For all $x \in X$
$$\varphi(1, x) = x \tag{2}$$
where $1 \in G$ is the identity element of G.

The set X is called a (left) G-set.

Equivariance. Adapted from [30]. Given two G-sets X and Y, and group actions $\varphi : G \times X \to X$ and $\phi : G \times Y \to Y$, a function $f : X \to Y$ is said to be equivariant, if and only if for all $x \in X$, and $g \in G$,
$$f(\varphi(g, x)) = \phi(g, f(x)). \tag{3}$$

When f is invariant to the action of G, the group action ϕ is the identity map and we write

$$f(\varphi(g, x)) = f(x). \tag{4}$$

The equivariance of our network to hue shifts is illustrated in Figure 1a.

Group convolution. In a conventional CNN, the input to convolution layer l, denoted $f^l : \mathbb{Z}^2 \to \mathbb{R}^{K^l}$, is convolved with a set of K^{l+1} filters, denoted $\psi_i^l : \mathbb{Z}^2 \to \mathbb{R}^{K^l}$, where i ranges from 1 to K^{l+1}. The result of the convolution can be written:

$$f_i^{l+1} = [f^l * \psi_i^l](x) = \sum_{y \in \mathbb{Z}^2} \sum_{k=1}^{K^l} f_k^l(y) \psi_{i,k}^l(x - y), \; x \in \mathbb{Z}^2. \tag{5}$$

The convolution of conventional CNNs is equivariant to the action of the group $(\mathbb{Z}^2, +)$, that is, the group formed by summing over the integers. The more general group convolution can be written:

$$[f^l * \psi_i^l](g) = \sum_{h \in G} \sum_{k=1}^{K^l} f_k^l(h) \psi_{i,k}^l(h^{-1} g), \tag{6}$$

and is equivariant to the action of the group G.

4 METHOD

In this section we present our color equivariant network. We begin by presenting definitions for the hue and saturation groups and their group actions, then define the lifting layers, and group convolution layers for our color equivariant networks.

Hue group and group action. In the HSL color space, hue is represented by angular position, and can therefore be identified with the 2D rotation group. As in group convolutional network [6], we consider a finite group representation. Specifically, we identify elements of the discretized hue group, H_N, where the subscript N indicates the cardinality of (i.e. the number of elements in) the group, with those of the cyclic group C_N.

We define the action of hue group on HSL images, $x \in X$ where $x : \mathbb{Z}^2 \to \mathbb{R}^3$, and functions on the discrete hue group, $y \in Y$, where $y : \mathbb{Z}^2 \times H_N \to \mathbb{R}^K$. An element of the hue group acts on an HSL image by the group action $\varphi_h : H_N \times X \to X$, which shifts the hue channel of the image. Concretely, for an HSL image $x \in X$ defined as the concatenation of hue, saturation and luminance channels, i.e., $x = (x_h, x_s, x_l)$, the action of an element h_i of the hue group H_N is given by

$$\varphi_h(h_i, x) = ((x_h + h_i)(\text{mod } c), x_s, x_l), \tag{7}$$

where $(\cdot)\text{mod}(\cdot)$ denotes the modulus operation and is applied to the pixel value which ranges from 0 to c.

An element of the hue group acts on a function on the discrete hue group by the group action $\phi_h : H_N \times Y \to Y$, which "rotates" the function on the group. Concretely, for a function f on the discrete hue group H_N defined as the concatenation of functions $f = (f_1, \ldots, f_N)$, the action of an element h_i in the hue group H_N is given by

$$\phi_h(h_i, f) = (f_{(1+i)(\text{mod } N)}, \ldots, f_{(N+i)(\text{mod } N)}). \tag{8}$$

The action of φ_h on an input image, and ϕ_h on a feature map are shown in Figure 6a. We prove φ_h and ϕ_h are group actions in Appendix A.

Saturation group and group action. We expand the notion of color equivariance proposed in [16] to include equivariance to saturation shifts. In the HSL color space, saturation can be represented by a real number in the interval $[0, 1]$; we introduce two approximations to give the saturation space group structure. First, we observe that there is a bijection between the real numbers and the open interval $(0, 1)$, so we can use the structure of the group $(\mathbb{R}, +)$. Second, we consider a finite subset of the group, a necessary and commonly used practice in both conventional CNNs (recall that the

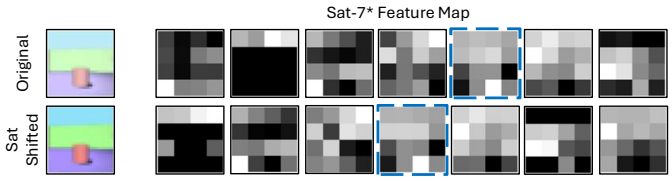


Figure 2: **Saturation equivariant feature maps.** We illustrate the equivariance of our saturation-equivariant model. A saturation shift in the input image space (top-left to bottom-left), results in a feature map translation at each layer of the network (top-right to bottom-right). Corresponding feature maps are highlighted with a blue border.

translation group is infinite) and group-equivariant CNNs such as [21]. With these approximations, we identify elements of the discretized saturation group S_N, with those of the integers with addition, $(\mathbb{Z}, +)$.

We define the action of the saturation group on HSL images, $x \in X$ where $x : \mathbb{Z}^2 \to \mathbb{R}^3$, and functions on the discrete saturation group, $y \in Y$, where $y : \mathbb{Z}^2 \times S_N \to \mathbb{R}^K$. An element of the saturation group acts on an HSL image by the group action $\varphi_s : S_N \times X \to X$, which shifts the saturation channel of the image. Concretely, for an HSL image $x \in X$ defined as the concatenation of hue, saturation and luminance channels, i.e., $x = (x_h, x_s, x_l)$, the action of an element s_i of the saturation group S_N is given by

$$\varphi_s(s_i, x) = (x_h, \min(x_s + s_i, c), x_l). \tag{9}$$

An element of the saturation group acts on a function on the discrete saturation group by the group action $\phi_s : S_N \times Y \to Y$, which "translate" the function on the group. Concretely, for a function f on the discrete saturation group S_N defined as the concatenation of functions $f = (f_1, \ldots, f_N)$, the action of an element s_i in the saturation group S_N is given by

$$\phi_s(s_i, f) = (f_{1+i}, \ldots, f_N, \underbrace{\mathbf{0}, \ldots, \mathbf{0}}_{i}). \tag{10}$$

The action of φ_s on an input image, and ϕ_s on a feature map are shown in Figure 2. We prove φ_s and ϕ_s are group actions in Appendix A. We define the luminance group and group action similarly (see Appendix E.1).

Hue-Saturation group action. We define the action of the hue-saturation group as a composition of the hue and saturation group actions. An element of the hue-saturation group acts on an HSL image by the group action $\varphi_{hs} : H_N \times S_M \times X \to X$, which shifts both the hue and saturation channels of the image. For an HSL image $x \in X$ defined as the concatenation of hue, saturation and luminance channels, i.e., $x = (x_h, x_s, x_l)$, the action of an element (h_i, s_j) of the hue-saturation group $H_N \times S_M$ is given by

$$\varphi_{hs}((h_i, s_j), x) = \varphi_h(h_i, \varphi_s(s_j, x)). \tag{11}$$

An element of the hue-saturation group acts on a function on the discrete saturation group by the group action $\phi_{hs} : H_N \times S_M \times Y \to Y$, which "rotates" and "translates" the function on the group. Concretely, for a function f on the discrete hue-saturation group $H_N \times S_M$ defined as the concatenation of functions $f = (f_{11}, \ldots, f_{1M}, f_{21}, \ldots, f_{NM})$, the action of an element (h_i, s_j) in the hue-saturation group $H_N \times S_M$ is given by

$$\phi_{hs}((h_i, s_j), f) = \phi_h(h_i, \phi_s(s_j, f)). \tag{12}$$

We prove φ_{hs} and ϕ_{hs} are group actions in Appendix A.

Lifting layer. The first layer of a group convolutional neural network "lifts" the input image to the group (see Figure 1b). We can lift an input image to the product space of the image grid $\mathbb{Z}^2$, and discretized hue shifts $H_N = \{h_0, h_1, \ldots, h_N\}$, by convolving with hue shifted filters,

$$[f^0 * \psi_i^0](g_{x,j}) = \sum_{y \in \mathbb{Z}^2} \sum_{k=1}^{K^0} f_k^0(y) h_j \psi_{i,k}^0(x - y). \tag{13}$$

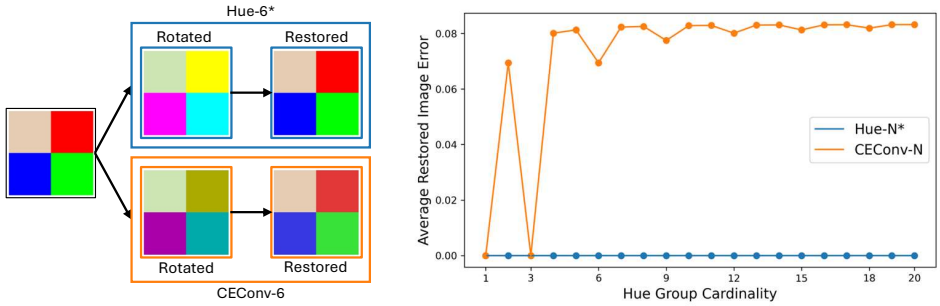


Figure 3: **Impact of cardinality on hue rotation invertibility.** Our lifting layer (blue) operates on HSL input images where each hue rotation is invertible. The lifting layer proposed in CEConv (orange) operates on RGB filters and suffers from invalid hue rotations for all discretizations of the hue group except for $N = 1$ and $N = 3$ (i.e., symmetries of the axis-aligned RGB cube). **(Left)** We show the impact of invalid hue rotations on a four pixel image. We rotate the image $60°$ using our proposed lifting layer (top-left) and the CEConv lifting layer (bottom-left). Subsequently applying a $-60°$ rotation yields an image that is indistinguishable from the original using our approach (top-left), and one with visible artifacts using the CEConv approach (bottom-left). **(Right)** We show the average restored image error for both approaches. Our approach results in a consistently negligible restored image error, however, the CEConv approach results in a restored image error exceeding 7% for all discretizations of the hue group except $N = 1$ and $N = 3$.

Here we denote an element of the product space $g_{x,i}$, where the subscript x references an element of $\mathbb{Z}^2$, and j references the element h_j in H_N. The input image f^0 and lifting filters $\psi_{m,n}$ are functions on $\mathbb{Z}^2$. This is the approach taken in CEonv [16]. The authors shift the hue of a filter in the RGB space by rotating its values about an axis passing through the point $p = (1, 1, 1)$ as shown in Figure 8b in Appendix B.1. As described in [16], this approach results in invalid hue rotations for all discretizations of the hue group that are not symmetries of the axis-aligned RGB cube (i.e., $N = 1$ and $N = 3$). To remedy this, the authors project invalid RGB values to the nearest point on the RGB cube. However, this projection breaks invertibility of the hue shift (see Figure 3) yielding a descriptor that is only approximately hue equivariant.

Lifting to the group can alternatively be achieved by transforming the input image rather than the filters,

$$f^1(g_{x,j}) = \varphi_h(h_j, f^0)(x). \tag{14}$$

We use this approach in our implementation as it circumvents the issue of invalid hue rotations, producing a reliably equivariant descriptor. The lifting layers for the saturation group, and hue-saturation group are constructed analogously.

Equivariance of the lifting layer. We can show that our lifting layer is equivariant to hue shifts. We let $f^1(g_{x,j}) = \varphi_h(h_j, f^0)(x)$ as in equation 14 and show that a hue shift of f^0 results in a hue shift of f^1. In the first step we use the fact that φ_h is a group action (see Appendix A.1), and in the second step we use commutativity of the hue group:

$$\begin{aligned}
\varphi_h(h_j, \varphi_h(h_m, f^0))(x) &= \varphi_h(h_j h_m, f^0)(x) && (15)\\
&= \varphi_h(h_m h_j, f^0)(x) && (16)\\
&= \varphi_h(h_m, \varphi_h(h_j, f^0))(x) && (17)\\
&= \varphi_h(h_m, f^1(g_{x,j})). && (18)
\end{aligned}$$

Equivariance of the saturation, and hue-saturation lifting layers can be shown analogously.

Group convolution layer. For all layers after the first layer, the feature maps f^l are on the product space $\mathbb{Z}^2 \times H_N$. Since hue shifts can only be performed on three dimensional inputs, we leverage the identification of hue shifts with the discrete 2D rotation group, i.e., $H_N \cong C_N$ and interpret the feature maps f^l as functions on the product space $G := \mathbb{Z}^2 \times C_N$. With this interpretation we

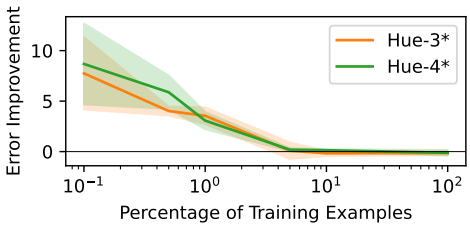


Figure 4: **Model sample efficiency.** We show the error improvement (higher is better) over the Z2CNN baseline as a function of the percentage of training examples used. The advantage of our Hue-N models increases as the percentage of training examples used decreases.

Table 1: **Generalization to global hue-shift.** Classification error on the Hue-shift MNIST dataset is reported. Our hue-equivariant models (Hue-N) perform comparably to baselines on the in-distribution case (A/A), and outperform baselines on the out-of-distribution case (A/B).

Network	A/A	A/B	Params
Z2CNN	**1.54 (0.10)**	57.38 (22.06)	22,130
Hue-3*	1.79 (0.25)	**1.81 (0.29)**	22,658
Hue-4*	1.97 (0.25)	2.08 (0.15)	25,690
CEConv-3	1.79 (0.13)	1.83 (0.19)	28,739
CEConv-4	3.04 (0.27)	3.09 (0.25)	30,539

perform group convolution on G as follows,

$$[f^l * \psi^l_i](g) = \sum_{h \in H} \sum_{k=1}^{K^0} f^l_k(h) \psi^l_{i,k}(g^{-1}h), \tag{19}$$

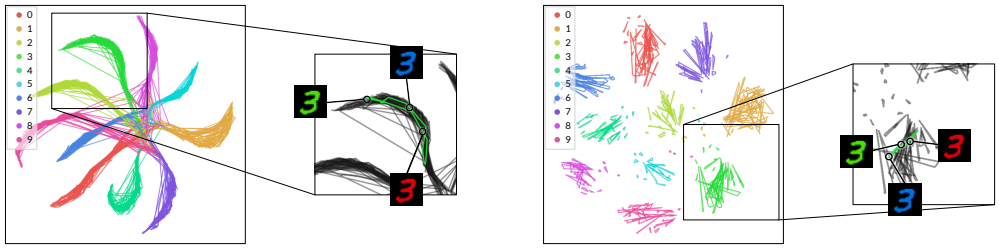
where ψ^l is a function on G. Group convolution on the saturation, and hue-saturation groups are performed analogously.

5 Experiments

In this section, we highlight improved sample efficiency over a conventional CNN baseline, improved hue-equivariance over CEConv [16], the utility of saturation equivariance on the Camelyon17 dataset [31], the utility of luminance equivariance on the small NORB dataset [32] in Appendix E.1, and the utility of our architectures for color based sorting. We additionally demonstrate the strong performance of our model through extensive experiments and against numerous baselines.

5.1 Hue-shift MNIST Classification

We demonstrate improved generalization to global hue-shifts, and higher sample efficiency compared to a conventional CNN model with a similar number of parameters on our Hue-shift MNIST dataset. Our dataset is a variation of MNIST [33] with 60k training examples and 10k test examples classified into one of 10 categories. Additional details are provided in Appendix C.1.



(a) Z2CNN (b) Hue-4

Figure 5: **Hue shift MNIST feature map visualization.** We compare the feature map trajectories of MNIST digits as their hue is varied from 1 to 360°. The color of the trajectory corresponds to the class label. **(a)** tSNE projection of hue shifted feature map trajectories in the Z2CNN baseline model. As the hue of the input changes, the location of the digit in the feature space changes significantly. **(b)** tSNE projection of hue shifted feature map trajectories in our hue-equivariant CNN. In contrast to the Z2CNN baseline, the location of the digit in the feature space changes minimally.

Table 2: **Generalization to local hue-shift.** Classification error on the 3D Shapes dataset is reported. Our model (Hue-N) and CEConv-N achieve improved generalization performance over the conventional CNN model (Z2CNN). Our approach is robust to the choice of hue group discretization, while the performance of CEConv deteriorates when N is not a symmetry of the axis-aligned RGB cube.

Network	A/A	A/B	A/C	Params
Z2CNN	0.00 (0.00)	51.25 (9.59)	26.66 (19.60)	20,192
Hue-3*	0.00 (0.00)	0.03 (0.04)	0.04 (0.03)	19,478
Hue-4*	**0.00 (0.00)**	**0.00 (0.00)**	**0.00 (0.00)**	21,832
CEConv-3	0.00 (0.00)	0.02 (0.03)	0.05 (0.04)	26,441
CEConv-4	0.00 (0.03)	0.60 (0.25)	0.53 (0.43)	30,792

Generalization to global hue-shift. The generalization performance of our hue-equivariant models, Hue-N (N indicates the cardinality of the discrete hue group), and a conventional CNN model, Z2CNN [6], are reported in Table 1. On the in-distribution test case (A/A), the performance of our model and the conventional CNN model are comparable. However, on the out-of-distribution test case (A/B), the performance of our model is preserved, while the performance of the conventional CNN model deteriorates significantly.

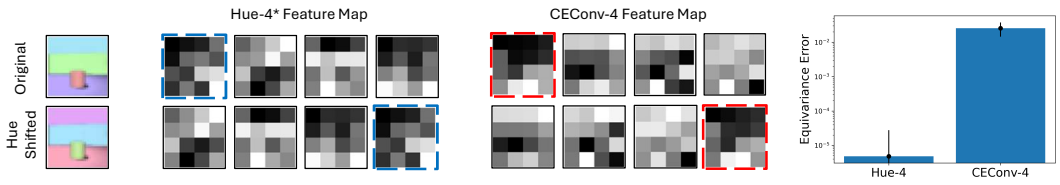
The performance gap on the out-of-distribution test case can be attributed to the difference in internal representation structure. To understand this better, we generate feature representation trajectories in the penultimate layer of our network and the baseline architecture by continuously varying the hue of an input image. We then visualize these trajectories using tSNE [34] projection (see Figure 5). For the baseline architecture, the trajectories of different digits overlap for some hues (Figure 5a), but for our model, the trajectories of different digits are confined to separate clusters (Figure 5b).

Model sample efficiency. The sample efficiency of our hue-equivariant models (Hue-N) and a conventional CNN model (Z2CNN) are reported in Figure 4. The advantage of our hue-equivariant models increases as the percentage of the training set used decreases, illustrating improved sample efficiency [35].

5.2 Hue-shift 3D Shapes Classification

We demonstrate improved generalization to local hue-shifts, and significantly reduced equivariance error compared to CEConv on the 3D Shapes classification dataset [36]. The 3D Shapes dataset consists of RGB images of 3D shapes, where the color of the shape, the floor, and the walls vary across examples, as does the scale and orientation of the shape. There are 48k examples in the training set, and 12k examples in each of the test sets. Additional details are provided in Appendix C.2.

Generalization to local hue-shift. The generalization performance of our hue-equivariant models, Hue-N (N indicates the cardinality of the discrete hue group), a conventional CNN model,



(a) Hue-shifted feature maps (b) Equivariance error

Figure 6: **Hue equivariant feature maps.** (a) We qualitatively compare hue-shifted feature maps produced using our lifting layer and the CEConv lifting layer. Highlighted feature maps obtained using our lifting layer (Hue-4) are qualitatively indistinguishable, while there are visible discrepancies in the feature maps of CEConv-4. (b) We quantitatively compare the equivariance-error of the hue-shifted feature maps for each approach. The equivariance error resulting from our lifting layer (Hue-4) is more than three orders of magnitude lower than the error resulting from CEConv-4.

Table 3: **Generalization to saturation-shift.** Classification error on the Camelyon17 dataset is reported. Our model (Sat-N and Hue-M-Sat-N) achieves improved generalization performance over the conventional CNN model (ResNet50).

Networks	Error	Param
ResNet50	28.91 (7.58)	23.5M
Sat-3*	**16.08 (2.68)**	23.3M
Hue-4-Sat-3*	19.06 (4.92)	23.0M
CEConv-3	28.76 (9.93)	23.1M

Z2CNN [6], and the CEConv-N models are reported in Table 2. Our Hue-4 model outperforms all other models on the in-distribution (A/A), global hue-shift out-of-distribution (A/B) and local hue-shift out-of-distribution (A/C) test sets. The performance of CEConv-4 is significantly worse than all other hue-equivariant models. We attribute the relatively poor generalization performance of the CEConv-4 model to the feature map equivariance error.

Lifting layer equivariance error. The lifting layer equivariance error of our Hue-4 model and the CEConv-4 model are reported in Figure 6b. As discussed in section 4, our lifting layer operates on HSL images and does not induce the invalid hue rotations observed in CEConv. We quantitatively assess how lifting layer invertibility impacts descriptor equivariance following [37]. We compute the equivariance error as the relative equivariance,

$$\text{Equivariance Error} = \frac{|f(\varphi_h(h_i, x)) - \phi_h(h_i, f(x))|}{|f(\varphi_h(h_i, x)) + \phi_h(h_i, f(x))|} \tag{20}$$

and show that the equivariance error of CEConv-4 is more than three orders of magnitude higher than our Hue-4 model (see Figure 6b).

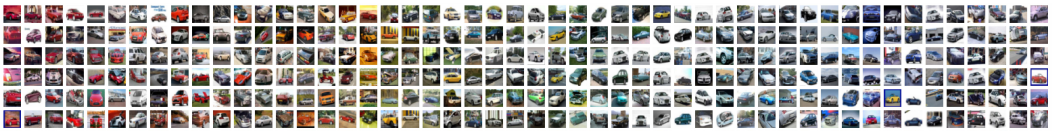
5.3 Camelyon17 Classification: Saturation shift in the wild

We demonstrate improved generalization to saturation-shifts compared to ResNet50 [3] and CEConv-3 on the Camelyon17 classification dataset [31]. The Camelyon17 dataset consist of images of human tissue collected from five different hospitals. Variation in tissue images results from variation in the data collection and processing procedures. The dataset consists of 387,490 examples, 302,436 of which are used for training and 85,054 of which are used for testing. Hospitals 1, 2, and 3 are represented in the training set, and hospital 5 is represented in the testing set. Additional details are provided in Appendix D.3.

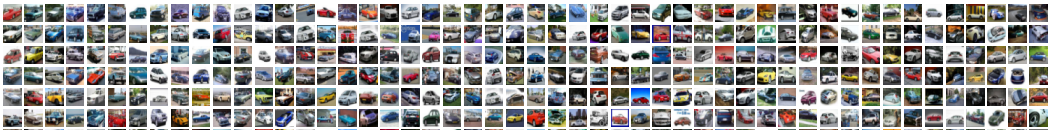
Generalization to saturation-shift. We expand the notion of color equivariance proposed in [16] to capture variation in saturation. The generalization performance of our saturation-equivariant models,

Table 4: **Generalization to color-shift.** Classification error on the Caltech-101, Oxford-IIT Pets, Stanford Cars, CIFAR-10, CIFAR-100, STL-10 datasets are reported. Our models (Hue-N, Sat-N and Hue-M-Sat-N) outperform CEConv-N models on all datasets, and are competitive with ResNet. We use ResNet-aug and Resnet-gray to denote a ResNet architecture trained on a color-jitter augmented dataset and grayscale image dataset respectively.

Network	Caltech 101	CIFAR-10	CIFAR-100	Stanford Cars	Oxford Pets	STL-10
ResNet	32.68 (1.55)	**7.86 (1.14)**	**32.00 (0.63)**	25.41 (0.96)	31.52 (2.05)	18.59 (1.65)
ResNet-gray	33.79 (3.09)	8.45 (0.68)	32.04 (0.66)	24.71 (0.93)	30.38 (0.35)	18.71 (1.47)
ResNet-aug	32.90 (0.82)	8.33 (0.44)	32.27 (0.18)	22.38 (1.65)	30.06 (0.52)	**17.89 (1.48)**
Hue-3*	34.32 (0.34)	8.41 (0.39)	33.28 (0.63)	22.19 (2.20)	31.37 (3.50)	19.82 (0.98)
Hue-4*	**32.23 (1.07)**	8.83 (0.64)	34.70 (0.89)	**20.38 (1.06)**	**27.39 (0.68)**	20.73 (1.11)
Hue-4-Sat-3*	38.14 (1.07)	10.68 (0.78)	33.27 (0.31)	24.79 (3.87)	29.84 (1.34)	20.53 (0.73)
Sat-3*	41.64 (1.40)	9.24 (0.27)	39.33 (0.45)	31.90 (10.03)	36.87 (5.57)	20.71 (1.10)
CEConv-3	34.74 (0.83)	8.86 (0.33)	34.95 (0.44)	23.97 (1.56)	31.08 (2.54)	24.29 (1.31)
CEConv-4	33.52 (0.48)	9.28 (0.24)	35.46 (0.35)	24.08 (0.66)	33.70 (1.50)	21.90 (1.64)



(a) Hue-4



(b) ResNet44

Figure 7: **Color sorting on CIFAR-10.** We show images from CIFAR-10 automobile class ordered by pairwise distance using Hue-4 and ResNet44 feature maps. The structure of the Hue-4 feature maps naturally allow for color based sorting, whereas the ResNet44 feature maps do not. We include visualizations of the entire automobile class sorted using Hue-N and CEConv-N in Appendix B.3.

Resnet50 [3], and CEConv-3 are reported in Table 3. Our saturation- and color-equivariant models outperforms all other models, while CEConv performs comparably to the conventional CNN model.

5.4 Color shift in the wild

We demonstrate improved generalization to global color-shifts compared to ResNet [3] and CEConv on the Caltech-101 [38], Oxford-IIT Pets [39], Stanford Cars [40], STL-10 [41], CIFAR-10 and CIFAR-100 [42]. Additional details are provided in Appendix D.

Generalization to color-shift. The generalization performance of our hue-equivariant models, Hue-N, saturation-equivariant model, Sat-N, color-equivariant models Hue-M-Sat-N, ResNet44 model, and the CEConv-N models are reported in Table 4.

Color sorting on CIFAR-10. While our hue-equivariant model performs on-par with the conventional CNN model on this dataset, the structure of our representation can be leveraged for tasks that are not possible with conventional architectures alone. To illustrate this, we use our model to sort images in the automobile class by hue (see Figure 7a). To produce an hue ordering, we compute the hue-shifted pairwise Euclidean distance between hue group feature maps of instances in the automobile class. Images x_1 and x_2 are close in the hue space if

$$\arg\min_i \|\phi_h(h_i, \Psi(x_1)), \Psi(x_2)\| = 0, \tag{21}$$

where $\Psi(x_j)$ denotes the penultimate feature map for image j and h_i is the i-th element of the hue group. Sorting using the penultimate feature map of the ResNet44 model is shown in Figure 7b.

6 Conclusion

In this paper we address the challenge of learning hue- and saturation-equivariant representations. Leveraging the observation that these perceptual transformations have geometric structure, we propose a group structure for each, and a group convolutional neural network that is equivariant to the transformations by design. By encoding perceptual variation in a finite group structure, we achieve strong generalization performance and sample efficiency and can use learned representations for tasks such as color based sorting.

6.1 Limitations and future work

GCNs are more computationally expensive than their conventional counterparts since they require computation of the filter orbit at each layer. Concretely, the computational cost of GCNNs is approximately equal to that of conventional CNNs with a filter bank size equal to that of the augmented filter bank used in GCNNs [6]. This is compounded with the requirement for a finite filter orbit which can at best approximate a continuous group. Future work will investigate a continuous representation of the group to address both of the aforementioned challenges.

7 REPRODUCIBILITY STATEMENT

Source code will be open-sourced upon acceptance. Proofs for hue and saturation group actions are shown in Appendix A. Proof of equivariance of the proposed lifting layer is shown in Section 4. Datasets, training and implementation details are described in Appendix C.

REFERENCES

[1] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012.

[2] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[4] Karel Lenc and Andrea Vedaldi. Understanding image representations by measuring their equivariance and equivalence. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 991–999, 2015.

[5] Joan Bruna and Stéphane Mallat. Invariant scattering convolution networks. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1872–1886, 2013.

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

[7] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *International conference on machine learning*, pages 1263–1272. PMLR, 2017.

[8] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017.

[9] Geoffrey E Hinton, Sara Sabour, and Nicholas Frosst. Matrix capsules with em routing. In *International conference on learning representations*, 2018.

[10] Samuel Greydanus, Misko Dzamba, and Jason Yosinski. Hamiltonian neural networks. *Advances in neural information processing systems*, 32, 2019.

[11] Mitko Veta, Paul J Van Diest, Mehdi Jiwa, Shaimaa Al-Janabi, and Josien PW Pluim. Mitosis counting in breast cancer: Object-level interobserver agreement and comparison to an automatic method. *PloS one*, 11(8):e0161286, 2016.

[12] Martin Engilberge, Edo Collins, and Sabine Süsstrunk. Color representation in deep neural networks. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 2786–2790. IEEE, 2017.

[13] Kanjar De and Marius Pedersen. Impact of colour on robustness of deep neural networks. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 21–30, 2021.

[14] Arezou Pakzad, Kumar Abhishek, and Ghassan Hamarneh. Circle: Color invariant representation learning for unbiased classification of skin lesions. In *European Conference on Computer Vision*, pages 203–219. Springer, 2022.

[15] Gregory Benton, Marc Finzi, Pavel Izmailov, and Andrew G Wilson. Learning invariances in neural networks from training data. *Advances in neural information processing systems*, 33: 17605–17616, 2020.

[16] Attila Lengyel Ombretta Strafforello Robert-Jan and Bruintjes Alexander Gielisse Jan van Gemert. Color equivariant convolutional networks.

[17] Risi Kondor and Shubhendu Trivedi. On the generalization of equivariance and convolution in neural networks to the action of compact groups. In *International Conference on Machine Learning*, pages 2747–2755. PMLR, 2018.

[18] Taco S Cohen, Mario Geiger, and Maurice Weiler. A general theory of equivariant cnns on homogeneous spaces. *Advances in neural information processing systems*, 32, 2019.

[19] Daniel E Worrall, Stephan J Garbin, Daniyar Turmukhambetov, and Gabriel J Brostow. Harmonic networks: Deep translation and rotation equivariance. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5028–5037, 2017.

[20] Carlos Esteves, Christine Allen-Blanchette, Xiaowei Zhou, and Kostas Daniilidis. Polar transformer networks. *arXiv preprint arXiv:1709.01889*, 2017.

[21] Daniel Worrall and Max Welling. Deep scale-spaces: Equivariance over scale. *Advances in Neural Information Processing Systems*, 32, 2019.

[22] Zikai Sun and Thierry Blu. Empowering networks with scale and rotation equivariance using a similarity convolution. In *The Eleventh International Conference on Learning Representations*, 2022.

[23] 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.

[24] Carlos Esteves, Yinshuang Xu, Christine Allen-Blanchette, and Kostas Daniilidis. Equivariant multi-view networks. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1568–1577, 2019.

[25] Simon Batzner, Albert Musaelian, Lixin Sun, Mario Geiger, Jonathan P Mailoa, Mordechai Kornbluth, Nicola Molinari, Tess E Smidt, and Boris Kozinsky. E (3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nature communications*, 13(1): 2453, 2022.

[26] Marc Finzi, Samuel Stanton, Pavel Izmailov, and Andrew Gordon Wilson. Generalizing convolutional neural networks for equivariance to lie groups on arbitrary continuous data. In *International Conference on Machine Learning*, pages 3165–3176. PMLR, 2020.

[27] Taco Cohen, Maurice Weiler, Berkay Kicanaoglu, and Max Welling. Gauge equivariant convolutional networks and the icosahedral cnn. In *International conference on Machine learning*, pages 1321–1330. PMLR, 2019.

[28] Sharut Gupta, Joshua Robinson, Derek Lim, Soledad Villar, and Stefanie Jegelka. Structuring representation geometry with rotationally equivariant contrastive learning. In *The Twelfth International Conference on Learning Representations*, 2023.

[29] Hamilton Y Chong, Steven J Gortler, and Todd Zickler. A perception-based color space for illumination-invariant image processing. *ACM Transactions on Graphics (TOG)*, 27(3):1–7, 2008.

[30] Jean Gallier and Jocelyn Quaintance. *Differential geometry and lie groups*, volume 12. Springer, 2020.

[31] Peter Bandi, Oscar Geessink, Quirine Manson, Marcory Van Dijk, Maschenka Balkenhol, Meyke Hermsen, Babak Ehteshami Bejnordi, Byungjae Lee, Kyunghyun Paeng, Aoxiao Zhong, et al. From detection of individual metastases to classification of lymph node status at the patient level: the camelyon17 challenge. *IEEE Transactions on Medical Imaging*, 2018.

[32] Yann LeCun, Fu Jie Huang, and Léon Bottou. Learning methods for generic object recognition with invariance to pose and lighting. *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2:II–104 Vol.2, 2004.

[33] Yann LeCun, Corinna Cortes, and CJ Burges. Mnist handwritten digit database. *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, 2, 2010.

[34] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86), 2008.

[35] Bryn Elesedy. Group symmetry in pac learning. In *ICLR 2022 workshop on geometrical and topological representation learning*, 2022.

[36] Chris Burgess and Hyunjik Kim. 3d shapes dataset. https://github.com/deepmind/3dshapes-dataset/, 2018.

[37] Maksim Zhdanov, David Ruhe, Maurice Weiler, Ana Lucic, Johannes Brandstetter, and Patrick Forré. Clifford-steerable convolutional neural networks. *arXiv preprint arXiv:2402.14730*, 2024.

[38] Fei-Fei Li, Marco Andreeto, Marc'Aurelio Ranzato, and Pietro Perona. Caltech 101, Apr 2022.

[39] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and C. V. Jawahar. The oxford-iiit pet dataset. URL `https://www.robots.ox.ac.uk/~vgg/data/pets/`.

[40] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *2013 IEEE International Conference on Computer Vision Workshops*, pages 554–561, 2013. doi: 10.1109/ICCVW.2013.77.

[41] Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 215–223. JMLR Workshop and Conference Proceedings, 2011.

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

[43] Ya Le and Xuan Yang. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7):3, 2015.

[44] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[45] Olivia Wiles, Sven Gowal, Florian Stimberg, Sylvestre Alvise-Rebuffi, Ira Ktena, Krishnamurthy Dvijotham, and Taylan Cemgil. A fine-grained analysis on distribution shift, 2021.

[46] Kaggle. Stanford cars dataset, 2018. URL `https://www.kaggle.com/datasets/jessicali9530/stanford-cars-dataset`.

[47] github. Stanford cars devkit, 2024. URL `https://github.com/pytorch/vision/files/11644847/car_devkit.tgz`.

[48] Kaggle. Pytorch-stanfordcars-classification, 2023. URL `https://www.kaggle.com/code/subhangaupadhaya/pytorch-stanfordcars-classification/input?select=cars_test_annos_withlabels+%281%29.mat`.

[49] Sergey Ioffe. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[50] AF Agarap. Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375*, 2018.

APPENDIX

A HUE AND SATURATION GROUP ACTION

In this section we prove the proposed hue group action and saturation group action satisfy the axioms for a group action. By definition of a group action, φ_h is a group action if it satisfies the following properties:

1. For all $g, h \in G$ and all $x \in X$
$$\varphi(g, \varphi(h, x)) = \varphi(gh, x) \tag{22}$$
2. For all $x \in X$
$$\varphi(1, x) = x \tag{23}$$
where $1 \in G$ is the identity element of G.

In the remainder of this section we refer to equation 22 as 'property one' and equation 23 as 'property two'.

A.1 HUE GROUP ACTION

Here we show that the proposed hue group action is indeed a group action. The proposed hue group action on the input space is defined:
$$\varphi_h(h_i, x) = ((x_h + h_i)(\text{mod } 255), x_s, x_l), \tag{24}$$
in equation 7 of the main text.

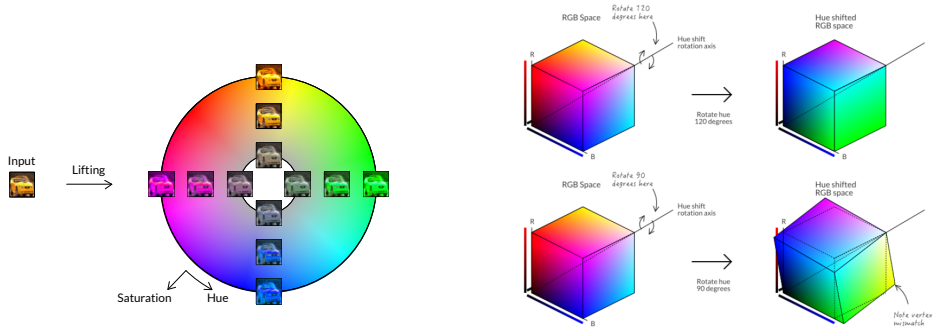
First we show the proposed hue group action φ_h satisfies property one. For any hue shifts h_i and h_j, we have
$$\begin{aligned}
\varphi_h(h_i, \varphi_h(h_j, x)) &= \varphi_h(h_i, ((x_h + h_j)(\text{mod } 255), x_s, x_l)) && (25)\\
&= (((x_h + h_j)(\text{mod } 255) + h_i)(\text{mod } 255), x_s, x_l)) && (26)\\
&= (((x_h + (h_j + h_i))(\text{mod } 255), x_s, x_l)) && (27)\\
&= \varphi_h((h_j + h_i), x), && (28)
\end{aligned}$$
which shows that φ_h satisfies the property one. Now we show φ_h satisfies property two. We have the identity element hue transformation $h_0 = 0$, and
$$\begin{aligned}
\varphi_h(h_0, x) &= ((x_h + h_0)(\text{mod } 255), x_s, x_l) && (29)\\
&= (x_h, x_s, x_l) = x, && (30)
\end{aligned}$$
which shows φ_h satisfies property two. Having shown that φ_h satisfies all properties of a group action, we have that φ_h is, in fact, a group action.

A.2 SATURATION GROUP ACTION

Here we show the proposed saturation group action is a group action. The proposed saturation group action is defined:
$$\varphi_s(s_i, x) = (x_h, \min(x_s + s_i, 255), x_l), \tag{31}$$
in equation 9 of the main text. First we show the proposed saturation group action φ_s satisfies property one. For any saturation shifts s_i and s_j, we have
$$\begin{aligned}
\varphi_s(s_i, \varphi_s(s_j, x)) &= \varphi_s(s_i, (x_h, \min(x_s + s_j, 255), x_l)) && (32)\\
&= (x_h, \min(\min(x_s + s_j, 255) + s_i, 255), x_l) && (33)\\
&= (x_h, \min(x_s + (s_j + s_i), 255), x_l) && (34)\\
&= \varphi_s((s_j + s_i), x), && (35)
\end{aligned}$$
which shows φ_s satisfies property one. Now we show φ_s satisfies property two. We have the identity element saturation transformation $s_0 = 0$, and
$$\begin{aligned}
\varphi_s(s_0, x) &= (x_h, \min(x_s + s_0, 255), x_l) && (36)\\
&= (x_h, x_s, x_l) = x, && (37)
\end{aligned}$$
which shows φ_s satisfies property two. Having shown that φ_s satisfies all properties of a group action, we have that φ_s is, in fact, a group action.

B CECONV EXPERIMENTS

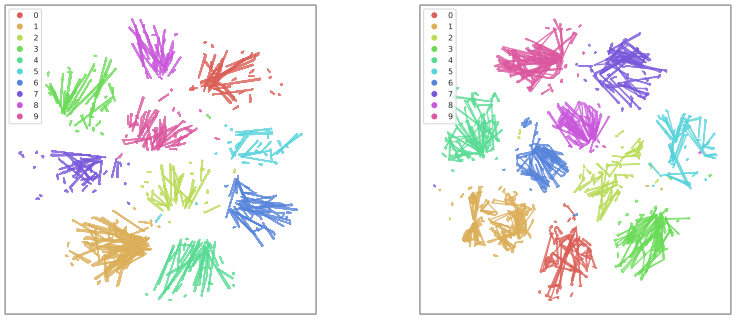
B.1 LIFTING LAYER



(a) Hue-N Lifting (b) CEConv Lifting

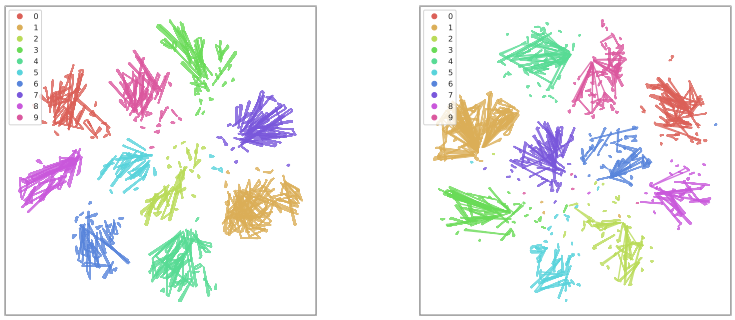
Figure 8: **Lifting layer. (a)** In Hue-N, an input image (left) is lifted to the hue-saturation group (right) by shifting its hue and saturation values. **(b)** CEConv shifts the hue of a filter in the RGB space by rotating its values about an axis passing through the point $p = (1, 1, 1)$. This approach results in invalid hue rotations for all discretizations of the hue group that are not symmetries of the axis-aligned RGB cube (i.e., $N = 1$ and $N = 3$). (Top) $N = 3$. Rotations symmetric to axis-aligned RGB cube and yields valid rotations. (Bottom) $N = 4$. Rotations yield results where invalid RGB values that lie outside of the RGB cube and needs to be projected.

B.2 HUE-SHIFT MNIST FEATURE MAP VISUALIZATION



(a) Hue-3 (b) CEConv-3

Figure 9: **Hue shift MNIST feature map visualization.** tSNE projection of hue shifted feature map trajectories in Hue-3 and CEConv-3.



(a) Hue-4 (b) CEConv-4

Figure 10: **Hue shift MNIST feature map visualization.** tSNE projection of hue shifted feature map trajectories in Hue-4 and CEConv-4. In contrast Hue-4, there is more ambiguous scatter in the feature maps of CEConv-4.

B.3 COLOR SORTING ON CIFAR-10

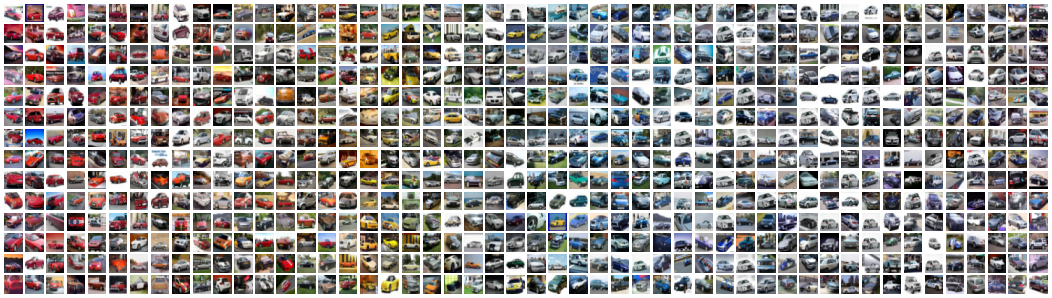


Figure 11: **Color sorting on CIFAR-10.** We show images from CIFAR-10 automobile class ordered by pairwise distance using Hue-3.

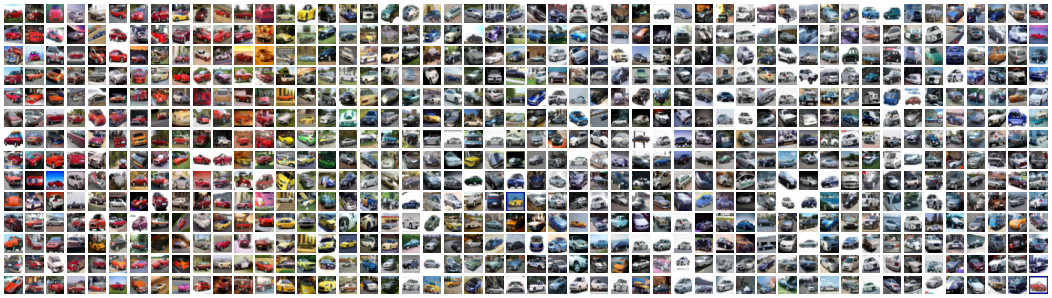


Figure 12: **Color sorting on CIFAR-10.** We show images from CIFAR-10 automobile class ordered by pairwise distance using CEConv-3.

Figure 13: **Color sorting on CIFAR-10.** We show images from CIFAR-10 automobile class ordered by pairwise distance using Hue-4.

Figure 14: **Color sorting on CIFAR-10.** We show images from CIFAR-10 automobile class ordered by pairwise distance using CEConv-4.

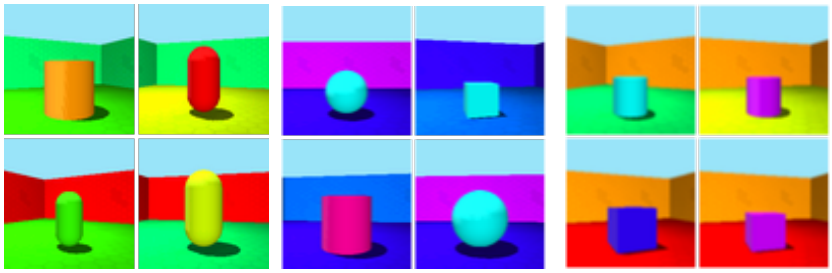
C DATASETS

C.1 HUE-SHIFT MNIST

We introduce the Hue-shift MNIST dataset to evaluate the performance of our equivariant architectures in the presence of global hue shift. Examples in the training set are randomly assigned a hue between $0°$ and $240°$ (see Figure 15a) and examples in the test set are partitioned into an in-distribution set and an out-of-distribution set. In-distribution examples are randomly assigned a hue from the same distribution as the training set, and out-of-distribution examples are randomly assigned a hue outside of the training set distribution (i.e., between $240°$ and $360°$) (see Figure 15b).

(a) Dataset A examples (b) Dataset B examples

Figure 15: **Hue-shift MNIST dataset.** (a) Examples from the training dataset and in-distribution testing dataset A are colored with a randomly selected hue between $0°$ and $240°$. (b) Examples from out-of-distribution testing dataset B are colored with a randomly selected hue between $240°$ and $360°$.



(a) Dataset A examples (b) Dataset B examples (c) Dataset C examples

Figure 16: **Hue-shift 3D Shapes dataset.** (a) Examples from the training dataset and in-distribution testing dataset A. The color of the wall, the floor and the shape are randomly selected from the first half of the color space (colors 0-4). (b) Examples from the out-of-distribution testing dataset B. The color of the wall, the floor and the shape are randomly selected from the second half of the color space (colors 5-9). (c) Examples from the out-of-distribution testing dataset C. The color of the wall and the floor are randomly selected from the first half of the color space, and the color of the shape is randomly selected from the second half of the color space.

C.2 HUE-SHIFT 3D SHAPES

We introduce the Hue-shift 3D Shapes dataset to evaluate the performance of our equivariant architectures in the presence of local hue shift. Examples in the training set are randomly assigned a hue from the first half of the color space (colors 0-4 as defined in [36]) and examples in the test set are are partitioned into an in-distribution set and two out-of-distribution sets. The out-of-distribution test sets are designed to measure robustness to global hue-shift, and local hue-shift. In the out-of-distribution test set designed to measure robustness to global hue-shift, the color of the wall, the floor and the shape are randomly selected from the second half of the color space (colors 5-9 as defined in [36]), and in the out-of-distribution test set designed to measure robustness to local hue-shift, the color of the wall and the floor are randomly selected from the first half of the color space, and the color of the shape is randomly selected from the second half of the color space.

C.3 SMALL NORB

We use the small NORB dataset to evaluate the performance of our equivariant architectures in the presence of luminance shifts. Examples in the training set are assigned with medium lighting conditions (2-3 as defined in [32]). and examples in the test set are are partitioned into an in-distribution set and two out-of-distribution sets. Examples in testset A are in-distribution with lighting label 2-3 as defined in [32]; examples in testset B are out-of-distribution with lower lighting (lighting label 0-1 as defined in [32]); and examples in testset C are out-of-distribution with higher lighting (lighting label 4-5 as defined in [32]). The out-of-distribution testsets are designed to measure robustness to luminance-shift.

(a) Dataset A examples (b) Dataset B examples (c) Dataset C examples

Figure 17: **Small NORB dataset.** (a) Examples from the training dataset and in-distribution testing dataset A. (b) Examples from the out-of-distribution testing dataset B with lower lighting conditions. (c) Examples from the out-of-distribution testing dataset C with higher lighting conditions.

C.4 TINY-IMAGENET

We use the Tiny-ImageNet dataset to evaluate the performance of our equivariant architectures in the presence of color shifts in large datasets. The Tiny-ImageNet dataset [43] is a downsized subset of ImageNet consisting of 100k 64x64 RGB images. Of the 1000 classes represented in ImageNet, 200 are represented in Tiny-ImageNet, each with 500 training examples, 50 validation examples, and 50 unlabeled test examples.

D IMPLEMENTATION AND TRAINING DETAILS

In this section we provide architectural details for our equivariant networks, and training details for each experiment.

All experiments were performed over multiple random seeds to assess the robustness of the model to initialization. Performance statistics on the Hue-shift MNIST, Hue-shift 3D Shapes, and CIFAR-10 datasets were computed over three random seeds (i.e., 1999, 2000, and 2001). Performance statistics the on Camelyon17 dataset were computed over five random seeds (i.e., 1997, 1998, 1999, 2000, and 2001).

All models except Tiny-ImageNet were trained on a shared research computing cluster. Each compute node allocates an Nvidia L40 GPU, 24 core partitions of an Intel Xeon Gold 5320 CPU, and 24GBs of DDR4 3200MHz RDIMMs. Tiny-ImageNet was trained on compute nodes with 4 Nvidia L40 GPUs.

D.1 HUE-SHIFT MNIST

For this task, we compare against the Z2CNN architecture proposed in Cohen and Welling [6]. Our hue-equivariant architecture has the same number of layers as Z2CNN, but with a reduced filter count to maintain a similar number of parameters at each layer. In the final layer of our hue-equivariant architecture, we perform hue group pooling to yield an hue-invariant representation.

We train our equivariant networks and the conventional architectures for 5 epochs with a batch size of 128. We optimize over a cross-entropy loss using the Adam optimizer [44] with $\beta_1 = 0.9$, and $\beta_2 = 0.999$. We use an initial learning rate of 10^{-3} for the $\mathbb{Z}^2$ network, and 10^{-4} for the hue-equivariant network. We train CEConv architectures using the hyperparameters reported in [16].

D.2 HUE-SHIFT 3D SHAPES

For this task we compare against the Z2CNN architectures proposed in [6]. Our hue-equivariant architectures are designed with the same network structure as the baselines, but with a reduced filter count to maintain a similar number of parameters at each layer. In the final layer of our hue-equivariant architectures, we perform hue group pooling to yield an hue-invariant representation.

Following [45], we train the ResNet architectures for 100k iterations with a batch size of 128. We optimize over a cross-entropy loss using SGD with a learning rate of 10^{-2}. Images are down-sampled by a factor of 2 to train the CNN architectures. CNN architectures are trained as described in Section D.1.

D.3 CAMELYON17

For this task, we compare against the ResNet50 architecture [3]. Our hue- and saturation-equivariant architectures have the same number of layers as ResNet50, but with a reduced filter count to maintain a similar number of parameters at each layer. In the final layer of our hue- and saturation-equivariant architectures, we perform a group pooling to yield group invariant representation.

We train our equivariant networks and the conventional architectures for 10k iterations with a batch size of 32. We optimize over a cross-entropy loss using the Adam optimizer [44] with an initial learning rate of 10^{-2}, $\beta_1 = 0.9$, and $\beta_2 = 0.999$. We train CEConv architectures using the hyperparameters reported in [16].

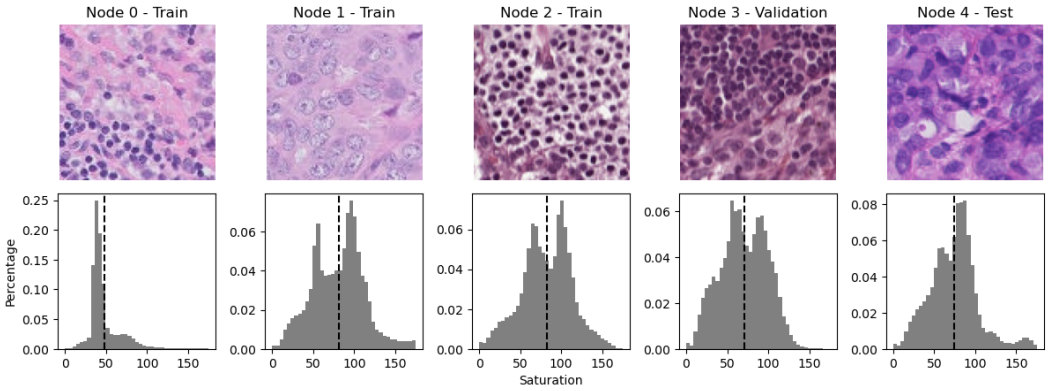


Figure 18: **Camelyon-17 dataset saturation statistics.** Example images and saturation statistics by node (hospital).

We present experimental results for four discretizations of the saturation space. Our choice of discretizations, i.e., $d \in \{1/20, 1/10, 3/20, 1/2\}$, are determined with consideration of the saturation range of the training and validation set (see Figure 18). For a given choice of discretization d, an element $s_i \in S_N$ shifts the saturation of an input image by $s_i = i * d * c$, where c is the maximum pixel value. We limit computational expense of lifting and convolution by performing these operations over the truncated set $\{s_{-1}, s_0, s_1\}$ for our Sat-3 model.

D.4 CIFAR-10

For this task, we compare against the ResNet44 architecture proposed in [6]. Our hue-, saturation- and color-equivariant architectures have the same number of layers as ResNet44, but with a reduced filter count to maintain a similar number of parameters at each layer. In the final layer of our networks, we perform group pooling to yield group invariant representation.

We train our equivariant networks and the conventional architectures for 300 epochs, and a batch size of 128. We optimize over a cross-entropy loss using SGD with an initial learning rate of 10^{-1} and a cosine-annealing scheduler. We evaluate the performance of our model in the presence of natural hue shifts on the CIFAR-100 [42] classification dataset. The dataset consists of 60k examples, 50k of which are used for training and 10k for testing.

D.5 CIFAR-100

For this task, we compare against the ResNet44 architecture proposed in [6]. Our hue-, saturation- and color-equivariant architectures have the same number of layers as ResNet44, but with a reduced filter count to maintain a similar number of parameters at each layer. In the final layer of our networks, we perform group pooling to yield group invariant representation.

We train our equivariant networks and the conventional architectures for 300 epochs, and a batch size of 128. We optimize over a cross-entropy loss using SGD with an initial learning rate of 10^{-1} and a cosine-annealing scheduler. We evaluate the performance of our model in the presence of natural hue shifts on the CIFAR-100 [42] classification dataset. The dataset consists of 60k examples, 40k of which are used for training, 10k for validation, and 10k for testing.

D.6 CALTECH-101

For this task, we compare against the ResNet18 architecture proposed in [3]. Our hue-, saturation- and color-equivariant architectures have the same number of layers as ResNet18, but with a reduced filter count to maintain a similar number of parameters at each layer. In the final layer of our networks, we perform group pooling to yield group invariant representation.

We train our equivariant networks and the conventional architectures for 300 epochs, and a batch size of 16. We optimize over a cross-entropy loss using Adam [44] with an initial learning rate of 10^{-2}. We evaluate the performance of our model in the presence of natural hue shifts on the Caltech-101 [38] classification dataset. The dataset consists of 9,146 examples, $2/3$ of which are used for training and $1/3$ for testing.

D.7 STL-101

For this task, we compare against the ResNet18 architecture proposed in [3]. Our hue-, saturation- and color-equivariant architectures have the same number of layers as ResNet18, but with a reduced filter count to maintain a similar number of parameters at each layer. In the final layer of our networks, we perform group pooling to yield group invariant representation.

We train our equivariant networks and the conventional architectures for 300 epochs, and a batch size of 16. We optimize over a cross-entropy loss using Adam [44] with an initial learning rate of 10^{-2}. We evaluate the performance of our model in the presence of natural hue shifts on the STL-10 [41] classification dataset. The dataset consists of 5k training examples and 8k testing examples.

D.8 STANFORD CARS

For this task, we compare against the ResNet18 architecture proposed in [3]. Our hue-, saturation- and color-equivariant architectures have the same number of layers as ResNet18, but with a reduced filter count to maintain a similar number of parameters at each layer. In the final layer of our networks, we perform group pooling to yield group invariant representation.

We train our equivariant networks and the conventional architectures for 300 epochs, and a batch size of 16. We optimize over a cross-entropy loss using Adam [44] with an initial learning rate of 10^{-2}. We evaluate the performance of our model in the presence of natural hue shifts on the Stanford Cars [40] classification dataset. As the original host for the Stanford cars dataset is no longer maintained, the dataset was constructed from the images [46], the devkit [47], and the annotation files [48]. The dataset consists of 8,144 training examples and 8,041 testing examples.

D.9 OXFORD-IIT PETS

For this task, we compare against the ResNet18 architecture proposed in [3]. Our hue-, saturation- and color-equivariant architectures have the same number of layers as ResNet18, but with a reduced filter count to maintain a similar number of parameters at each layer. In the final layer of our networks, we perform group pooling to yield group invariant representation.

We train our equivariant networks and the conventional architectures for 300 epochs, and a batch size of 16. We optimize over a cross-entropy loss using Adam [44] with an initial learning rate of 10^{-2}. We evaluate the performance of our model in the presence of natural hue shifts on the Oxford-IIT

Pets [39] classification dataset. The dataset consists of 3,680 training examples and 3,669 testing examples.

D.10 SMALL NORB

For this task, we compare against the ResNet18 architecture proposed in [3]. Our luminance-equivariance architectures have the same number of layers as ResNet18, but with a reduced filter count to maintain a similar number of parameters at each layer. In the final layer of our networks, we perform group pooling to yield group invariant representation.

We train our equivariant networks and the conventional architectures for 300 epochs, and a batch size of 16. We optimize over a cross-entropy loss using Adam [44] with an initial learning rate of 10^{-2}. We evaluate the performance of our model in the presence of natural luminance shifts smallnorbs [32] classification dataset.

D.11 TINY-IMAGENET

For this task, we compare against the ResNet18 architecture proposed in [3]. Our hue-equivariant architectures have the same number of layers as ResNet18, but with a reduced filter count to maintain a similar number of parameters at each layer. In the final layer of our networks, we perform group pooling to yield group invariant representation.

We train our equivariant networks and the conventional architectures for 100 epochs, and a batch size of 128. We optimize over a cross-entropy loss using Adam [44] with an initial learning rate of 10^{-2}. We evaluate the performance of our model in the presence of natural hue shifts on the Tiny-ImageNet [43] classification dataset. The dataset consists 200 categories, each with 500 training and 50 validation images. The 50 validation images were split into 25 validation and 25 testing in lieu of the original 50 unlabeled testing images.

E ADDITIONAL RESULTS

E.1 LUMINANCE SHIFT SMALL NORB

We demonstrate improved generalization to luminance-shifts compared to ResNet18 [3] on the small NORB classification dataset [32]. The small NORB dataset consist of 5 categories of objects under different lighting conditions. Variation in images reflects variation in the location and strength of the lighting. Additional details are provided in Appendix C.3.

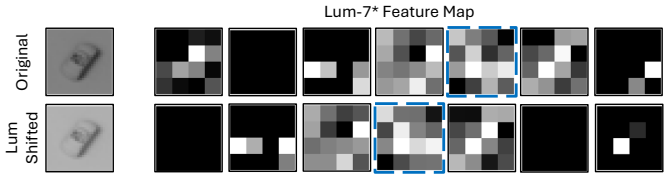


Figure 19: **Luminance equivariant feature maps.** We illustrate the equivariance of our luminance-equivariant model. A luminance shift in the input image space (top-left to bottom-left), results in a feature map translation at each layer of the network (top-right to bottom-right). Corresponding feature maps are highlighted with a blue border.

Table 5: **Generalization to luminance-shift.** Classification error on the small NORB dataset is reported. Our model (Lum-3) achieves improved generalization performance over the conventional CNN model (ResNet18).

Networks	A/A	A/B	A/C	Params
ResNet18	8.32 (0.86)	37.70 (1.51)	33.88 (5.11)	11.2M
Lum-3*	**7.04 (0.65)**	**18.45 (4.83)**	**25.14 (1.07)**	11.1M

Luminance group and group action. We expand the notion of color equivariance proposed in [16] to include equivariance to luminance shifts. In the HSL color space, luminance can be represented by a real number in the interval $[0, 1]$. Using the approximations presented for construction of the discretized saturation group S_N, we similarly identify elements of the discretized luminance group L_N, with those of the integers with addition, $(\mathbb{Z}, +)$.

We define the action of the luminance group on HSL images, $x \in X$ where $x : \mathbb{Z}^2 \to \mathbb{R}^3$, and functions on the discrete luminance group, $y \in Y$, where $y : \mathbb{Z}^2 \times L_N \to \mathbb{R}^K$. An element of the luminance group acts on an HSL image by the group action $\varphi_l : L_N \times X \to X$, which shifts the luminance channel of the image. Concretely, for an HSL image $x \in X$ defined as the concatenation of hue, saturation and luminance channels, i.e., $x = (x_h, x_s, x_l)$, the action of an element l_i of the luminance group L_N is given by

$$\varphi_l(l_i, x) = (x_h, x_s, \min(x_l + l_i, c)), \tag{38}$$

where c is the maximum luminance value. An element of the luminance group acts on a function on the discrete luminance group by the group action $\phi_l : L_N \times Y \to Y$, which "translate" the function on the group. Concretely, for a function f on the discrete luminance group L_N defined as the concatenation of functions $f = (f_1, \ldots, f_N)$, the action of an element l_i in the luminance group L_N is given by

$$\phi_l(l_i, f) = (f_{1+i}, \ldots, f_N, \underbrace{\mathbf{0}, \ldots, \mathbf{0}}_{i}). \tag{39}$$

The action of φ_l on an input image, and ϕ_l on a feature map are shown in Figure 19.

Generalization to luminance-shift. We expand the notion of color equivariance proposed in [16] to capture variation in luminance. The generalization performance of our luminance-equivariant models and Resnet18 [3] are reported in Table 5. Our luminance-equivariant model significantly outperforms the ResNet model in all out-of-distribution cases.

E.2 TINY-IMAGENET

We include classification results on the Tiny-ImageNet dataset [43]. Additional details are provided in Appendix D.

Table 6: **Generalization to color-shift.** Classification error on the Tiny-ImageNet dataset is reported. Our model (Hue-3) achieves generalization performance on-par with conventional CNN model (ResNet18) and baseline (CEConv-3).

Networks	Error	Params
ResNet18	52.95 (1.09)	11.3M
Hue-3*	53.62 (0.35)	11.1M
CEConv-3	54.50 (0.81)	11.1M

E.3 FOREGROUND-BACKGROUND SEGMENTATION

We demonstrate the utility of our method to foreground-background segmentation on the Caltech-101 dataset [38]. The dataset consists of 9,146 examples with annotations containing bounding boxes and contours, which are used to generate ground truth foreground masks. The training set consists of 6941 examples, the validation set consists of 868 examples, and the test set consists of 868 examples. For uniformity of the dataset, we resize all input images and target masks to have a height and width of 224x224.

For this task, we compare the quantitative and qualitative performance of a conventional foreground-background segmentation architecture with our hue-equivariant variant. Each architecture is based on the bottleneck segmentation backbone described in Table 7 where all convolutional layers use residual connections, and each is followed by a batch normalization layer [49] and ReLU [50] activation function. In the final layer, the conventional architecture uses a max pooling layer followed by a sigmoid activation function to produce the predicted foreground-background segmentation mask. Our

hue-equivariant architecture is designed similarly, but uses a group pooling layer instead of a max pooling layer.

We report the quantitative performance of our hue-equivariant model and the conventional model in Table 8, and provide a qualitative comparison of the predicted foreground-background segmentation masks in Figure 20.

Table 7: **Foreground-background segmentation network backbone.**

Layer #	Layer type	Kernel size	Output channels
0	input	-	3
1	conv	7x7	32
2	conv	7x7	64
3	conv	7x7	128
4	conv	7x7	256
5	conv	1x1	256
6	conv	1x1	256
7	conv	1x1	256
8	deconv	7x7	128
9	deconv	7x7	64
10	deconv	7x7	32
11	deconv	7x7	32
12	deconv	7x7	3
13	pooling	-	1

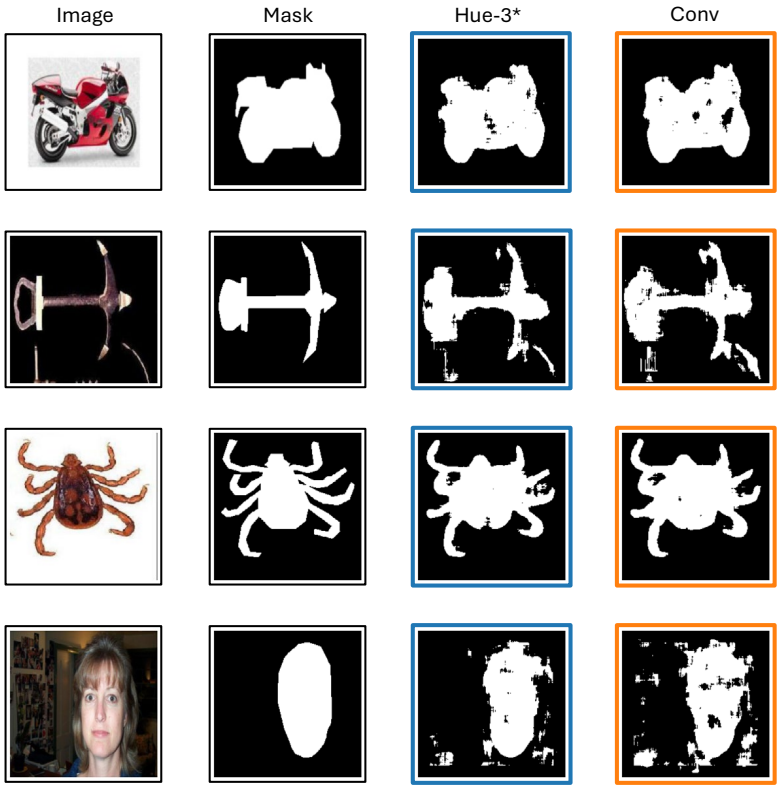


Figure 20: **Foreground-background segmentation.** We show the foreground mask predicted by our approach and a conventional architecture. (Left to right) We show the input image, the ground truth foreground segmentation mask, the foreground segmentation mask predicted by our color-equivariant network, and the foreground segmentation mask predicted by a conventional architecture.

Table 8: **Foreground-background segmentation.** Segmentation error on the Caltech-101 dataset is reported. Our proposed color-equivariant convolutions achieve improved performance over the conventional convolutions.

Convolution	Error	Params
Hue-3*	**9.67 (0.23)**	1.2M
Conv	11.25 (0.12)	1.1M