Orthogonal Transforms For Learning Invariant Representations In Equivariant Neural Networks

Jaspreet Singh
Punjabi University, Patiala
jaspreet_rs21@pbi.ac.in

Chandan Singh
Punjabi University, Patiala
chandan.csp@gmail.com

Ankur Rana
Punjabi University, Patiala
ankurrana628@gmail.com

Abstract

The convolutional layers of the standard convolutional neural networks (CNNs) are equivariant to translation. Recently, a new class of CNNs is introduced which is equivariant to other affine geometric transformations such as rotation and reflection by replacing the standard convolutional layer with the group convolutional layer or using the steerable filters in the convloutional layer. We propose to embed the 2D positional encoding which is invariant to rotation, reflection and translation using orthogonal polar harmonic transforms (PHTs) before flattening the feature maps for fully-connected or classification layer in the equivariant CNN architecture. We select the PHTs among several invariant transforms, as they are very efficient in performance and speed. The proposed 2D positional encoding scheme between the convolutional and fully-connected layers of the equivariant networks is shown to provide significant improvement in performance on the rotated MNIST, CIFAR-10 and CIFAR-100 datasets.

1. Introduction

Convolutional neural networks (CNNs) have achieved state-of-the-art performance for various computer vision tasks, especially the task of image recognition for which CNNs have surpassed the human-level intelligence on the ImageNet dataset. The architectures of standard CNN models consist of feature extraction layers, pooling layers, nonlinear activation functions and fully-connected layers [14]. The convolutional layer is responsible to learn the abstractions from the given input data. However, the convolutional layers of the CNN are only equivariant to translation and the fully-connected layers are neither equivariant nor invariant to any affine geometric transformation. A network is equivariant if the transformation T applied to the input produce predictable transformation T' of the feature space and invariant if the transformation T applied to the input does not affect the output. In real-life scenarios, the images are generally distorted by different geometric transformations such as rotation, translation, reflection, etc., which increases the complexity of image recognition task by manifolds. The one straightforward solution is to encode these transformations via data augmentation simply by transforming the input images while keeping the labels fixed. However, there are inevitable downsides of data augmentation which are 1) invariance to these transformations is not guaranteed, 2) it only captures geometric invariance globally and 3) the network capacity is spend on learning geometric behavior which implicitly affects the descriptive representation learning. Worrall et al. [28] discussed the importance of relative local pose preservation throughout the network layers which is only possible through equivariance and it also conveys more information about an input to the deeper layers. Moreover, the equivariance also guarantees of no information loss when the input get transformed. Thus, it is important that the intermediate layers of CNN models must be equivariant not invariant which has led to the idea of designing the equivariant neural networks. As per our knowledge, the equivariant CNNs developed so far eliminates the spatial dimensions of the filter responses by performing equivariant convolutions and down-sampling to get the final feature vector for classification because the fully-connected layers cannot retain the equivariant representations learned by the intermediate equivariant layers of the equivariant CNNs.
Our contributions are as follows:

- With the assumption that intermediate layers of the network are equivariant to rotation, reflection and translation, we use polar harmonic transforms (PHTs) to encode the global invariance with respect to rotation, reflection and translation.

- The PHTs encode the high-order 2D positional differences of the filter responses or feature maps into the fully-connected layer, as a result, fully-connected layer retain the spatial information in addition to being invariant.

- The proposed invariant encoding scheme adds one more degree of freedom to the design of equivariant neural networks by removing the restriction to eliminate the spatial dimensions of the feature maps to get the final classification vector.

2. Related Work

2.1. Equivariant 2D CNNs

In the seminal work on equivariant CNNs, Cohen and Welling [4] proposed a framework for group equivairant CNNs (G-CNNs). In G-CNNs, the convolutional, pooling, batch normalization and activation operators are redefined in terms of action on a transformation group. G-CNN is defined as the composition of group operations to ensure the equivariance throughout the network. G-CNN showed a significance gain in performance over the standard CNN because it exploits more symmetries in the images. However, G-CNN is limited to discrete transformations such as 90-degree rotations and reflections which leaves the pixel grid intact. Subsequent works on G-CNN are focused on expanding the transformation groups. Hoogeboom et al. [10] proposed the HexaConv network which has a 6-fold rotational symmetry in contrast to the original G-CNN [4]. Chidester et al.[2] introduced the Conic Convolutional and DFT Network (CFNet) which enforce equivariance and invariance in CNN with respect to rotation in the conic regions which originates from the center of an image. Equivariance is enforced by using conic convolutional layer and 2D-DFT is used to enforce invariance. Bekkers et al. [1] introduced the $SE(2)$ equivariant G-CNN for arbitrary angular resolutions by using bilinear interpolation to efficiently transform convolutional kernels. Romero et al. [16] proposed the attention based G-CNN in which the attention is applied during convolution to exploit meaningful symmetries and while suppressing the non-plausible and misleading symmetries.

Based on the idea of exploiting the more symmetries in the data, Cohen and Welling [6] proposed the steerable CNNs. The steerable representation is a composition of elementary feature types where each feature type is associated with a particular symmetry. Worall et al. [28]proposed the Harmonic networks (H-Nets) equivariant to 360-rotations and patch-wise translations by restricting the CNN filters to circular harmonic filters. Weiler et al. [25] proposed the Steerable Filter CNN (SFCNN) jointly equivariant to translation and rotation. SFCNN efficiently computes orientation dependent responses without suffering interpolation artifacts for filter rotation.

Ruthotto and Haber [17] has provided a new understanding on convolutional filters in which a conventional convolutional filter is viewed as a linear combination of partial differential operators (PDOs). Based on this new understanding, Shen et al. [18] introduced the PDO equivariant convolution network (PDO-eCOnvs) which is equivariant to n-dimensional Euclidean group (a more general continuous group) instead of discrete transformation group [4].

2.2. Equivariant 3D CNNs

Equivariance is also important in 3D cases because 3D symmetries are inevitable in 3D objects around vertical axis. Winkels and Cohen [26]proposed the 3D roto-translation G-CNN for pulmonary nodule detection. Worrall and Brostow [27] introudced the CubeNet, a G-CNN with linear equivariance to translation and right angle rotations in 3D. Weiler et al. [24] presented a $SE(3)$-equivariant CNN which is equivariant to rigid body motions. Shen et al. [19] extended their previous work [18] and employed PDO to design 3D PDO-eConv networks. Thomas et al. [22] introduced the Tensor field network which is equivariant to 3D rotations, translation and permutations for 3D point clouds. Further, the equivariant CNN such as spehrical [5] and gauge equivariant CNN [3] are introduced for data defined in other spaces.

3. Group Equivariant Neural Networks

The convolutional layer of a standard CNN is equivariant to translation. Let f be a feature map $f : \mathbb{Z}^2 \rightarrow \mathbb{R}^K$ and O_t a translation operator which translate f by $t \in \mathbb{Z}^2$. The translation equivariance is expressed as follows [4, 10]:

$$\left[\left[O_t f\right] * \psi\right](x) = \left[O_t\left[f * \psi\right]\right](x), \tag{1}$$

where ψ represent a filter. Instead of translation, if we consider a rotation r, (1) is rewritten as follows:

$$\left[\left[O_r f\right] * \psi\right](x) = \left[O_r\left[f * O_{r^{-1}}\psi\right]\right](x). \tag{2}$$

Here, the convolution of a rotated feature map f by a filter ψ equals to the rotation of convolution between f and inversely rotated filter $O_{r^{-1}}\psi$. It is clear from (2) that convolution is not equivariant translation.

Let g be a particular transformation (e.g. rotation or reflection) from a larger group G. Then G-convolutional operation for the first layer operates on functions on $\mathbb{Z}^2$ as follows:

$$\left[f * \psi\right](g) = \sum_{z\in\mathbb{Z}^2}\sum_{k} f_k(z)\psi_k(g^{-1}z). \tag{3}$$

where k denotes the input channels, f_k and ψ_k are function on $\mathbb{Z}^2$. The G-convolutional operation for all other layers is defined as follows:

$$\left[f * \psi\right](g) = \sum_{h\in G}\sum_{k} f_k(h)\psi_k(g^{-1}h). \tag{4}$$

Here, f_k and ψ_k are functions on G instead of $\mathbb{Z}^2$. It can be easily shown that the G-convolution is equivariant to transformations defined by group $g \in G$ as follows:

$$\left[\left[O_g f\right] *_g \psi\right](g) = \left[O_g\left[f *_g \psi\right]\right](g). \tag{5}$$

4. Mathematical Framework for Polar Harmonic Transforms

Polar Harmonic transforms (PHTs) are $2D$ orthogonal transforms defined over the unit disk in the polar coordinate system. PHTs consist of polar complex exponential transforms (PCETs), polar cosine transforms (PCTs) and polar sine transforms (PSTs). Let $f(r, \theta)$ be a 2D feature map which is defined in the continuous polar domain. The PHTs of order n and repetition m for $f(r, \theta)$ is defined as [29, 15]:

$$A_{n,m}(f) = \lambda \int_0^{2\pi} \int_0^1 \left[H_{n,m}(r, \theta)\right]^* f(r, \theta) r dr d\theta, \quad (6)$$

where $\left[H_{n,m}(r, \theta)\right]^*$ is the complex conjugate of $H_{n,m}(r, \theta)$ which can be rewritten in the separable form of the kernel or radial basis function and angular function as follows:

$$H_{n,m}(r, \theta) = R_n(r) e^{im\theta}, \quad (7)$$

where $i = \sqrt{-1}$. The mathematical framework of PHTs defined in (6) is similar for PCETs, PCTs and PSTs while they differ in the form of their kernel or radial basis function $R_n(r)$ and the normalizing parameter λ which are expressed as [29]:

$$\begin{aligned} PCET : R_n(r) = e^{i2\pi nr^2}, \lambda = \frac{1}{\pi}, \\ |n| = |m| = 0, 1, \ldots, \infty. \end{aligned} \quad (8)$$

$$\begin{aligned} PCT : R_n(r) = cos(\pi nr^2), \\ n, |m| = 0, 1, \ldots, \infty. \end{aligned} \quad (9)$$

$$\begin{aligned} PST : R_n(r) = sin(\pi nr^2), \\ n = 1, 2, \ldots, \infty, \\ |m| = 0, 1, \ldots, \infty. \end{aligned} \quad (10)$$

where

$$\lambda = \begin{cases} \frac{1}{\pi} & n = 0 \\ \frac{2}{\pi} & n \neq 0. \end{cases}$$

for PCTs and PSTs.
The kernel function and the angular function of the PHTs satisfies the orthogonality condition

$$\int_0^{2\pi} \int_0^1 \left[H_{n,m}(r, \theta)\right]^* H_{n',m'}(r, \theta) r dr d\theta = \pi \delta_{n,n'} \delta_{m,m'}, \quad (11)$$

where $\delta_{nn'} = 1$ if $n = n'$, and 0 otherwise. Also the radial basis function $R_n(r)$ satisfies the orthogonal condition separately

$$\int_0^1 R_n(r) [R_{n'}(r)]^* r dr = \frac{1}{2} \delta_{n,n'}. \quad (12)$$

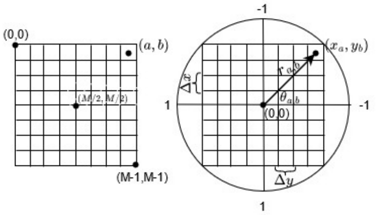

Figure 1. Mapping from rectangular cartesian domain in the left to unit disk in polar domain in the right.

Deriving PHTs using (6) is difficult because the filter responses (feature maps) generated by the CNNs are discrete and defined in the cartesian coordinate system while the PHTs are defined in the continuous polar domain. Therefore, a mapping is performed from cartesian domain to polar domain. Let $f(a, b)$ be a feature map of size, say $M \times M$, and (a, b) is a coordinate in $f(a, b)$. A mapping is performed from the $M \times M$ square domain to $[-1, 1] \times [-1, 1]$, shown in Fig. 1 using the following transformation [20]:

$$\begin{aligned} x_a = \frac{2a + 1 - M}{M\sqrt{2}}, y_b = \frac{2b + 1 - M}{M\sqrt{2}}, \\ a, b = 0, 1, 2, \ldots M - 1, \end{aligned} \quad (13)$$

with $\Delta x = \Delta y = \frac{2}{M\sqrt{2}}$.

Let (a, b) be a coordinate in the rectangular cartesian coordinate system then the corresponding location in polar domain (r_{ab}, θ_{ab}) is derived as $r_{ab} = \sqrt{x_a^2 + y_b^2}$ and $\theta_{ab} = tan^{-1}(y_b, x_a)$, where $\theta_{ab} \in [0, 2\pi)$. Since there is no analytical solution exist to the double integration given in (6), generally, the zeroth order approximation is used:

$$A_{n,m}(f) = \frac{4\lambda}{2M^2} \sum_{a=0}^{M-1} \sum_{b=0}^{M-1} f(a, b) [H_{n,m}(x_a, y_b)]^* \Delta x \Delta y. \quad (14)$$

5. Invariance Properties of PHT

In this section, we discuss the rotation, reflection and translation invariance properties of PHTs.

5.1. Rotation Invariance

Let $f(r, \theta)$ be a function rotated by an arbitrary angle α (anti-cloclkwise) becomes $f^\alpha(r, \theta) = f(r, \theta + \alpha)$ then the PHTs of the rotated $A_{n,m}(f^\alpha)$ and unrotated function $A_{n,m}(f)$ has the following relationship [8, 7] (see Appendix A) :

$$A_{n,m}(f^\alpha) = A_{n,m}(f) e^{-im\alpha}. \quad (15)$$

It is clear from the above relationship that rotation by an angle α shift in the phase by $-m\alpha$. The magnitude cancels out the exponential factor and become invariant to rotation as follows:

$$|A_{n,m}(f^{\alpha})| = |A_{n,m}(f)|. \tag{16}$$

5.2. Reflection Invariance

Let $f^h(a,b)$ and $f^v(a,b)$ are the horizontal and vertical flipped versions of the original function $f(a,b)$ then the relationship between the PHTs of the original and flipped functions are defined as follows (see Appendix B) [13]:

$$A_{n,m}(f^h) = (-1)^m [A_{n,m}(f)]^*. \tag{17}$$

and

$$A_{n,m}(f^v) = [A_{n,m}(f)]^*. \tag{18}$$

where $[.]^*$ is the complex conjugate. The magnitude of (17) and (18) is invariant to horizontal and vertical flipping.

5.3. Translation Invariance

For PHTs, the invariance to translation can be simply achieved by shifting the center of the coordinate system (a,b) in such a way that it coincides with the centroid of the feature maps. Let $f'(a+\Delta a, b+\Delta b)$ is the translated version of $f(a,b)$ by the translation factor Δa and Δb. The central PHTs invariant to translation are computed by replacing the center of coordinate system $(M/2, M/2)$ of (a,b) with its centroid as follows [21]:

$$\overline{AM}_{n,m}(f) = \lambda \sum_{a=0}^{M-1} \sum_{b=0}^{M-1} f(\overline{x}_a, \overline{y}_b) R_n(\overline{r}_{a,b}) e^{-im\overline{\theta}_{ab}}, \tag{19}$$

where $\overline{x_a}$ and $\overline{y_b}$ are obtained as follows:

$$\overline{x}_a = \frac{2a+1-\overline{x}}{D}, \, \overline{y}_b = \frac{2b+1-\overline{y}}{D}. \tag{20}$$

The centroid $(\overline{x}, \overline{y})$ are obtained as follows[13]:

$$\overline{x} = \frac{\sum_{a=0}^{M-1}\sum_{b=0}^{M-1} a.f(a,b)}{\sum_{a=0}^{M-1}\sum_{b=0}^{M-1} f(a,b)}, \overline{y} = \frac{\sum_{a=0}^{M-1}\sum_{b=0}^{M-1} b.f(a,b)}{\sum_{a=0}^{M-1}\sum_{b=0}^{M-1} f(a,b)}. \tag{21}$$

6. Invariant 2D Positional Encoding Using PHTs

In a general standard CNN architecture, some number of fully-connected layers are applied after the final convolution layer. As discussed earlier, the equivariant CNNs are the composition of equivariant operations (e.g., convolution, pooling, batch normalization, and activation) to ensure the equivariance throughout the network. Since the fully-connected layers are not equivariant/invariant to the transformations, therefore, can not preserve the equivariant/invariant representations learned by the equivariant CNNs. The convolution and down-sampling is applied to eliminate the spatial dimensions of the feature maps until feature maps become merely a vector in order to retain the learned equivariance/invariance representations.

Table 1. Test error(%) obtained using proposed G-CNN+PCETs and G-CNN+PCTs for different transform order (n_{max}).

Methods	**3**	**5**	**7**	**9**	**11**	**13**
G-CNN($p4$) +PCETs	1.78	1.60	1.62	1.66	1.71	1.74
G-CNN($p4$) +PCTs	1.80	1.61	1.64	1.68	1.76	1.81

Let F be a equvariant CNN which is defined as a composition of L equivarint layers and l is a particular layer in F. The feature maps $\boldsymbol{Y}^l$ generated by F at a particular layer l are denoted as:

$$\boldsymbol{Y}^l = F^l(\boldsymbol{X}), \tag{22}$$

where $\boldsymbol{Y}^l$ is of dimensions $h^l \times w^l \times \theta \times c^l$ and h, w, θ, c represent the height, width, transformations and channels, respectively. The feature maps $\boldsymbol{Y}^l$ are functions on group G, a mapping function Ω_θ^l is applied which maps G into Z^2 which eliminates the transformation axis θ by linearly concatenating the transformation groups to the channels c^l one after another and sort the feature maps based on their activation. The operation is defined as:

$$\boldsymbol{W}^l = \Omega_\theta^l(\boldsymbol{Y}^l), \tag{23}$$

where $\boldsymbol{W}^l$ is of dimensions $h^l \times w^l \times c_\theta^l$ and $c_\theta^l = \theta \times c^l$. It is important to note that mapping function Ω_θ^l is equivariant and can retain the equivariance representations learned by the intermediate equivariant layers of CNN. Finally, the central PHTs are computed over c^l as follows:

$$I_k = |\overline{AM}_{n,m}(\boldsymbol{W}_k^l)|, k = 1, 2, \ldots c_\theta^l, \tag{24}$$

where $\boldsymbol{I}_k$ is a vector of size $(n_{max}+1)^2$ and (n_{max}) is the maximum PHT order. The final invariant representation $\boldsymbol{I}$ is obtained by linearly concatenating the vectors $\boldsymbol{I}_k$, obtained as $\boldsymbol{I} = [\boldsymbol{I}_1, \boldsymbol{I}_2, \ldots \boldsymbol{I}_{c_\theta^l}]$ and passed to the following fully-connected layer for classification. The proposed invariant 2D positional encoding scheme using the equivariant and invariant operators for equivariant CNN is shown in Fig. 2. In CFNet[2], DFTs is used to encode invariance with respect to rotation only. We selected PHTs among the various orthogonal transforms due to their invariance to the large group transformations (i.e., rotation, reflection and translation), high performance and low-computation complexity. Moreover, we can compute the infinite number of invariants with the help of PHTs in contrast to DFTs which are limited and finite to the size of f. The reason is that the PHTs are continuous transforms in contrast to DFT which is discrete. In the proposed framework for equivariant CNNs, the

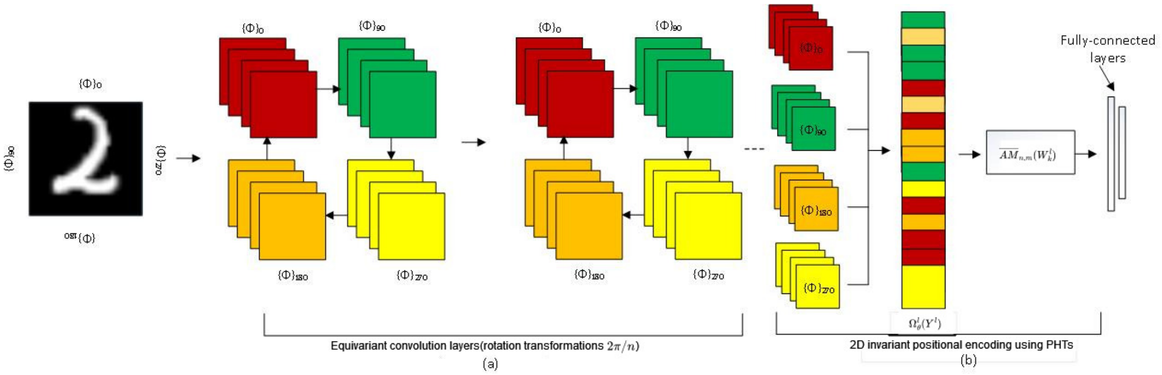
$\{\Phi\}_0$
$\{\Phi\}_{90}$
$\{\Phi\}_{180}$
$\{\Phi\}_{270}$
Fully-connected layers
$\overline{AM}_{n,m}(W_k^l)$
$\Omega_\theta^l(Y^l)$
Equivariant convolution layers(rotation transformations $2\pi/n$)
(a)
2D invariant positional encoding using PHTs
(b)

Figure 2. Network architecture with equivariant intermediate layers and invariant fully-connected layers:(a) Intermediate equivariant layers and (b) PHTs-based 2D positional encoding followed by fully-connected layers.

Table 2. Test error(%) obtained by existing and proposed methods on rotated MNIST dataset.

Methods	**Test error (%)**	**params**
Z2CNN [4]	5.03	22k
Z2CNN+data aug [4]	3.50	22k
G-CNN($p4$) [4]	3.21	25k
CFNet [2]	2.00	-
H-Net [28]	1.69	33k
PDO-eConv($p8$) [18]	1.87	26k
G-CNN+PCETs($p4$)	1.60± 0.002	26.5k
G-CNN+PCTs($p4$)	1.61± 0.003	26.5k
PDO-eConv+PCETs($p8$)	1.56± 0.004	27.5k
PDO-eConv+PCTs($p8$)	1.58± 0.004	27.5k

intermediate layers of the equivariant CNNs are equivariant to the transformations which preserve the local relative poses without loosing any vital information and the final fully-connected layers become invariant to the transformation globally. Moreover, the intuition behind using all the transformations is that the learned feature maps are unique and represent the independent information or features of the input data. This aspect is very useful and further validated using experiments in Section 7. Another important use of the proposed integration is for the high-resolution representation learning [23] which is important for the sensitive vision problems such as human pose estimation, semantic segmentation, and object detection including image classification because it resolve the issue of down-sampling of feature maps to a vector for final classification.

7. Experimental Results

In this section, we evaluate the proposed invariant 2D positional encoding scheme on top of G-CNN and PDO-eConv on rotated MNIST, CIFAR-10, and CIFAR-100 datasets. The rotated MNIST dataset is chosen to evaluate the performance of the proposed equivariant and invariant architecture under rotation. CIFAR-10 and CIFAR-100 are the more natural large-scale color benchmarking image datasets commonly used to evaluate the deep neural networks architecture. The experiments are conducted on a NVIDIA Quadro P4000 8GB GPU and the proposed 2D invariant positional encoding scheme using PCETs is implemented using TensorFlow 1.14.

7.1. Rotated MNIST

The rotated MNIST dataset [12] is most frequently used to investigate the equivariance properties of the equivariant CNNs. It is split into train, validation and test sets of size 10000, 2000 and 50000 images, respectively. The test split is rotated to random rotations in $[0, 2\pi)$. For experimental purpose, G-CNN($p4$)[1] architecture is used [4] which contains 6 layers of 3×3 convolutional kernels. The proposed invariant 2D positional encoding scheme is integrated after layer 6 ($l = 6$) in the architecture of G-CNN followed by a fully-connected layer. The dimensions of the feature maps ($\boldsymbol{Y}^6$) at layer 6 are $4 \times 4 \times 4 \times 10$, and after applying the mapping operator (Ω_θ^l), $\boldsymbol{W^l}$ is obtained and the dimensions of $\boldsymbol{W^l}$ are $4 \times 4 \times 40$. In the case of PDO-eConv ($p8$), the dimensions of ($\boldsymbol{Y}^6$) at layer 6 are $4 \times 4 \times 8 \times 7$ and $\boldsymbol{W^l}$ are $4 \times 4 \times 56$. The proposed model architecture is trained using the Adam optimizer with a weight decay of 0.01 and the weights of the fully-connected layer are initialized using Xavier initialization. The model is trained using a batch of size 128 upto 200 epochs. The initial learning rate is set to 0.001 and divided by 10 after 50% and 75% of the total 200 epochs. The recognition rates obtained by PHTs (PCETs and PCTs) are shown in Table 1 for different

[1] pn denotes a group generated by translations and rotations by $2\pi/n$ and pnm denote a group generated by translations, reflections and rotations by $2\pi/n$.

Table 3. Test error(%) obtained by existing and proposed methods on CIFAR-10 and CIFAR-100.

Methods	**G**	**Depth**	**CIFAR-10**	**CIFAR-100**	**params**
ResNet	Z^2	44	5.61	24.1	2.64M
G-CNN	$p4m$	44	4.98	23.24	2.62M
PDO-eConv	$p8$	44	3.76	20.1	2.62M
G-CNN+PCETs	$p4m$	44	4.76	23.02	2.63M
G-CNN+PCTs	$p4m$	44	4.80	23.08	2.63M
PDO-eConv+PCETs	$p8$	44	3.58	18.23	2.63M
PDO-eConv+PCTs	$p8$	44	3.62	18.51	2.63M

transform orders(n_{max}). PHTs obtains lowest test error at order $n_{max} = 5$ which is selected for further experiments in this section. For order $n_{max} = 5$, the invariant 2D positional encoding scheme adds 36×40 and 36×56 number of learning parameters to G-CNN and PDO-eConv, respectively. Table 2 shows the recognition accuracy obtained by existing Z2CNN, Z2CNN with data augmentation (i.e., Z2CNN+data aug.) G-CNN, CFNet, H-Net, PDO-eConv and the proposed G-CNN+PCETs, G-CNN+PCTs, PDO-eConv+PCETs, PDO-eConv+PCTs. As it can be easily observed from the table that the proposed invariant scheme on top of G-CNN and PDO-eConv networks reduces the test error significantly as compared to the existing methods.

7.2. Natural Image Classification

Here, we evaluate the performance of the proposed invariant 2D positional encoding scheme using two more natural image datasets which are CIFAR-10 and CIFAR-100 [11]. The CIFAR-10 and CIFAR-100 datasets consists of colored natural images of size 32×32. The CIFAR-10 dataset is categorized into 10 classes while CIFAR-100 dataset is categorized into 100 classes. Both CIFAR datasets are divided into training and test sets of size 50000 and 10000 images, respectively. The experiments are performed according to the specification specified in [18]. The 5000 images are selected as a validation set from the training set and the model with lowest validation error is selected during training. The training set is augmented using the standard augmentation scheme which is by mirroring/shifting [18] and the images are normalized by the means and standard deviation of their corresponding channels. ResNet [9] is choosen as the basis model to evaluate the proposed invariant 2D positional encoding on top of G-CNN and PDO-eConv. The ResNet model consist of an initial convolutional layer, followed by three stages of $2n$ convolutional layers using k_i filters at stage i, and a final classification layer which makes total $6n+2$ layers. The results are shown in Table 3 for ResNet-44, where $k_i = 11, 23, 45$ and $n = 7$. The convolutional layers of ResNet-44 are replaced by G-convolutional layers and PDO-eConv layers for G-CNN and PDO-eConv networks, respectively. The models are trained using stochastic gradient descent (SGD) with momentum 0.9 with a batch of size 128 for 300 epochs. The initial learning rate is set to 0.1, weight decay is 0.001 and the learning rate is divided by 10 at 50% and 75% of the total training epochs which are 300. The weights of the fully-connected layer are initialized using Xavier initialization method [18]. The dimensions of the filter responses after the final convolutional stage is $8 \times 8 \times 8 \times 45$ for G-CNN and $8 \times 8 \times 8 \times 45$ for PDO-eConv. After mapping using the $\Omega_{\theta}^{l}(\boldsymbol{Y}^{l})$ the dimensions get mapped to $8 \times 8 \times 360$ and $8 \times 8 \times 360$ for G-CNN and PDO-eConv, respectively. The proposed invariant 2D positional encoding scheme is applied which generates the feature vectors of size 36×360 for G-CNN and 36×360 for PDO-eConv. The number of additional parameters added to the network is 12,960. It can again observed from the Table 3 that the proposed invariant scheme on top of G-CNN and PDO-eConv networks reduces the test error significantly for both the datasets.

8. Conclusion

In this paper, we have proposed a 2D positional encoding scheme using orthogonal PHTs to learn the invariant representations in the equivariant CNNs by integrating the orthogonal PHTs in the transition between equivariant convolutional layers and fully-connected layers. The proposed encoding scheme is invariant to rotation, reflection and translation. Moreover, the kernel computation of PHTs is extremely simple and has no numerical instability issues. The experiments are conducted using PCETs and PCTs on rotated MNIST to evaluate the equivariance and invariance properties of the proposed architecture and CIFAR-10 and CIFAR-100 datasets. The proposed invariant encoding scheme provides improved recognition accuracy as compared to CNN(Z2-CNN), Z2CNN+data aug., G-CNN($p4, p4m$), CFNet, PDO-eConv($p8$) and H-Net and put off the need to remove the spatial dimensions by downsampling filter responses for equivariant CNNs.

References

[1] Erik J Bekkers, Maxime W Lafarge, Mitko Veta, Koen AJ Eppenhof, Josien PW Pluim, and Remco Duits. Roto-translation covariant convolutional networks for medical im-

age analysis. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 440–448. Springer, 2018.

[2] Benjamin Chidester, Tianming Zhou, Minh N Do, and Jian Ma. Rotation equivariant and invariant neural networks for microscopy image analysis. *Bioinformatics*, 35(14):i530–i537, 2019.

[3] 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.

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

[5] Taco S Cohen, Mario Geiger, Jonas Köhler, and Max Welling. Spherical cnns. *arXiv preprint arXiv:1801.10130*, 2018.

[6] Taco S Cohen and Max Welling. Steerable cnns. *arXiv preprint arXiv:1612.08498*, 2016.

[7] Jan Flusser, Tomas Suk, and Barbara Zitová. *2D and 3D image analysis by moments*. John Wiley & Sons, 2016.

[8] Jan Flusser, Barbara Zitova, and Tomas Suk. *Moments and moment invariants in pattern recognition*. John Wiley & Sons, 2009.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European conference on computer vision*, pages 630–645. Springer, 2016.

[10] Emiel Hoogeboom, Jorn WT Peters, Taco S Cohen, and Max Welling. Hexaconv. In *International Conference on Learning Representations*, 2018.

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

[12] Hugo Larochelle, Dumitru Erhan, Aaron Courville, James Bergstra, and Yoshua Bengio. An empirical evaluation of deep architectures on problems with many factors of variation. In *Proceedings of the 24th international conference on Machine learning*, pages 473–480, 2007.

[13] Miros law Pawlak. Image analysis by moments: reconstruction and computational aspects. *Oficyna Wydawnicza Politechniki Wrocławskiej*, 2006.

[14] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

[15] Yue Nan Li. Quaternion polar harmonic transforms for color images. *IEEE Signal Processing Letters*, 20(8):803–806, 2013.

[16] David Romero, Erik Bekkers, Jakub Tomczak, and Mark Hoogendoorn. Attentive group equivariant convolutional networks. In *International Conference on Machine Learning*, pages 8188–8199. PMLR, 2020.

[17] Lars Ruthotto and Eldad Haber. Deep neural networks motivated by partial differential equations. *Journal of Mathematical Imaging and Vision*, 62(3):352–364, 2020.

[18] Zhengyang Shen, Lingshen He, Zhouchen Lin, and Jinwen Ma. Pdo-econvs: Partial differential operator based equivariant convolutions. In *International Conference on Machine Learning*, pages 8697–8706. PMLR, 2020.

[19] Zhengyang Shen, Tao Hong, Qi She, Jinwen Ma, and Zhouchen Lin. Pdo-s3dcnns: Partial differential operator based steerable 3d cnns. In *International Conference on Machine Learning*, pages 19827–19846. PMLR, 2022.

[20] Chandan Singh and Amandeep Kaur. Fast computation of polar harmonic transforms. *Journal of Real-Time Image Processing*, 10(1):59–66, 2015.

[21] Chandan Singh and Jaspreet Singh. A survey on rotation invariance of orthogonal moments and transforms. *Signal Processing*, page 108086, 2021.

[22] 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.

[23] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, Wenyu Liu, and Bin Xiao. Deep high-resolution representation learning for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(10):3349–3364, 2021.

[24] Maurice Weiler and Gabriele Cesa. General $e(2)$-equivariant steerable cnns. *arXiv preprint arXiv:1911.08251*, 2019.

[25] Maurice Weiler, Fred A Hamprecht, and Martin Storath. Learning steerable filters for rotation equivariant cnns. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 849–858, 2018.

[26] Marysia Winkels and Taco S Cohen. 3d g-cnns for pulmonary nodule detection. *arXiv preprint arXiv:1804.04656*, 2018.

[27] Daniel Worrall and Gabriel Brostow. Cubenet: Equivariance to 3d rotation and translation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 567–584, 2018.

[28] 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.

[29] Pew-Thian Yap, Xudong Jiang, and Alex Chichung Kot. Two-dimensional polar harmonic transforms for invariant image representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(7):1259–1270, 2009.

Appendix A. Rotation Invariance

Let $f^{\alpha}(r,\theta)$ is the rotated version of $f(r,\theta)$ rotated by an angle α then the relationship between the PHTs of original and rotated function is defined as follows[8]:

$$\begin{aligned}AM_{n,m}(f^{\alpha}) &= \lambda \int_0^{2\pi}\int_0^1 f^{\alpha}(r,\theta)R_n(r)e^{-im\theta}rdrd\theta,\\ &= \lambda \int_0^{2\pi}\int_0^1 f(r,\theta+\alpha)R_n(r)e^{-im\theta}rdrd\theta,\\ &= \lambda \int_0^{2\pi}\int_0^1 f(r,\theta^{'})R_n(r)e^{-im(\theta^{'}-\alpha)}rdrd\theta^{'},\\ &= \lambda \int_0^{2\pi}\int_0^1 f(r,\theta^{'})R_n(r)e^{-im\theta^{'}}e^{im\alpha}rdrd\theta^{'},\\ &= e^{im\alpha}AM_{n,m}(f).\end{aligned} \tag{A1}$$

This relationship shows that PHTs of the original and the rotated function undergo phase-shift by an angle $m\alpha$ and the magnitude cancels out the effect of rotation angle α.

Appendix B. Reflection Invariance

Let $f^v(a,b) = f(a,-b)$ is the vertical flipped version of $f(a,b)$ then the PHTs of the original and vertical flipped version has the following relationship[13]:

$$\begin{aligned}AM_{n,m}(f^v) &= \lambda \sum_{a=0}^{M-1}\sum_{b=0}^{M-1} f(x_a,-y_b)R_n(r_{a,b})e^{-im\theta_{ab}},\\ &= \lambda \sum_{a=0}^{M-1}\sum_{b=0}^{M-1} f(x_a,y_b)R_n(r_{a,b})e^{-im(-\theta_{ab})},\\ &= \lambda \sum_{a=0}^{M-1}\sum_{b=0}^{M-1} f(x_a,y_b)R_n(r_{a,b})e^{im\theta_{ab}},\\ &= AM_n^*(f).\end{aligned} \tag{A2}$$

Similarly, the relationship for horizontal flipped version $f^h(a,b) = f(-a,b)$ is defined as follows:

$$\begin{aligned}AM_n(f^h) &= \lambda \sum_{s=0}^{M-1}\sum_{t=0}^{M-1} f(-x_a,y_b)R_n(r_{a,b})e^{-im\theta_{ab}},\\ &= \lambda \sum_{a=0}^{M-1}\sum_{b=0}^{M-1} f(x_a,y_b)R_n(r_{a,b})e^{-im(\pi-\theta_{ab})},\\ &= \lambda \sum_{a=0}^{M-1}\sum_{b=0}^{M-1} (-1)^m f(x_a,y_b)R_n(r_{a,b})e^{im\theta_{ab}},\\ &= (-1)^m AM_{n,m}^*(f).\end{aligned} \tag{A3}$$