$E(2)$-Equivariant Vision Transformer

Renjun Xu[*1] **Kaifan Yang**[*1,2] **Ke Liu**[*†1,2] **Fengxiang He**[3,2]

[1]College of Computer Science and Technology, Zhejiang University
[2]JD Explore Academy, JD.com, Inc.
[3]AIAI, School of Informatics, University of Edinburgh

Abstract

Vision Transformer (ViT) has achieved remarkable performance in computer vision. However, positional encoding in ViT makes it substantially difficult to learn the intrinsic equivariance in data. Initial attempts have been made on designing equivariant ViT but are proved defective in some cases in this paper. To address this issue, we design a Group Equivariant Vision Transformer (GE-ViT) via a novel, effective positional encoding operator. We prove that GE-ViT meets all the theoretical requirements of an equivariant neural network. Comprehensive experiments are conducted on standard benchmark datasets, demonstrating that GE-ViT significantly outperforms non-equivariant self-attention networks. The code is available at https://github.com/ZJUCDSYangKaifan/GEVit.

1 INTRODUCTION

Equivariance is an intrinsic property of many domains, such as image processing [Krizhevsky et al., 2012], 3D point cloud processing [Li et al., 2018], chemistry [Faber et al., 2016], astronomy [Ntampaka et al., 2016], economics [Qin et al., 2022], etc. Translation equivariance is naturally guaranteed in CNNs, i.e., if a pattern in an image is translated, the learned image representation by a CNN is also translated in the same way. However, realizing equivariance is not natural for other models or groups. Zaheer et al. [2017], Cohen and Welling [2016a], and Cohen et al. [2019] adopt machine learning to realize the equivariance via modifying classic neural networks. In visual tasks, the equivariance has been highlighted in the aspects of permutation [Romero and Cordonnier, 2020], symmetry [Krizhevsky et al., 2012], and translation [Worrall et al., 2017].

Vision Transformer (ViT) [Dosovitskiy et al., 2020] based on self-attention has been widely used in computer vision. According to the theoretical analyze (§4.1), it is the positional encoding that destroys the equivariance of self-attention. To extend the equivariance of ViT to arbitrary affine groups, a new positional encoding should be designed to replace the traditional one. Initial attempts have been made to modify the self-attention to be equivariant [Romero and Cordonnier, 2020, Fuchs et al., 2020, Hutchinson et al., 2021]. The SE(3)-Transformers [Fuchs et al., 2020] takes the irreducible representations of SO(3) and LieTransformer [Hutchinson et al., 2021] utilizes the Lie algebra. However, they focus on processing 3-D point cloud data. GSA-Nets [Romero and Cordonnier, 2020] proposed new positional encoding operations, which meet challenges in some cases.

To address this issue, we propose a **G**roup **E**quivariant **Vi**sion **T**ransformer (**GE-ViT**) via a novel, effective equivariant positional encoding operation. We prove that the GE-ViT has met the theoretical requirements of a group equivariant neural network.

The equivariance in GE-ViT brought advantages over previous works. The group equivariance significantly improves the generalization for its equivariance on group [Sannai et al., 2021, He and Tao, 2020]. Parameter efficiency and steerability [Cohen and Welling, 2016b, Weiler et al., 2018] are also guaranteed. The weights of group equivariant CNN kernels are tied to particular positions of neighborhoods on the group, which requires a large number of parameters. While GE-ViT leverages long-range dependencies on group functions under a fixed parameter budget, which can express any group convolutional kernel [Romero and Cordonnier, 2020]. GE-ViT is steerable since group operations are performed directly on the positional encoding [Weiler et al., 2018]. This performance of GE-ViT is evaluated by experiments which fully support our algorithm.

The contributions of this work are summarized as follows:

- We propose a novel Group Equivariant Vision Transformer (GE-ViT). Mathematical analysis demonstrates

[*]Contributed equally.
[†]Corresponding author: Ke Liu

Accepted for the 39th Conference on Uncertainty in Artificial Intelligence (UAI 2023).

that the theoretical requirements of an equivariance neural network are met in GE-ViT.

- We conduct experiments on standard benchmark datasets. The empirical results demonstrate consistent improvements of GE-ViT over previous works.

The rest of this paper is organized as follows. Section 2 reviews the related works. Section 3 introduces self-attention in detail and defines the notations in our paper. Preliminary concepts on groups and equivariance are introduced in Section 4. Theory analysis of GE-ViT, especially that regarding positional encoding, is presented in Section 5. We report the experiments in Section 6. The discussion and future work are given in Section 7.

2 RELATED WORK

Transformer [Vaswani et al., 2017] and its variants [Devlin et al., 2018] have achieved remarkable success in natural language processing (NLP) [Vaswani et al., 2017], computer vision (CV) [Carion et al., 2020, Dosovitskiy et al., 2020, Liu et al., 2021], and other fields [Jumper et al., 2021, Liu et al., 2022]. Different from previous methods, e.g., recurrent neural networks (RNNs) [Elman, 1990] and convolutional neural networks (CNNs) [LeCun et al., 1989], transformer handles the input tokens simultaneously, which has shown competitive performance and superior ability in capturing long-range dependencies between these tokens. The core of transformer is the self-attention operation [Vaswani et al., 2017], which excels at modeling the relationship of tokens in a sequence. Self-attention takes the similarity of token representations as attention scores and update the representations with the score weighted sum of them in an iterative manner.

The group equivariant neural network was first proposed by Cohen and Welling [2016a], which extended the equivariance of CNNs from translation to discrete groups. The main idea of the approach is that it uses standard convolutional kernels and transforms them or the feature maps for each of the elements in the group [Cohen et al., 2019]. This approach is easy to implement and has been used widely [Marcos et al., 2017, Zhou et al., 2017]. However, this kind of approach can only be used in particular circumstances where locations are discrete and the group cardinality is small such as image data.

Nowadays, many methods have been proposed for designing group equivariant networks. The equivariance of networks has been extended to general symmetry groups [Bekkers, 2019, Venkataraman et al., 2019, Weiler and Cesa, 2019]. Macroscopically, equivariant neural networks can be broadly categorised by whether the input spatial data is lifted onto the space of functions on group G or not [Hutchinson et al., 2021]. Without lifting, the equivariant map is defined on the homogeneous input space $\mathcal{X}$. For convolutional networks, the kernel is always expressed using a basis of equivariant functions, such as circular harmonics [Weiler et al., 2018, Worrall et al., 2017], spherical harmonics [Thomas et al., 2018]. With lifting, the equivariant map is defined on G [Cohen et al., 2018, Esteves et al., 2018, Finzi et al., 2020, Hutchinson et al., 2021, Romero and Hoogendoorn, 2019]. Both GE-ViT and GSA-Nets use lifting to define equivariant self-attention [Romero and Cordonnier, 2020].

Research on how to make the self-attention satisfy the general group equivariance is already existed [Romero et al., 2020]. The SE(3)-Transformers [Fuchs et al., 2020] achieves this goal via the irreducible representations of SO(3) and LieTransformer [Hutchinson et al., 2021] achieves this by means of Lie algebra. However, GE-ViT, the model proposed by this paper, achieved this by designing a new positional encoding. Besides, the above two models are specifically designed for processing 3-D point cloud data while GE-ViT is good at processing regular image data.

3 VISION TRANSFORMER

In this section, we formally formulate vision transformers [Dosovitskiy et al., 2020].

3.1 ARCHITECTURE

We first define some notations for brevity. Set $\{1, 2, 3, \ldots, n\}$ is denoted by $[n]$. Let $\mathcal{S} = [N]$. $L_{\mathcal{V}}(\mathcal{S})$ denote the space of functions $\{f : \mathcal{S} \to \mathcal{V}\}$, where $\mathcal{V}$ represents a vector space. Accordingly, a matrix $\mathbf{X} \in \mathbb{R}^{N \times C_{\text{in}}}$ can be interpreted as a vector-valued function $f_X : \mathcal{S} \to \mathbb{R}^{C_{\text{in}}}$ that maps element $i \in \mathcal{S}$ to C_{in}-dimension vector $\mathbf{X}_i \in \mathbb{R}^{C_{\text{in}}}$. A matrix multiplication, $\mathbf{X}\mathbf{W}_y^\top$ between matrices $\mathbf{X} \in \mathbb{R}^{N \times C_{\text{in}}}$ and $\mathbf{W}_y \in \mathbb{R}^{C_{\text{out}} \times C_{\text{in}}}$ can be represented as a function $\varphi_y : L_{\mathbb{R}^{C_{\text{in}}}}(\mathcal{S}) \to L_{\mathbb{R}^{C_{\text{out}}}}(\mathcal{S})$, as $\varphi_y(f_X) = f_{XW_y^T}$.

ViT reshapes the image into a sequence of 2D tokens, which are then flattened and mapped into token embeddings, i.e. vectors, with a trainable linear projection [Dosovitskiy et al., 2020]. To get the structural information of an image involved, the positional encodings are calculated and aggregated with the corresponding token embeddings to form the representations of the tokens. Finally, self-attention mechanisms are performed on these token representations.

3.2 SELF-ATTENTION

The overview of self-attention is shown in Fig. 1. A self-attention module takes in N inputs and returns N outputs. Let $\mathbf{X} \in \mathbb{R}^{N \times C_{\text{in}}}$ be an input matrix consisting of N tokens of C_{in} dimensions. Let $\mathbf{Y} \in \mathbb{R}^{N \times C_{\text{out}}}$ be an output matrix consisting of N tokens of C_{out} dimensions obtained from

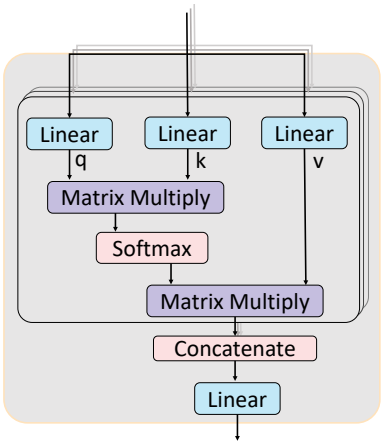


Figure 1: Illustration of self-attention. q, k, and v denote the query, key, and value, respectively. "Linear" denotes the fully connected neural network layers. For multi-head self-attention, each black box denotes one head and gives a representation. All the representations are concatenated through the concatenate layer and input into the linear layer.

$\mathbf{X}$ through self-attention. The whole calculation process can be divided into the following two steps:

1. Calculate the attention scores matrix $\mathbf{A} \in \mathbb{R}^{N\times N}$.

$$\mathbf{A} := \mathbf{X}\mathbf{W}_{\text{qry}}(\mathbf{X}\mathbf{W}_{\text{key}})^{\top}, \tag{1}$$

   where $\mathbf{W}_{\text{qry}}, \mathbf{W}_{\text{key}} \in \mathbb{R}^{C_{\text{in}}\times C_{\text{h}}}$ represent query and key matrices respectively. $\mathbf{A}_{i,j}$ represents the correlation between the i-th item and the j-th item of the input.

2. Get the output through softmax and summation.

$$\mathbf{Y} = \text{SA}(\mathbf{X}) := \text{softmax}_{[\,,:\,]}(\mathbf{A})\mathbf{X}\mathbf{W}_{\text{val}}, \tag{2}$$

   where $\mathbf{W}_{\text{val}} \in \mathbb{R}^{C_{\text{in}}\times C_{\text{h}}}$ represents value matrix.

In practical application, Multi-Headed Self-Attention (MHSA) that focuses on different aspects of the input is applied. The outputs of different heads of dimension C_{h} are concatenated firstly and then projected to output via a projection matrix $\mathbf{W}_{\text{out}} \in \mathbb{R}^{HC_{\text{h}}\times C_{\text{out}}}$. The H denotes the number of heads.

$$\text{MHSA}(\mathbf{X}) := \underset{h\in[H]}{\text{concat}}\,[\text{SA}^{(h)}(\mathbf{X})]\mathbf{W}_{\text{out}}. \tag{3}$$

According to the above, we may define attention score matrix (Eq. 1) without positional encoding as below:

$$\mathbf{A}_{i,j} = \alpha[f](i,j) = \langle \varphi_{\text{qry}}(f(i)), \varphi_{\text{key}}(f(j))\rangle\,. \tag{4}$$

The function $\alpha[f] : \mathcal{S}\times\mathcal{S} \to \mathbb{R}$ maps pairs of set elements $i, j \in \mathcal{S}$ to an attention score $\mathbf{A}_{i,j}$. Consequently, the self-attention (Eq. 2) can be represented as below:

$$\begin{aligned}\mathbf{Y}_{i,:} = \zeta[f](i) &= \sum_{j\in\mathcal{S}} \sigma_j(\alpha[f](i,j))\varphi_{\text{val}}(f(j)) \\ &= \sum_{j\in\mathcal{S}} \sigma_j(\langle\varphi_{\text{qry}}(f(i)), \varphi_{\text{key}}(f(j))\rangle)\varphi_{\text{val}}(f(j)),\end{aligned} \tag{5}$$

where $\zeta[f] : \mathcal{S} \to \mathbb{R}^{C_{\text{h}}}$, $\sigma = \text{softmax}$, and $\sigma_j = \frac{e^{zj}}{\sum_{i=1}^{N} e^{zi}}$. Similarly, the MHSA(Eq. 3) can be expressed as below:

$$\begin{aligned}\text{MHSA}(\mathbf{X}_i) = m[f](i) &= \varphi_{out}(\bigcup_{h\in[H]} \zeta^{(h)}[f](i)) \\ &= \varphi_{out}(\bigcup_{h\in[H]}\sum_{j\in\mathcal{S}} \sigma_j(\Big\langle \varphi^{(h)}_{\text{qry}}(f(i)), \\ &\qquad \varphi^{(h)}_{\text{key}}(f(j))\Big\rangle)\varphi^{(h)}_{\text{val}}(f(j))),\end{aligned} \tag{6}$$

where $\cup$ is the concatenation operator and $\mathcal{S} \to \mathbb{R}^{C_{\text{out}}}$.

To handle the quadratic time complexity of the self-attention, ViT only uses the regions on an image nearest to the i_{th} item, when calculating the output of the i_{th} item. Let $\eta_{(i)}$ be the selected part related to the i_{th} item, which is also called the local neighbourhood of the token i in the later section. Therefore, replacing $\mathcal{S}$ with $\eta_{(i)}$, Eq. 6 can be written as below:

$$\begin{aligned}\text{MHSA}(\mathbf{X}_i) = m[f](i) &= \varphi_{out}(\bigcup_{h\in[H]} \zeta^{(h)}[f](i)) \\ &= \varphi_{out}(\bigcup_{h\in[H]}\sum_{j\in\eta(i)} \sigma_j(\Big\langle \varphi^{(h)}_{\text{qry}}(f(i)), \\ &\qquad \varphi^{(h)}_{\text{key}}(f(j))\Big\rangle)\varphi^{(h)}_{\text{val}}(f(j))).\end{aligned} \tag{7}$$

3.3 POSITIONAL ENCODING

The self-attention overlooks structural information. To solve this issue, positional encoding $\mathbf{P}$ is proposed [Dosovitskiy et al., 2020], as introduced below.

Absolute Positional Encoding In absolute positional encoding, every position is given a unique positional encoding. The whole positional encoding can be represented by a matrix $\mathbf{P} \in \mathbb{R}^{N\times C_{\text{in}}}$. Consequently, the attention scores matrix $\mathbf{A}$ can be formulated as follows:

$$\mathbf{A} := (\mathbf{X}+\mathbf{P})\mathbf{W}_{\text{qry}}((\mathbf{X}+\mathbf{P})\mathbf{W}_{\text{key}})^{\top}. \tag{8}$$

The positional encoding is a function $\rho : \mathcal{S} \to \mathbb{R}^{C_{\text{in}}}$ that maps set elements $i \in \mathcal{S}$ to a vector representation. Using this definition, Eq. 8 can be written as:

$$\begin{aligned}m[f,p](i) = \varphi_{out}(\underset{h\in[H]}{\cup}\sum_{j\in\eta(i)} \sigma_j(\Big\langle \varphi^{(h)}_{\text{qry}}(f(i)+\rho(i)), \\ \varphi^{(h)}_{\text{key}}(f(j)+\rho(j))\Big\rangle)\varphi^{(h)}_{\text{val}}(f(j)))\end{aligned} \tag{9}$$

Relative Positional Encoding Proposed by Shaw et al. [2018], relative positional encoding considers the relative distance between the query token i and the key token j. The corresponding attention score $\mathbf{A}_{i,j}$ can be calculated by the following formula:

$$\mathbf{A}^{\text{rel}}_{i,j} := \mathbf{X}_i \mathbf{W}_{\text{qry}}((\mathbf{X}_j + \mathbf{P}_{x(j)-x(i)})\mathbf{W}_{\text{key}})^\top, \quad (10)$$

where $x(i)$ is the position of token i, and $\mathbf{P}_{x(j)-x(i)} \in \mathbb{R}^{1\times C_{\text{in}}}$ is the positional encoding of the relative distance of token i and token j. Similarly, relative positional encoding can be defined as $\rho(i,j) = \rho^P(x(j) - x(i))$ of pairs $(i,j), i \in \mathcal{S}, j \in \eta(i)$. Thus, the Eq. 10 can be written as:

$$m[f,p](i) = \varphi_{out}(\bigcup_{h\in[H]} \sum_{j\in\eta(i)} \sigma_j(\Big\langle \varphi^{(h)}_{\text{qry}}(f(i)), \varphi^{(h)}_{\text{key}}(f(j) + \rho(i,j))\Big\rangle)\varphi^{(h)}_{\text{val}}(f(j))) \quad (11)$$

4 GROUP EQUIVARIANCE

This section presents necessary definitions and notations in group representation theory and group equivariance properties.

4.1 GROUP REPRESENTATION THEORY

Group A group is an abstract mathematical concept. Formally a group $(G;\circ)$ consists of a set G and a binary composition operator $\circ : G \times G \to G$. All groups must adhere to the following 4 axioms:

1. Closure: $g \circ h \in G$ for all $g, h, \in G$.
2. Associativity: $f \circ (g \circ h) = (f \circ g) \circ h = f \circ g \circ h$ for all $f, g, h \in G$.
3. Identity: There exists an element such that $e \circ g = g \circ e = g$ for all $g \in G$.
4. Inverses: For each $g \in G$ there exists a $g^{-1} \in G$ such that $g^{-1} \circ g = g \circ g^{-1} = e$.

Each group element $g \in G$ corresponds to a symmetry transformation. In practice, the binary composition operator $\circ$ can be omitted. Groups can be finite or infinite, countable or uncountable, compact or non-compact. Note that they are not necessarily commutative; that is, $gh \neq hg$ in general. If a group is commutative, that is $gh = hg$ for all $g, h \in G$, it is called the Abelian Group. One example of the infinite group is $E(2)$, the set of all 2D rotations about the origin and the 2D translation. Because the image is transformed in 2D, $E(2)$ is the focus of this paper.

Group Action A group action $\rho(g)$ is a bijective map from a space into itself: $\rho(g) : \mathcal{X} \to \mathcal{X}$. It is parameterized by an element g of a group G. For $\rho(g)x$, we say that $\rho(g)$ acts on x. A symmetry transformation of group element $g \in G$ on object $x \in \mathcal{X}$ is referred to as the group action of G on $\mathcal{X}$. $\rho(g)x$ is often written as gx to reduce clutter. In the context of group equivariant neural networks, grouping action object $\mathcal{X}$ is commonly defined to be the space of scalar-valued functions or vector-valued functions on some set $\mathcal{E}$, so that $\mathcal{X} = \{f | f : \mathcal{E} \to \mathbb{R}^d\}$. This set $\mathcal{E}$ could be a Euclidean space, e.g., a grey-scale image can be expressed as a feature map $f : \mathbb{R}^2 \to \mathbb{R}$ from pixel coordinate x_i to pixel intensity f_i, supported on the grid of pixel coordinates.

Group Representation A group representation $\rho : G \to GL(N)$ is a map from a group G to the set of $N \times N$ invertible matrices $GL(N)$. Critically, ρ is a group homomorphism, i.e., it satisfies the following property:

$$\rho(g1 \circ g2) = \rho(g1)\rho(g2), \quad \forall g1, g2 \in G. \quad (12)$$

For $SO(2)$, the standard rotation matrix is an example of a representation that acts on $\mathbb{R}^2$:

$$\rho(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}.$$

Accordingly, the rotation of the image can be expressed as a representation of $SO(2)$ by extending the action ρ on the pixel coordinates x to a representation π that acts on the space of feature maps $\{f | f : \mathcal{E} \to \mathbb{R}^d\}$:

$$[\pi(g)(f)](x) \triangleq f(\rho(g^{-1})x),$$

where $\mathcal{E} = \{x_i\}$. We can write gx instead of $\rho(g)x$ to reduce clutter:

$$[\pi(g)(f)](x) \triangleq f(g^{-1}x).$$

And it is equivalent to the mapping:

$$(x_i, f_i)_{i=1}^n \to (\rho(g)x_i, f_i)_{i=1}^n,$$

where n is the total number of pixels in the image.

Affine Group Affine groups have the following form: $\mathcal{G} = \mathbb{R}^d \rtimes \mathcal{H}$. It is resulted from the semi-direct product $(\rtimes)$ between the translation group $(\mathbb{R}^d, +)$ and an group $\mathcal{H}$ that acts on $\mathbb{R}^d$. $\mathcal{H}$ can be rotation, mirroring, etc.

Group Equivariance A map $\Phi : V_1 \to V_2$ is G-equivariant with respect to actions $\rho 1$, $\rho 2$ of G acting on V_1, V_2 respectively if:

$$\Phi[\rho 1(g)f] = \rho 2(g)[\Phi[f]], \quad \forall g \in G, f \in V_1. \quad (13)$$

As is well-known, convolution is an equivariant map for the translation group.

Lifting We can view $\mathcal{X}$ as a quotient group G/H for some subgroup H of a group G, which means $\mathcal{X}$ is isomorphic to G/H. Then, naturally, the function f defined on $\mathcal{X}$ can

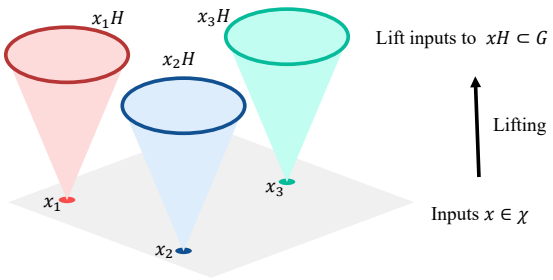


Figure 2: The illustration of lifting. For any $x \in \mathcal{X}$, $f(x)$ equals to $\mathcal{L}(f)(g)$ on G, where $g \in xH$, $\mathcal{L}$ is the lifting operation, and f is a function defined on $\mathcal{X}$.

be viewed as defined on G/H. Thus, we define the lifting operation $\mathcal{L}$ on the function f as below,

$$\mathcal{L}(f)(g) = f([g]),$$

where $[g] \in G/H$ is the equivalent class of g. For example, $\mathbb{R}^2$ is isomorphic to $SE(2)/SO(2)$, and every element $g \in SE(2)$ can be written as tr uniquely, where $t \in \mathbb{R}^2$ and $r \in SO(2)$. Furthermore, for any function f on $\mathbb{R}^2$, the lifting function $\mathcal{L}(f)$ is defined as $\mathcal{L}(f)(g) = f(t)$.

Equivariance of Self-Attention There are several important results about the equivariance of self-attention which have been proved by Romero and Cordonnier [2020]:

1. The global self-attention formulation without positional encoding (Eq. 3) is permutation equivariant.
2. Absolute position-aware self-attention (Eq. 8) is neither permutation nor translation equivariant.
3. Relative position-aware self-attention (Eq. 10) is translation equivariant.

Our model covers $SE(2)$- and $E(2)$-equivariance, which correspond to (1) translational and rotational equivariance, and (2) translational, rotational, and reflection equivariance, respectively.

5 GROUP EQUIVARIANT VISION TRANSFORMER

In this section, we recall the lifting and the group self-attention of the GSA-Nets [Romero and Cordonnier, 2020], and then point out that the positional encoding wherein is not effective to preserve rotation equivariance. To address this issue, a modified version of the positional encoding is proposed.

To design an equivariant network, there are usually two choices of group representation: irreducible representation and regular representation. The experimental results [Fuchs et al., 2020, Hutchinson et al., 2021, Weiler and Cesa, 2019] show that regular representation is more expressive and Ravanbakhsh [2020] has theoretically proved it. A lifting self-attention layer is an essential module to obtain feature representation based on regular representation. The main function of the lifting layer is mapping $f_{\mathcal{X}}$ (a function defined on $\mathbb{R}^d$) to $\mathcal{L}[f_{\mathcal{X}}]$ (a function defined on G). After the lifing layer, the feature has been defined on the group G, which brings practical implementation problems that the group G is infinite. The summation over group elements $g \in G$ is an essential step. Fortunately, extensive experiments [Weiler and Cesa, 2019] have shown that networks using regular representations can achieve satisfactory results via proper discrete approximations.

5.1 LIFTING SELF-ATTENTION

As previously mentioned, the lifting self-attention is a map from functions on $\mathbb{R}^d$ to functions on $\mathcal{G}$ and can be expressed as: $m^r_{\mathcal{G}\uparrow}[f,\rho] : L_{\mathcal{V}}(\mathbb{R}^d) \to L_{\mathcal{V}'}(\mathcal{G})$, where $\mathcal{G}$ is an affine group and $\mathcal{G} = \mathbb{R}^d \rtimes \mathcal{H}$. The action of group element $\hbar \in \mathcal{H}$ on relative positional encoding $\rho(i,j)$ is defined as: $\{\mathcal{L}_\hbar[\rho](i,j)\}_{\hbar\in\mathcal{H}}$, $\mathcal{L}_\hbar[\rho](i,j) = \rho^P(\hbar^{-1}x(j) - \hbar^{-1}x(i))$. Consequently, the formula of lifting self-attention can be expressed as:

$$\begin{aligned} m^r_{\mathcal{G}\uparrow}[f,\rho](i,\hbar) &= m^r[f,\mathcal{L}_\hbar[\rho]](i) \\ &= \varphi_{out}\Big(\bigcup_{h\in[H]}\sum_{j\in\eta(i)} \sigma_j\Big(\Big\langle \varphi^{(h)}_{\text{qry}}(f(i)), \varphi^{(h)}_{\text{key}}(f(j)) + \\ &\qquad \mathcal{L}_\hbar[\rho](i,j)\Big)\Big\rangle\Big)\varphi^{(h)}_{\text{val}}(f(j))\Big). \end{aligned} \tag{14}$$

It has been proven that the lifting self-attention defined above is equivariant to the affine group $\mathcal{G}$ [Romero and Cordonnier, 2020].

5.2 GROUP SELF-ATTENTION

After the lifting self-attention layer, the feature map can be viewed as a function defined on $\mathcal{G}$. So the action of group elements $\hbar \in \mathcal{H}$ on relative positional encoding $\rho(i,j)$ is defined as: $\{\mathcal{L}_\hbar[\rho]((i,\tilde{h}),(j,\hat{h}))\}_{\hbar\in\mathcal{H}}$. The positional encoding used in [Romero and Cordonnier, 2020] is:

$$\rho((i,\tilde{h}),(j,\hat{h})) = \rho^P(x(j) - x(i), \tilde{h}^{-1}\hat{h}). \tag{15}$$

Therefore, the group action on relative positional encoding can be expressed as:

$$\begin{aligned} \{\mathcal{L}_\hbar[\rho]((i,\tilde{h}),(j,\hat{h}))\}_{\hbar\in\mathcal{H}} = \rho^P(&\hbar^{-1}(x(j) - x(i)), \\ &\hbar^{-1}(\tilde{h}^{-1}\hat{h})). \end{aligned}$$

Similar to the lifting self-attention layer, the formula of group self-attention can be expressed as:

$$m^r_{\mathcal{G}}[f,\rho](i,\hbar) = \sum_{\tilde{h}\in\mathcal{H}} m^r[f,\mathcal{L}_{\hbar}[\rho]](i,\tilde{h}) \tag{16}$$

$$= \varphi_{out}\Big(\bigcup_{h\in[H]}\sum_{\tilde{h}\in\mathcal{H}}\sum_{(j,\hat{h})\in\eta(i,\tilde{h})}\sigma_{j,\hat{h}}\Big(\Big\langle\varphi^{(h)}_{\text{qry}}(f(i,\tilde{h})),$$
$$\varphi^{(h)}_{\text{key}}(f(j,\hat{h})+\mathcal{L}_{\hbar}[\rho]((i,\tilde{h}),(j,\hat{h})))\Big\rangle\Big)\varphi^{(h)}_{\text{val}}(f(j,\hat{h}))\Big). \tag{17}$$

However, we prove the group self-attention using the positional encoding defined as Eq. 15 is not $\mathcal{G}$-equivariant. That is,

$$m^r_{\mathcal{G}}[\mathcal{L}_g[f],\rho](i,\hbar) \neq \mathcal{L}_g[m^r_{\mathcal{G}}[f,\rho]](i,\hbar), \quad g\in\mathcal{G}.$$

In order to make the module satisfy the equivariant property, we propose a novel positional encoding to replace the old one (Eq. 15):

$$\rho((i,\tilde{h}),(j,\hat{h})) = \rho^P(x(j)-x(i),\tilde{h}\hat{h}^{-1}\tilde{h}). \tag{18}$$

Correspondingly, the group action on relative positional encoding can be expressed as:

$$\mathcal{L}_{\hbar}[\rho]((i,\tilde{h}),(j,\hat{h})) = \rho^P(\hbar^{-1}(x(j)-x(i)),\hbar^{-1}(\tilde{h}\hat{h}^{-1}\tilde{h})).$$

It can be proven (Appendix A) that using the modified version of positional encoding (Eq. 18), the group self-attention is $\mathcal{G}$-equivariant. That is,

$$m^r_{\mathcal{G}}[\mathcal{L}_g[f],\rho](i,\hbar) = \mathcal{L}_g[m^r_{\mathcal{G}}[f,\rho]](i,\hbar), \quad g\in\mathcal{G}.$$

5.3 GE-VIT

Fig. 3 shows the structure of our GE-ViT, which is modified from GSA-Nets [Romero and Cordonnier, 2020]. The core modules of the GE-ViT and GSA-Nets are the lifting self-attention and group self-attention, the computation details can be found in GSA-Nets [Romero and Cordonnier, 2020]. Linear map, layer normalization, and activation function are interspersed in the model. Following Dosovitskiy et al. [2020], Liu et al. [2021], Romero and Cordonnier [2020], the Global Pooling block, in the end, consists of max-pool over group elements followed by spatial mean-pool. In our experiments, we choose the local self-attention because of the computational constraints. The neighborhood size $n \times n$ denotes the chosen size of the local region. Following [Romero and Cordonnier, 2020], rotation equivariant models are notated as **Rn**, where **n** represents the angle discretization. Specifically speaking, $\mathbf{R4_SA}$ depicts a model equivariant to rotations by 90 degrees and $\mathbf{R8_SA}$ depicts a model equivariant to rotations by 45 degrees.

6 EXPERIMENTS

We conduct a study on standard benchmark datasets, rotMNIST, CIFAR-10, and PATCHCAMELYON to evaluate the performance of GE-ViT compared with GSA-Nets.

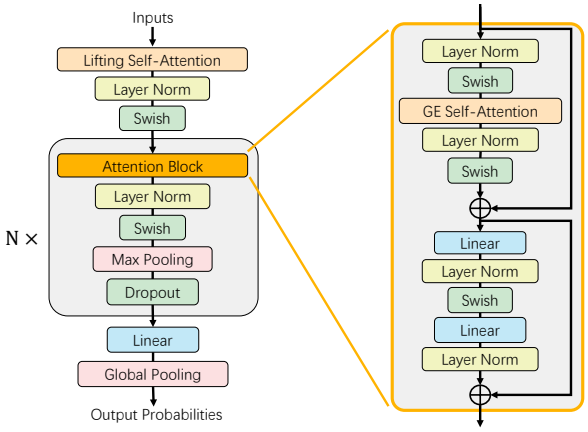


Figure 3: Illustration of GE-ViT and Attention Block. The blocks on the left show the structure of GE-ViT. Functions are transformed from R2 to Group through Lifting Self-Attention. N denotes the number of blocks in the black box. The Global Pooling block consists of max-pool over group elements followed by spatial mean-pool. Swish is an activation function [Ramachandran et al., 2017]. The flow on the right illustrates the structure of the Attention Block. Linear denotes the fully connected neural network layers. GE Self-Attention contains lifting self-attention and group self-attention.

6.1 EXPERIMENT SETUP

Dataset RotMNIST dataset is constructed by rotating the MNIST dataset. It is a classification dataset often used as a standard benchmark for rotation equivariance [Weiler and Cesa, 2019]. RotMNIST contains 62,000 gray-scale 28×28 uniformly rotated handwritten digits. The rotMNIST has been divided into training, validation, and test sets of 10k, 2k, and 50k images. CIFAR-10 dataset [Krizhevsky et al., 2009] consists of 60,000 real-world 32×32 RGB images uniformly drawn from 10 classes. PATCHCAMELYON dataset [Veeling et al., 2018] includes 327,000 96×96 RGB image patches of tumorous/non-tumorous breast tissues.

Compared Approaches Following [Romero and Cordonnier, 2020], we compare our GE-ViT with $\mathbf{Z2_SA}$ and **GSA-Nets**. $\mathbf{Z2_SA}$ is a translation equivariant self-attention model. **GSA-Nets** is also a self-attention-based model, which tried to introduce more kinds of equivariance to $\mathbf{Z2_SA}$.

6.2 IMPLEMENTATION DETAILS

This section gives the implementation details of the experiments.

Invariant Network The invariant network is a special case of the equivariant network. The function composited

Table 1: Classification accuracy (%) of R4_SA with different neighborhood size on rotMNIST.

MODEL	GSA-Nets	GE-ViT (ours)
3×3	96.28	**96.63**
5×5	97.47	**97.58**
7×7	97.33	**97.45**
9×9	97.10	**97.15**
11×11	97.06	**97.16**
15×15	96.89	**97.12**
19×19	96.86	**97.37**
23×23	96.90	**97.01**

Table 2: Classification accuracy (%) of different equivariant models on rotMNIST. All architectures based on self-attention use 5×5 neighborhood size.

MODEL	GSA-Nets	GE-ViT (ours)
Z2_SA	96.63	
R4_SA	97.46	**97.58**
R8_SA	97.79	**97.88**
R12_SA	97.97	**98.01**
R16_SA	97.66	**97.83**

Table 3: Classification accuracy (%) on the PATCHCAMELYON dataset.

MODEL	GSA-Nets	GE-ViT (ours)
Z2_SA (ViT)	80.14	80.14
R4_SA	79.40	82.73
R8_SA	82.26	83.82

of several equivariant functions followed by an invariant function f, is an invariant function [Hutchinson et al., 2021]. Therefore, the Gobal Pooling layer, an invariant map, is added to the end of the GE-ViT and GSA-Nets in our experiments.

Hyperparameters Setting To ensure fairness, the hyperparameters remain fixed for all experiments. The number of epochs is 300 and the batch size is 8. The learning rate is set to 0.001 and the weight decay is set to 0.0001. Attention dropout rate and value dropout rate are both set to 0.1. Adam optimizer is applied.

6.3 EXPERIMENTS AND RESULTS

Experiments are conducted to compare our GE-ViT with previous methods. Table 1 shows the classification results of **R4_SA** with different neighborhood size. Table 2 show the classification results of different equivariant models with 5×5 neighborhood size. Table 3 shows the classification results on PATCHCAMELYON dataset. The classification

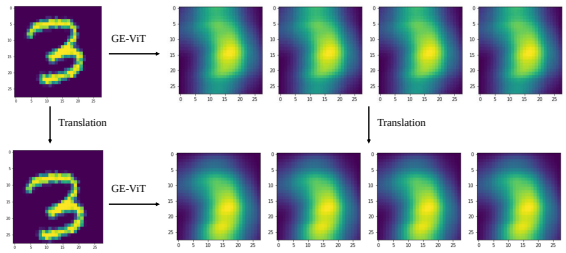


Figure 4: Translation equivariance of GE-ViT. The images on the left are the raw data and the images on the right are feature representations. Specifically speaking, feature representations of the original data are shown in the top right of the image, and feature representations obtained by translating the original data are in the lower right of the image.

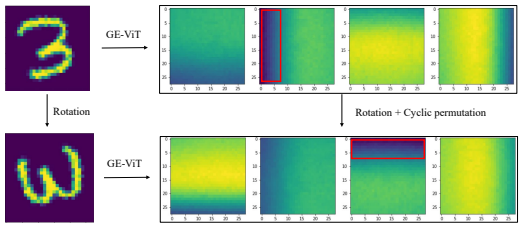


Figure 5: Rotation equivariance of GE-ViT. The images on the left are the raw data and the images on the right are feature representations. Specifically speaking, feature representations of the original data are shown in the top right of the image, and feature representations obtained by rotating the original data are in the lower right of the image.

results of GE-ViT and GSA-Nets on CIFAR-10 are 70.40% and 69.31% respectively. The reported performance of GSA-Nets is reproduced from the official released code (GSA-Nets).

It is clearly observed from the experimental results that our GE-ViT outperforms other methods consistently in all settings. With more kinds of equivariance, GSA-Nets beats Z2_SA on most settings since some errors exist in GSA-Nets. Our novel positional encoding improves the classification accuracy in GE-ViT and makes a new state-of-the-art. Besides, **R4_SA** with the neighborhood size of 5×5 achieves the best accuracy. This finding is also available in [Romero and Cordonnier, 2020]. Since in the whole experiment, only the positional encodings are different and the rest remains the same, the experimental results can demonstrate the superiority of the positional encoding we proposed. The results clearly show that our GE-ViT significantly outperforms existing methods.

The translation, rotation, and reflection equivariances of our GE-ViT are shown visually in Fig. 4, Fig. 5, and Fig. 6 respectively.

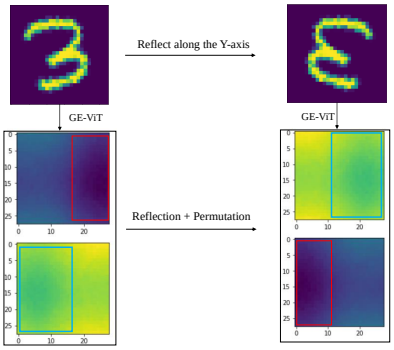


Figure 6: Reflection equivariance of GE-ViT. The images on the top are the raw data and the images on the bottom are feature representations. Specifically speaking, feature representations of the original data are shown in the lower left of the image, and feature representations obtained by flipping the original data are in the lower right of the image.

Table 4: Comparison with equivariant convolutional networks on rotMNIST

Model	ACC(%)	Model	ACC(%)
Z2_SA	96.63%	R16_SA	97.83%
R4_SA	97.58%	Z2-CNN	95.14%
R8_SA	97.88%	P4-CNN	98.21%
R12_SA	98.01%	α-P4-CNN	98.31%

We compare with equivariant convolutional networks. Following GSA_Net [Romero and Cordonnier, 2020], we compare our GE-ViT with classic convolutional networks (Z2CNN) Cohen and Welling [2016a] and equivariant convolutional networks that incorporate attention mechanisms (P4-CNN, Alpha-P4-CNN) [Romero et al., 2020]. The size of the convolutional kernel is 3, and the settings for the other hyperparameters follow the original paper. The experimental results on the rotMNIST dataset are shown in the Table 4, from which, we can draw two conclusions:

- Z2_SA performs better than Z2CNN, which demonstrates the potential of equivariant attention networks on image classification tasks. This is consistent with the conclusion in Section 5.3 of the paper [Romero and Cordonnier, 2020] that attention-based equivariant networks theoretically outperform convolution-based equivariant networks.
- Although our GE-ViT achieves comparable performance with equivariant convolutional networks, there is still a slight gap between them. As is mentioned in GSA-Nets [Romero and Cordonnier, 2020], there are two reasons. Firstly, the number of parameters for GE-ViT is approximately 45,000, while the number of parameters for G-CNN is around 75,000. The smaller number of parameters limit the expressiveness of the model. Secondly, although GE-ViT is theoretically superior, it is more difficult to optimize [Liu et al., 2020, Zhao et al., 2020]. With further research on optimization issues related to attention mechanisms, the performance of GE-ViT would gain a significant improvement.

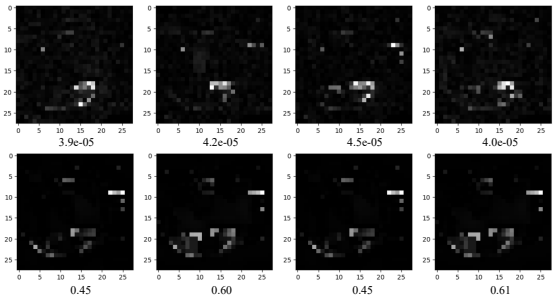


Figure 7: Error maps of GE-ViT and GSA-Net. The numbers between under the image are the average error. Images in the first and second rows are the error maps of GE-ViT and GSA-Net respectively.

We also conducted additional experiments to compare our GE-ViT with CPVT [Chu et al., 2021], which proposed a novel positional encoding method. The accuracy of CPVT on RotMNIST is 97.69% which is worse than our GE-ViT since the positional encoding in CPVT is not equivariant.

Fig. 7 shows the visual comparison between GSA-Net and GE-ViT. We visualize the errors of the feature maps in GE-ViT and GSA-Nets. Like Fig. 5, we visualize the error between corresponding feature maps. The process of our visualization is as follows: (1) Given an image, we extracted feature maps F_{GE} and F_{GSA} from GE-ViT and GSA-Nets respectively. (2) Then, we rotated the image and extracted feature maps F'_{GE} and F'_{GSA} from GE-ViT and GSA-Nets, respectively. (3) By rotating and cyclic permutating the feature maps F_{GE} and F_{GSA}, we obtained feature maps F''_{GE} and F''_{GSA} which are the ground truths of the feature maps of rotated image. (4) Finally, we got error maps $E_{GE} = F''_{GE} - F'_{GE}$, $E_{GSA} = F''_{GSA} - F'_{GSA}$ as shown in Fig. 7 which shows that our GE-ViT performs better. The average errors of our GE-ViT are on the order of 10^{-5} while the average errors of GSA-Nets are on the order of 10^{-1}.

7 DISCUSSION

GE-ViT with a novel and effective positional encoding outperforms GSA-Nets and non-equivariant self-attention networks are competitive to G-CNNs. However, G-CNNS still performs better on most data sets, which may be due to the optimization problem of GE-ViT or the limits on computing resources. From the theoretical perspective, the group equivariant self-attention can be more expressive than G-CNNS, so the GE-ViT has a lot of potential for improvement in the aspect of initialization, optimization, generalization, etc.

References

E. J. Bekkers. B-spline cnns on lie groups. In *International Conference on Learning Representations*, 2019.

N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020.

X. Chu, Z. Tian, B. Zhang, X. Wang, X. Wei, H. Xia, and C. Shen. Conditional positional encodings for vision transformers. *arXiv preprint arXiv:2102.10882*, 2021.

T. Cohen and M. Welling. Group equivariant convolutional networks. In *International conference on machine learning*, pages 2990–2999. PMLR, 2016a.

T. S. Cohen and M. Welling. Steerable cnns. *arXiv preprint arXiv:1612.08498*, 2016b.

T. S. Cohen, M. Geiger, J. Köhler, and M. Welling. Spherical cnns. In *International Conference on Learning Representations*, 2018.

T. S. Cohen, M. Geiger, and M. Weiler. A general theory of equivariant cnns on homogeneous spaces. *Advances in neural information processing systems*, 32:9142–9153, 2019.

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

J. L. Elman. Finding structure in time. *Cognitive science*, 14(2):179–211, 1990.

C. Esteves, C. Allen-Blanchette, A. Makadia, and K. Daniilidis. Learning so (3) equivariant representations with spherical cnns. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 52–68, 2018.

F. A. Faber, A. Lindmaa, O. A. Von Lilienfeld, and R. Armiento. Machine learning energies of 2 million elpasolite (a b c 2 d 6) crystals. *Physical review letters*, 117(13):135502, 2016.

M. Finzi, S. Stanton, P. Izmailov, and A. G. 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.

F. Fuchs, D. Worrall, V. Fischer, and M. Welling. Se (3)-transformers: 3d roto-translation equivariant attention networks. *Advances in Neural Information Processing Systems*, 33:1970–1981, 2020.

F. He and D. Tao. Recent advances in deep learning theory. *arXiv preprint arXiv:2012.10931*, 2020.

M. J. Hutchinson, C. Le Lan, S. Zaidi, E. Dupont, Y. W. Teh, and H. Kim. Lietransformer: Equivariant self-attention for lie groups. In *International Conference on Machine Learning*, pages 4533–4543. PMLR, 2021.

J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Žídek, A. Potapenko, et al. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, 2021.

A. Krizhevsky, G. Hinton, et al. Learning multiple layers of features from tiny images. 2009.

A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.

Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551, 1989.

Y. Li, R. Bu, M. Sun, W. Wu, X. Di, and B. Chen. Pointcnn: Convolution on x-transformed points. *Advances in neural information processing systems*, 31, 2018.

K. Liu, K. Yang, J. Zhang, and R. Xu. S2snet: A pretrained neural network for superconductivity discovery. In L. D. Raedt, editor, *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pages 5101–5107. International Joint Conferences on Artificial Intelligence Organization, 7 2022. doi: 10.24963/ijcai.2022/708. URL `https://doi.org/10.24963/ijcai.2022/708`. AI for Good.

L. Liu, X. Liu, J. Gao, W. Chen, and J. Han. Understanding the difficulty of training transformers. *arXiv preprint arXiv:2004.08249*, 2020.

Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021.

D. Marcos, M. Volpi, N. Komodakis, and D. Tuia. Rotation equivariant vector field networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5048–5057, 2017.

M. Ntampaka, H. Trac, D. J. Sutherland, S. Fromenteau, B. Póczos, and J. Schneider. Dynamical mass measurements of contaminated galaxy clusters using machine learning. *The Astrophysical Journal*, 831(2):135, 2016.

T. Qin, F. He, D. Shi, H. Wenbing, and D. Tao. Benefits of permutation-equivariance in auction mechanisms. In *Advances in Neural Information Processing Systems*, 2022.

P. Ramachandran, B. Zoph, and Q. V. Le. Searching for activation functions. *arXiv preprint arXiv:1710.05941*, 2017.

S. Ravanbakhsh. Universal equivariant multilayer perceptrons. In *International Conference on Machine Learning*, pages 7996–8006. PMLR, 2020.

D. Romero, E. Bekkers, J. Tomczak, and M. Hoogendoorn. Attentive group equivariant convolutional networks. In *International Conference on Machine Learning*, pages 8188–8199. PMLR, 2020.

D. W. Romero and J.-B. Cordonnier. Group equivariant stand-alone self-attention for vision. In *International Conference on Learning Representations*, 2020.

D. W. Romero and M. Hoogendoorn. Co-attentive equivariant neural networks: Focusing equivariance on transformations co-occurring in data. In *International Conference on Learning Representations*, 2019.

A. Sannai, M. Imaizumi, and M. Kawano. Improved generalization bounds of group invariant/equivariant deep networks via quotient feature spaces. In *Uncertainty in Artificial Intelligence*, pages 771–780. PMLR, 2021.

P. Shaw, J. Uszkoreit, and A. Vaswani. Self-attention with relative position representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 464–468, 2018.

N. Thomas, T. Smidt, S. Kearnes, L. Yang, L. Li, K. Kohlhoff, and P. Riley. Tensor field networks: Rotation- and translation-equivariant neural networks for 3d point clouds. *arXiv preprint arXiv:1802.08219*, 2018.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30:5998–6008, 2017.

B. S. Veeling, J. Linmans, J. Winkens, T. Cohen, and M. Welling. Rotation equivariant cnns for digital pathology. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2018: 21st International Conference, Granada, Spain, September 16-20, 2018, Proceedings, Part II 11*, pages 210–218. Springer, 2018.

S. R. Venkataraman, S. Balasubramanian, and R. R. Sarma. Building deep equivariant capsule networks. In *International Conference on Learning Representations*, 2019.

M. Weiler and G. Cesa. General e(2)-equivariant steerable cnns. In *Advances in Neural Information Processing Systems*, pages 14334–14345, 2019.

M. Weiler, F. A. Hamprecht, and M. Storath. Learning steerable filters for rotation equivariant cnns. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 849–858, 2018.

D. E. Worrall, S. J. Garbin, D. Turmukhambetov, and G. 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.

M. Zaheer, S. Kottur, S. Ravanbakhsh, B. Poczos, R. R. Salakhutdinov, and A. J. Smola. Deep sets. *Advances in neural information processing systems*, 30, 2017.

H. Zhao, J. Jia, and V. Koltun. Exploring self-attention for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10076–10085, 2020.

Y. Zhou, Q. Ye, Q. Qiu, and J. Jiao. Oriented response networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 519–528, 2017.

A PROOF

In this appendix, we prove that GE-ViT is group equivariant. For brevity, we also use the substitutions:

$$\bar{i} = x^{-1}(\bar{\hbar}^{-1}(x(i) - y)) \Rightarrow i = x^{-1}(\bar{\hbar}x(\bar{i}) + y)), \tilde{\hbar}' = \bar{\hbar}^{-1}\tilde{\hbar},$$

and

$$\bar{j} = x^{-1}(\bar{\hbar}^{-1}(x(j) - y)) \Rightarrow j = x^{-1}(\bar{\hbar}x(\bar{j}) + y)), \hat{\hbar}' = \bar{\hbar}^{-1}\hat{\hbar}.$$

A.1 DEFINITIONS AND NOTATIONS.

A.1.1 Definition of Group Equivariant Self-Attention.

If the group self-attention formulation $m^r_{\mathcal{G}}[f, \rho](i, \hbar)$ is $\mathcal{G}$-equivariant, if and only if it satisfies:

$$m^r_{\mathcal{G}}[\mathcal{L}_g[f], \rho](i, \hbar) = \mathcal{L}_g[m^r_{\mathcal{G}}[f, \rho]](i, \hbar), \quad g \in \mathcal{G}$$

A.1.2 Input under g-Transformed

A g-transformed input can be expressed as:

$$\mathcal{L}_g[f](i, \tilde{h}) = \mathcal{L}_y\mathcal{L}_{\bar{\hbar}}[f](i, \tilde{\hbar}) = f(\rho^{-1}(\bar{\hbar}^{-1}(\rho(i) - y)), \bar{\hbar}^{-1}\tilde{\hbar}),$$
$$g = (y, \bar{\hbar}),\ y \in \mathbb{R}^d,\ \bar{\hbar} \in \mathcal{H}.$$

A.2 PROOF OF GE-VIT

The complete proof process is as follows:

$$\begin{aligned}
&m^r_{\mathcal{G}}\big[\mathcal{L}_y\mathcal{L}_{\bar{\hbar}}[f], \rho\big](i, \hbar) && (19)\\
&= \varphi_{\text{out}}\Big(\bigcup_{h\in[H]} \sum_{\tilde{\hbar}\in\mathcal{H}} \sum_{(j,\hat{\hbar})\in\mathcal{N}(i,\tilde{\hbar})} \sigma_{j,\hat{\hbar}}\big(\langle \varphi^{(h)}_{\text{qry}}(\mathcal{L}_y\mathcal{L}_{\bar{\hbar}}[f](i, \tilde{\hbar})), \varphi^{(h)}_{\text{key}}(\mathcal{L}_y\mathcal{L}_{\bar{\hbar}}[f](j, \hat{\hbar}) \\
&\qquad + \mathcal{L}_{\hbar}[\rho]((i, \tilde{\hbar}), (j, \hat{\hbar}))\rangle\big)\varphi^{(h)}_{\text{val}}(\mathcal{L}_y\mathcal{L}_{\bar{\hbar}}[f](j, \hat{\hbar}))\Big) \\
&= \varphi_{\text{out}}\Big(\bigcup_{h\in[H]} \sum_{\tilde{\hbar}\in\mathcal{H}} \sum_{(j,\hat{\hbar})\in\mathcal{N}(i,\tilde{\hbar})} \sigma_{j,\hat{\hbar}}\big(\langle \varphi^{(h)}_{\text{qry}}(f(x^{-1}(\bar{\hbar}^{-1}(x(i) - y)), \bar{\hbar}^{-1}\tilde{\hbar})), && (20)\\
&\qquad \varphi^{(h)}_{\text{key}}(f(x^{-1}(\bar{\hbar}^{-1}(x(j) - y)), \bar{\hbar}^{-1}\hat{\hbar}) + \mathcal{L}_{\hbar}[\rho]((i, \tilde{\hbar}), (j, \hat{\hbar}))\rangle\big) \\
&\qquad\qquad \varphi^{(h)}_{\text{val}}(f(x^{-1}(\bar{\hbar}^{-1}(x(j) - y)), \bar{\hbar}^{-1}\hat{\hbar}))\Big) \\
&= \varphi_{\text{out}}\Big(\bigcup_{h\in[H]} \sum_{\bar{\hbar}\tilde{\hbar}'\in\mathcal{H}} \sum_{(x^{-1}(\bar{\hbar}x(\bar{j})+y),\bar{\hbar}\hat{\hbar}')\in\mathcal{N}(x^{-1}(\bar{\hbar}x(\bar{i})+y),\bar{\hbar}\tilde{\hbar}')} \sigma_{x^{-1}(\bar{\hbar}x(\bar{j})+y),\bar{\hbar}\hat{\hbar}'}\big(\langle \varphi^{(h)}_{\text{qry}}(f(\bar{i}, \tilde{\hbar}')), \varphi^{(h)}_{\text{key}}(f(\bar{j}, \hat{\hbar}') && (21)\\
&\quad + \mathcal{L}_{\hbar}[\rho]((x^{-1}(\bar{\hbar}x(\bar{i}) + y), \bar{\hbar}\tilde{\hbar}'), (x^{-1}(\bar{\hbar}x(\bar{j}) + y), \bar{\hbar}\hat{\hbar}'))\rangle\big)\varphi^{(h)}_{\text{val}}(f(\bar{j}, \hat{\hbar}'))\Big)
\end{aligned}$$

By using the definition:

$$\rho((i, \tilde{\hbar}), (j, \hat{\hbar})) = \rho^P(x(j) - x(i), \tilde{\hbar}\hat{\hbar}^{-1}\tilde{\hbar})$$

and

$$\mathcal{L}_{\hbar}[\rho]((i, \tilde{\hbar}), (j, \hat{\hbar})) = \rho^P(\hbar^{-1}(x(j) - x(i)), \hbar^{-1}(\tilde{\hbar}\hat{\hbar}^{-1}\tilde{\hbar})).$$

The above formula can be further derived:

$$= \varphi_{\text{out}}\Big(\bigcup_{h\in[H]} \sum_{\bar{\hbar}\tilde{\hbar}'\in\mathcal{H}} \sum_{(x^{-1}(\bar{\hbar}x(\bar{j})+y),\bar{\hbar}\hat{\hbar}')\in\mathcal{N}(x^{-1}(\bar{\hbar}x(\bar{i})+y),\bar{\hbar}\tilde{\hbar}')} \sigma_{x^{-1}(\bar{\hbar}x(\bar{j})+y),\bar{\hbar}\hat{\hbar}'}\big(\langle\varphi_{\text{qry}}^{(h)}(f(\bar{i},\tilde{\hbar}')),\varphi_{\text{key}}^{(h)}(f(\bar{j},\hat{\hbar}') \tag{22}$$
$$+\rho^P(\hbar^{-1}(\bar{\hbar}x(\bar{j})+y-(\bar{\hbar}x(\bar{i})+y)),\hbar^{-1}(\bar{\hbar}\tilde{\hbar}')(\bar{\hbar}\hat{\hbar}')^{-1}(\bar{\hbar}\tilde{\hbar}'))\rangle)\varphi_{\text{val}}^{(h)}(f(\bar{j},\hat{\hbar}'))\Big)$$

$$= \varphi_{\text{out}}\Big(\bigcup_{h\in[H]} \sum_{\bar{\hbar}\tilde{\hbar}'\in\mathcal{H}} \sum_{(x^{-1}(\bar{\hbar}x(\bar{j})+y),\bar{\hbar}\hat{\hbar}')\in\mathcal{N}(x^{-1}(\bar{\hbar}x(\bar{i})+y),\bar{\hbar}\tilde{\hbar}')} \sigma_{x^{-1}(\bar{\hbar}x(\bar{j})+y),\bar{\hbar}\hat{\hbar}'}\big(\langle\varphi_{\text{qry}}^{(h)}(f(\bar{i},\tilde{\hbar}')),\varphi_{\text{key}}^{(h)}(f(\bar{j},\hat{\hbar}') \tag{23}$$
$$+\rho^P(\hbar^{-1}(\bar{\hbar}x(\bar{j})+y-(\bar{\hbar}x(\bar{i})+y)),\hbar^{-1}\bar{\hbar}\tilde{\hbar}'\hat{\hbar}'^{-1}\tilde{\hbar}'))\rangle)\varphi_{\text{val}}^{(h)}(f(\bar{j},\hat{\hbar}'))\Big)$$

$$= \varphi_{\text{out}}\Big(\bigcup_{h\in[H]} \sum_{\bar{\hbar}\tilde{\hbar}'\in\mathcal{H}} \sum_{(x^{-1}(\bar{\hbar}x(\bar{j})+y),\bar{\hbar}\hat{\hbar}')\in\mathcal{N}(x^{-1}(\bar{\hbar}x(\bar{i})+y),\bar{\hbar}\tilde{\hbar}')} \sigma_{x^{-1}(\bar{\hbar}x(\bar{j})+y),\bar{\hbar}\hat{\hbar}'}\big(\langle\varphi_{\text{qry}}^{(h)}(f(\bar{i},\tilde{\hbar}')),\varphi_{\text{key}}^{(h)}(f(\bar{j},\hat{\hbar}') \tag{24}$$
$$+\rho^P(\hbar^{-1}\bar{\hbar}(x(\bar{j})-x(\bar{i})),\tilde{\hbar}'\hat{\hbar}'^{-1}\tilde{\hbar}'))\rangle)\varphi_{\text{val}}^{(h)}(f(\bar{j},\hat{\hbar}'))\Big)$$

$$= \varphi_{\text{out}}\Big(\bigcup_{h\in[H]} \sum_{\bar{\hbar}\tilde{\hbar}'\in\mathcal{H}} \sum_{(x^{-1}(\bar{\hbar}x(\bar{j})+y),\bar{\hbar}\hat{\hbar}')\in\mathcal{N}(x^{-1}(\bar{\hbar}x(\bar{i})+y),\bar{\hbar}\tilde{\hbar}')} \sigma_{x^{-1}(\bar{\hbar}x(\bar{j})+y),\bar{\hbar}\hat{\hbar}'}\big(\langle\varphi_{\text{qry}}^{(h)}(f(\bar{i},\tilde{\hbar}')),\varphi_{\text{key}}^{(h)}(f(\bar{j},\hat{\hbar}') \tag{25}$$
$$+\mathcal{L}_{\bar{\hbar}^{-1}\hbar}[\rho]((\bar{i},\tilde{\hbar}'),(\bar{j},\hat{\hbar}')))\rangle)\varphi_{\text{val}}^{(h)}(f(\bar{j},\hat{\hbar}'))\Big)$$

The subsequent proof is similar to the GSA-Nets [Romero and Cordonnier, 2020]. For unimodular groups, the area of summation remains equal for any transformation $g \in \mathcal{G}$, which means that:

$$\begin{aligned}\sum_{(x^{-1}(\bar{\hbar}x(\bar{j})+y),\bar{\hbar}\hat{\hbar}')\in\mathcal{N}(x^{-1}(\bar{\hbar}x(\bar{i})+y),\bar{\hbar}\tilde{\hbar}')}[\cdot] &= \sum_{(x^{-1}(\bar{\hbar}x(\bar{j})),\bar{\hbar}\hat{\hbar}')\in\mathcal{N}(x^{-1}(\bar{\hbar}x(\bar{i})),\bar{\hbar}\tilde{\hbar}')}[\cdot] \\ &= \sum_{(x^{-1}(x(\bar{j})),\hat{\hbar}')\in\mathcal{N}(x^{-1}(x(\bar{i})),\tilde{\hbar}')}[\cdot] \\ &= \sum_{(\bar{j},\hat{\hbar}')\in\mathcal{N}(\bar{i},\tilde{\hbar}')}[\cdot].\end{aligned}$$

and because of the basic properties of groups, we can get $\sum_{\bar{\hbar}\tilde{\hbar}'\in\mathcal{H}}[\cdot] = \sum_{\tilde{\hbar}'\in\mathcal{H}}[\cdot]$. Consequently, the above formula can be further simplified as:

$$\begin{aligned}m_{\mathcal{G}}^r\big[\mathcal{L}_y\mathcal{L}_{\bar{\hbar}}[f],\rho\big](i,\hbar) =& \varphi_{\text{out}}\Big(\bigcup_{h\in[H]} \sum_{\tilde{\hbar}'\in\mathcal{H}} \sum_{(\bar{j},\hat{\hbar}')\in\mathcal{N}(\bar{i},\tilde{\hbar}')} \sigma_{\bar{j},\hat{\hbar}'}\big(\langle\varphi_{\text{qry}}^{(h)}(f(\bar{i},\tilde{\hbar}')), \\ & \varphi_{\text{key}}^{(h)}(f(\bar{j},\hat{\hbar}')+\mathcal{L}_{\bar{\hbar}^{-1}\hbar}[\rho]((\bar{i},\tilde{\hbar}'),(\bar{j},\hat{\hbar}')))\rangle)\varphi_{\text{val}}^{(h)}(f(\bar{j},\hat{\hbar}'))\Big) \\ =& m_{\mathcal{G}}^r[f,\rho](\bar{i},\bar{h}^{-1}\hbar) \\ =& m_{\mathcal{G}}^r[f,\rho](x^{-1}(\bar{\hbar}^{-1}(x(i)-y)),\bar{h}^{-1}\hbar) \\ =& \mathcal{L}_y\mathcal{L}_{\bar{\hbar}}\big[m_{\mathcal{G}}^r[f,\rho]\big](i,\hbar).\end{aligned} \tag{26}$$

From the above formula, it can be seen that:

$$m_{\mathcal{G}}^r[\mathcal{L}_y\mathcal{L}_{\bar{h}}[f],\rho](i,\hbar) = \mathcal{L}_y\mathcal{L}_{\bar{h}}[m_{\mathcal{G}}^r[f,\rho]](i,\hbar),$$

which is the same as:

$$m_{\mathcal{G}}^r[\mathcal{L}_g[f],\rho](i,\hbar) = \mathcal{L}_g[m_{\mathcal{G}}^r[f,\rho]](i,\hbar), \quad g\in\mathcal{G}.$$

Therefore, with the positional encoding we proposed:

$$\rho((i,\tilde{\hbar}),(j,\hat{\hbar})) = \rho^P(x(j)-x(i),\tilde{\hbar}\hat{\hbar}^{-1}\tilde{\hbar}),$$

the group self-attention is group equivariant.