Group Equivariant CNNs Outperform Spatial Transformers on Tasks Which Require Rotation Invariance

1068707 [1]

Abstract

This study aims to show that group equivariant CNNs outperform spatial transformers, on tasks which demand rotation invariance, by providing theoretical background and experimental performance comparison with detailed analysis.

1. Introduction

Over the last decade, CNNs have demonstrated exceptional performance in various image processing tasks. However, they still lack the invariance to various geometric transformations of the input data, in computationally and parameter efficient way. An example of such an important and fundamental geometric transformation is a rotation. Consequently, in many tasks, we want our models to be rotation invariant. To accomplish rotation invariance, various approaches have been proposed. The least principled, but common approach is training ordinary CNNs with feature augmentation, which involves rotating images during training. However, this approach can be computationally expensive and there are no theoretical guarantees for its success. In this work, we focus on Spatial Transformers (Jaderberg et al., 2016) and SE(2)-Equivariant CNNs (Bekkers, 2020). Spatial transformers are based on a learnable spatial transformation module that can augment any convolutional layer with the ability to spatially transform feature maps, depending on the input feature map itself. The module is built from a small localization network which regresses the transformation parameters given the input feature map, while the differentiable sampler produces the transformed output feature map, given the regressed parameters. On the other hand, group equivariant CNNs are an implementation of a geometric deep learning blueprint (Bronstein et al., 2021), which provides a general approach to construction of deep learning models invariant to various transformations, arising from symmetries of the domain.

Consequently, both architectures have the inductive bias for rotation invariance, and to the author's knowledge, no work has focused on performance comparison of those two. I aim to answer the following research question: "Do group equivariant networks outperform spatial transformers, on task designed to require rotation invariance?"

I hypothesise that in tasks that demand rotation invariance, group equivariant neural networks should significantly outperform spatial transformer networks, due to their strong inductive bias built specifically for rotations. Apart from that, group equivariant neural networks incorporate parameter sharing, by learning group equivariant filters that are shared and applied across different regions of the input data. Intuitively, this should result in more efficient use of parameters and consequently lead to better generalization. On the other hand, Jaderberg et al. 2016 indicate that "the use of spatial transformers results in models which learn invariance to rotation, resulting in state-of-the-art performance on several benchmarks". However, reported advancements resulted from training on rotated images. Note that transformations applied by spatial transformer are conditioned on the input feature map. Intuitively, this implies that spatial transformer needs to see rotated images to learn which transformation to apply. Thus, I hypothesise that spatial transformer's ability to learn rotation invariance significantly depends on training with data augmentation and that its invariance to rotations depends on complexity of the object in the feature map. Due to the fact rotation is learnable via spatial attention module and there are no theoretical guarantees about power of such a module, I hypothesise that its invariance to rotations is only approximate, while group equivariant CNNs are significantly more robust to various rotations, on the same task.

2. Methodology

To provide theoretical background for this hypothesis, we will begin with an overview of spatial transformers and a detailed introduction to group equivariant CNNs. To study this hypothesis, we will present an adaptation of spatial transformer which applies only rotations. We will introduce group equivariant CNNs through the lens of geometric deep learning blueprint, following the formalization from Bekkers 2020. The main purpose of such an introduction to group equivariant CNNs is to provide theoretical reasons for their superiority on tasks which require rotation invariance. To test the hypothesis, we will experimentally evaluate the performance of models based on spatial transformers and group equivariant networks, on Rotated MNIST.

To ensure fair comparison, models will have similar number of parameters and experiments will be repeated three times, independently. See Implementation Details (6) for architectures and training configuration. To evaluate generalization power of considered models, models will be tested on unseen rotated images from the test set. We will consider training with and without rotated images. Quantitative and qualitative analysis will be based on model performance on test set. Quantitative analysis will be based on test set accuracy, while quantitative analysis will consist of pre-classifier representation visualization and feature map visualization.

3. (Rotation-only) Spatial Transformers

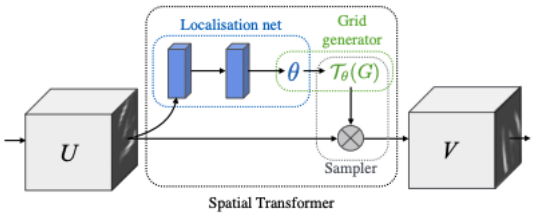

Figure 1. Spatial Transformer, taken from Jaderberg et al. 2016

The core architectural component of spatial transformer network is a spatial transformer layer, which consists of a localization network, a grid generator and a differentiable sampler. The localization network is a small neural network that takes the input feature map and regresses parameters of an affine transformation. Those parameters could be the rotation angle, translation vector, dilation factor or all of those. Regressed outputs of the localization network are fed to the grid generator, which constructs the sampling grid - a coordinate grid which specifies how the input feature map should be transformed. The sampler uses the constructed sampling grid to transform the input feature map. The output of the spatial transformer layer is a transformed feature map, which can be forwarded to the rest of the neural network for further processing. The spatial transformer layer can be used to augment any convolutional layer with ability to spatially transform its input feature map, using a learnable transformation conditioned on the input feature map itself.

As Equation 1 in Jaderberg et al. 2016 indicates, the default configuration of spatial transformer is to learn an affine transformation. To test the hypothesis, we need a way to bias spatial transformer to apply only rotations. This is achieved by constructing a localization network which regresses the value in range $(0, 1)$, which we multiply by 2π and treat as an angle. We parameterize generator's transformation T_θ by

$$T_\theta = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \end{bmatrix}.$$

To transform the input, we simply feed T_θ to the generator.

4. Group Equivariant CNNs through lens of geometric deep learning

Geometric deep learning blueprint suggests we design architectures based on linear equivariant layers, element-wise nonlinearities, local pooling layers and global invariant pooling layers. The main challenge is a design of linear equivariant layer. The following theorem will help us identify the structure of equivariant layers suggested by the blueprint.

Theorem (8.1)**.** *Let X, Y be homogeneous spaces on which Lie group G acts transitively, suppose μ_X is a Radon measure on X. Let $K : \mathcal{L}^2(X) \to \mathcal{L}^2(Y)$ be a bounded linear operator, which is G-equivariant. Then for some fixed $y_0 \in Y$, K is a group convolution with kernel k,*

$$[Kf](y) = \int_X f(x) k(g_y^{-1} \cdot x) \frac{\mathrm{d}\mu_X(g_y{}^{-1} \cdot x)}{\mathrm{d}\mu_X(x)} \, \mathrm{d}\mu_X(x),$$

where $g_y \in G$ is any $g \in G$ such that $y = g \cdot y_0$. Moreover, for every $h \in \mathrm{Stab}_G(y_0)$, for every $x \in X$,

$$k(x) = \frac{\mathrm{d}\mu_X(h^{-1} \cdot x)}{\mathrm{d}\mu_X(x)} k(h^{-1} \cdot x), \ \mu_X - a.e.$$

Proof of Theorem 8.1 with references is in the Appendix. Since we want roto-translation equivariant model, it is natural to choose $G = \mathrm{SE}(2) = \mathbb{R}^2 \rtimes \mathrm{SO}(2)$. Assume that images are signals in $\mathcal{L}^2(\mathbb{R}^2)$. Let $\rho(g)$ denote the matrix representation of $g \in \mathrm{SO}(2)$. Note that the origin is fixed by all rotations in $\mathrm{SO}(2)$, which implies $\mathbb{R}^2 \cong \mathrm{SE}(2)/\mathrm{SO}(2)$. To determine the initial $\mathrm{SE}(2)$-equivariant layer of our architecture, $K : \mathcal{L}^2(\mathbb{R}^2) \to \mathcal{L}^2(Y)$, consider $X = \mathbb{R}^2$. If $Y = \mathbb{R}^2$, by Theorem 8.1, for almost every $h \in \mathrm{SO}(2)$, $k(h^{-1} \cdot x) = k(x)$. In other words, k must be isotropic. Since we want maximally expressive models, we want no constraints on k, so $Y \cong \mathrm{SE}(2)/\{e\}$, which yields the

Lifting convolutional layer

$$[Kf](y, \theta) = \int_{\mathbb{R}^2} k(\rho(\theta^{-1})(\mathbf{x} - \mathbf{y})) f(\mathbf{x}) \, \mathrm{d}\mathbf{x}$$

After the lifting layer, resulting feature maps are signals on $\mathrm{SE}(2)$. Subsequent linear $\mathrm{SE}(2)$-equivariant layers can be formalized as $K : \mathcal{L}^2(\mathrm{SE}(2)) \to \mathcal{L}^2(\mathrm{SE}(2))$. Since $\mathrm{SE}(2)$ is a semi-direct product, we can split its action on itself into the spatial and group dimension. Theorem 8.1 indicates that such maximally expressive layers are as defined as follows.

Group convolutional layer

$$[Kf](y, \theta) = \int_{\mathbb{R}^2} \int_{S^1} k(\rho(\theta^{-1})(\mathbf{x} - \mathbf{y}), \psi - \theta) f(\mathbf{x}, \psi) \, \mathrm{d}\psi \, \mathrm{d}\mathbf{x}$$

Pooling layers We can construct local equivariant pooling layers by pooling only over the spatial dimension of feature map. We obtain the global invariant pooling layer by pooling over group dimension and all spatial dimensions. The principled construction provides strong theoretical foundation for superior performance on tasks depending on rotations.

5. Experiments

5.1. Results for training without rotated images

Table 1 shows performance results of spatial transformer networks. The value *single* in column *Mode* indicates that spatial transformer module is only in the first convolution block, while *multi* indicates that the module is in every convolution block. The results indicate that, when trained without rotated images, spatial transformer networks significantly struggle to classify rotated images. Interestingly, regardless of whether spatial transformer module is in single or in every convolution block, such networks are unable to learn invariance to rotation, without seeing rotated images. This can be seen directly in Figure 4, which shows that the spatial transformer trained without rotated images does not learn any rotation, since its first spatial transform is just the identity mapping, which was also the initialized mapping.

Mode	Conv Layers	Channels	Localization Channels	Kernel Size	Params	Test Accuracy
single	3	32	8	5	54.25 K	0.4341 ± 0.0099
single	3	64	8	5	209.07 K	0.442 ± 0.0118
single	5	32	16	5	110.54 K	0.4558 ± 0.0108
single	**5**	**64**	**8**	**5**	**414.0 K**	**0.464 ± 0.0061**
multi	3	32	17	5	101.88 K	0.4365 ± 0.0162
multi	3	64	35	5	412.31 K	0.4354 ± 0.0079
multi	5	32	18	5	202.52 K	0.4635 ± 0.0024
multi	5	64	37	5	821.58 K	0.462 ± 0.0121

Table 1. Performance of spatial transformers, identity initialization

Table 2 shows results for group equivariant models. The column *Discretization Order* indicates the order of discretization of SE(2). The results indicate that group equivariant models perform extremely well. Even the smallest group equivariant model (51.77K parameters) achieves test accuracy of 83.4%, while the best spatial transformer with 800K parameters achieves below 50% accuracy. Even without seeing rotated images, group equivariant networks can reach almost 95% accuracy, consistenly over three runs. Such a performance of group equivariant model is a consequence of its principled construction and a strong inductive bias for rotations. This illustrates the importance of inductive bias for learning rotations, while the accuracy of the smallest model demonstrates the efficiency of parameter sharing.

Discretization Order	Group Conv Layers	Channels	Kernel Size	Params	Test Accuracy
4	2	16	5	51.77 K	0.834 ± 0.0024
4	2	32	5	205.93 K	0.8401 ± 0.0128
4	4	16	5	102.97 K	0.9216 ± 0.0137
4	4	32	5	410.73 K	0.9294 ± 0.0083
8	2	16	5	102.97 K	0.8424 ± 0.0258
8	2	32	5	410.73 K	0.8652 ± 0.0168
8	4	16	5	205.37 K	0.9369 ± 0.0034
8	**4**	**32**	**5**	**820.33 K**	**0.9429 ± 0.007**

Table 2. Performance of various group equivariant models

A large difference in classification accuracy can be seen in embedding spaces of rotated test images, resulting from the global pooling layer preceding the classifier.

Figure 2 indicates that the best spatial transformer embeds most of the images very closely, resulting in embeddings that classifier cannot distinguish, which explains its bad classification performance. Figure 3 demonstrates that the group equivariant model projects images belonging to different labels far away from each other. We can see clusters of images belonging to the same label, and clusters are nicely separated, which is consistent with the excellent accuracy.

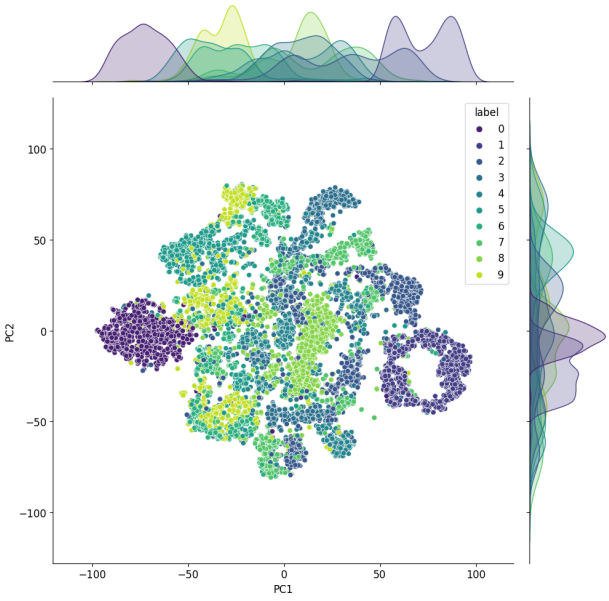

Figure 2. Test set embeddings of the best Spatial Transformer

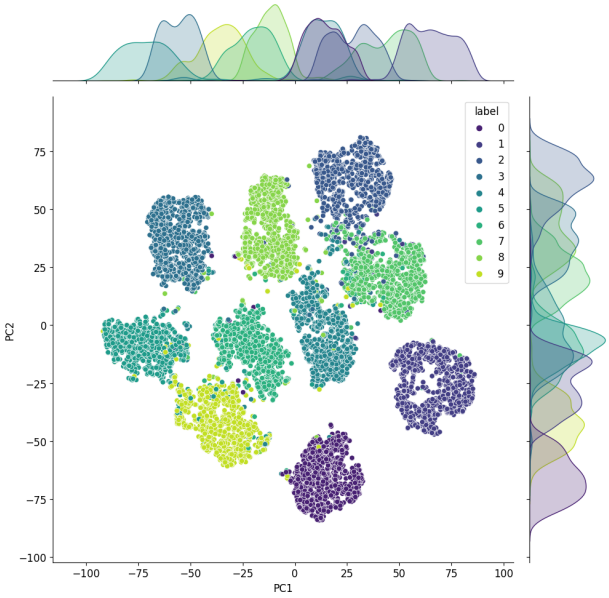

Figure 3. Test set embeddings of the order 4 group equivariant model, having 4 group convolutional layers, each with 32 channels

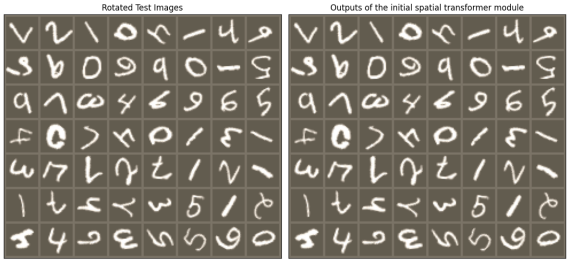

Figure 4. Transformations learned by the initial spatial transformer module in the best model

Structure of those embedding spaces can be explained by fixing the test image and examining the effects of various rotations of the input image on the learned representation of the image. In Figure 5, we can see that the group equivariant model yields exactly the same representations for the same image rotated by $0°, 90°, 180°$ and $270°$. This is expected, since the model is rotation equivariant and discretization order is 4. Due to such a discretization of SE(2), the model is not fully rotation equivariant. However, image representations for rotations by $45°, 135°, 225°$ and $315°$ are quite similar to the representation of the original image, indicating the model is robust to various rotations. Such representations are a direct consequence of equivariant feature maps, shown in Figure 5.1. Figure 5.1 illustrates the power of equivariance, which is the fact that rotation does not induce information loss, it only changes the feature map position along the group dimension in group convolutional layer. On the other hand, Figure 7 indicates that rotations of the fixed input image significantly affect the learned representation of the image in the best spatial transformer model. In particular, the model yields significantly different representations for the same image, rotated by $0°, 90°, 180°$ and $270°$. This is a consequence of two architectural issues, spatial transformer module's inability to learn appropriate rotation of the input and the fact that ordinary convolutional blocks are not rotation equivariant. This can bee seen in Figure 8, which shows that various rotations of the input image yield significantly different feature maps, which results in significantly different embeddings, illustrated in Figure 7.

Since the best group equivariant model achieves test accuracy of almost 95%, while the best Spatial Transformer achieves below 50% accuracy, quantitative results strongly support the hypothesis that group equivariant networks outperform spatial transformers, in the setting when models are trained on original images and tested on (possibly) rotated images. Qualitative embedding space analysis and feature map analysis support the hypothesis that geometric priors (SE(2) equivariance) and weight sharing are superior inductive bias for rotations. The consistenly bad test accuracy (Table 1) provides evidence that performance of spatial transformers depends on training with rotated images.

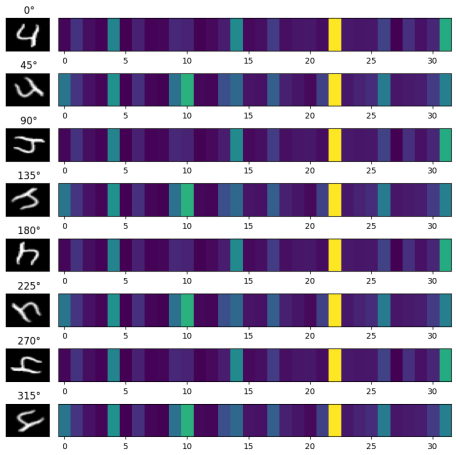

Figure 5. Embeddings generated by the group equivariant model

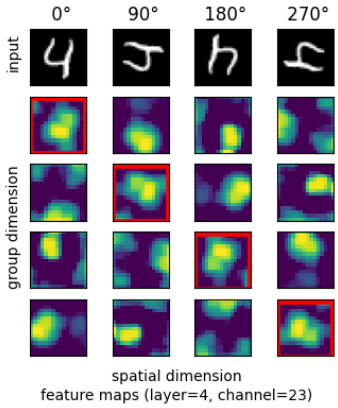

Figure 6. Feature maps from the group equivariant model

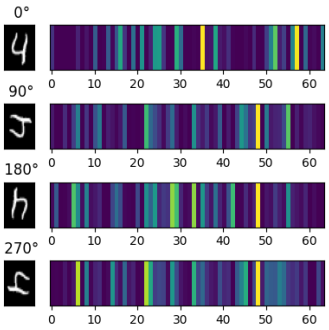

Figure 7. Embeddings generated by the best spatial transformer

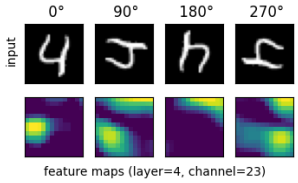

Figure 8. Feature maps from the best spatial transformer

5.2. Results for training with rotated images

According to Tables 3, 4, spatial transformers perform significantly better when trained with rotated images, which matches the observations and training setting in Jaderberg et al. 2016. All models achieve accuracy above 93%. The best spatial transformer has the single spatial transformer module in the first CNN block. It achieves test accuracy of 97.74%, while its best test accuracy is 97.83%. In comparison to Figure 2, test set embeddings by the best spatial transformer in Figure 9 seem to be significantly more robust to rotations, which explains much better performance.

Mode	Conv Layers	Channels	Localization Net Channels	Kernel Size	Params	Test Accuracy
single	3	32	8	5	54.25 K	0.9335 ± 0.0093
single	3	64	8	5	209.07 K	0.9379 ± 0.0021
single	5	32	16	5	110.54 K	0.9725 ± 0.0023
single	**5**	**64**	**8**	**5**	**414.0 K**	**0.9774 ± 0.0015**
multi	3	32	17	5	101.88 K	0.9354 ± 0.0086
multi	3	64	35	5	412.31 K	0.936 ± 0.0187
multi	5	32	18	5	202.52 K	0.9693 ± 0.0022
multi	5	64	37	5	821.58 K	0.9753 ± 0.0012

Table 3. Performance of spatial transformers, identity initialization

Mode	Conv Layers	Channels	Localization Net Channels	Kernel Size	Params	Test Accuracy
single	3	32	8	5	54.25 K	0.9407 ± 0.007
single	3	64	8	5	209.07 K	0.9429 ± 0.0039
single	5	32	16	5	110.54 K	0.9735 ± 0.0019
single	**5**	**64**	**8**	**5**	**414.0 K**	**0.9766 ± 0.0019**
multi	3	32	17	5	101.88 K	0.9471 ± 0.0113
multi	3	64	35	5	412.31 K	0.9446 ± 0.006
multi	5	32	18	5	202.52 K	0.9713 ± 0.0034
multi	5	64	37	5	821.58 K	0.9733 ± 0.0037

Table 4. Performance of spatial transformers, random initialization

According to Table 5, group equivariant models also perform better when trained with rotated images, The best such a model achieves average test accuracy of 98.55%, while its best test accuracy is 98.61%. These results are considerably better than the best spatial transformer. This supports the claim that group equivariant models outperform spatial transformers on tasks demanding rotation invariance.

Discretization Order	Group Conv Layers	Channels	Kernel Size	Params	Test Accuracy
4	2	16	5	51.77 K	0.9356 ± 0.0066
4	2	32	5	205.93 K	0.9507 ± 0.0038
4	4	16	5	102.97 K	0.9802 ± 0.0019
4	4	32	5	410.73 K	0.9801 ± 0.0078
8	2	16	5	102.97 K	0.9377 ± 0.0048
8	2	32	5	410.73 K	0.9405 ± 0.0027
8	4	16	5	205.37 K	0.9827 ± 0.001
8	**4**	**32**	**5**	**820.33 K**	**0.9855 ± 0.0007**

Table 5. Performance of various group equivariant models

Embeddings in Figures 9 and 12 support the claim that group equivariant models are more robust to various rotations of the input. This is a consequence of the fact that its feature maps are not rotation equivariant. Finnveden et al. also remark "that a rotation is not enough to align deeper layer feature maps." In Figures 10 and 11, we can see that spatial transformer's ability to learn appropriate rotation correction depends on the complexity of the object. We can see that the model knows how to handle zeros, while it seems confused by "more complex" eights. This supports the hypothesis that spatial transformer's invariance to rotations is only approximate, while Figure 12 demonstrates that group equivariant models are much more robust to various rotations, on the same task. Similar observations were made by Finnveden et al., who claim that "since STNs perform a purely spatial transformation, they do not, in the general case, have the ability to align the feature maps of a transformed image with those of its original. STNs are therefore unable to support invariance when transforming CNN feature maps."

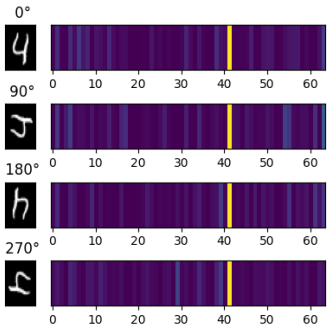

Figure 9. Embeddings generated by the best spatial transformer

Figure 10. Successful invariance transformation

Figure 11. Unsuccessful invariance transformation

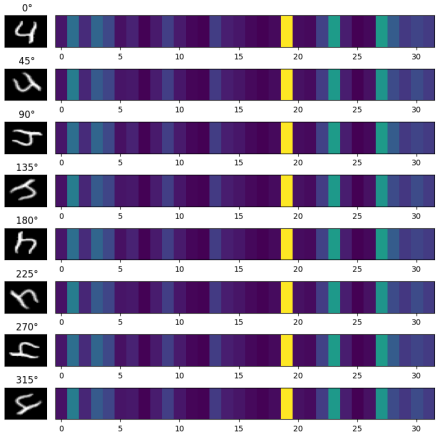

Figure 12. Embeddings generated by the group equivariant model

6. Implementation Details [Link to repository]

All spatial transformers models start with a `STConv` block, which is followed by a variable number of `(ST)Conv` blocks. Between every two `(ST)Conv` blocks, there is `MaxPool2D` with (2,2) kernel and stride 1. After the final `(ST)Conv` block, there is `GlobalAveragePooling` over spatial dimensions. Final layer is `Linear`, whose output size is the number of classes. Each `Conv` block consists of `Conv2D`, `LayerNorm` and `ReLU`. Each `STConv` block consists of a localization network, followed by a `Conv` block. `Conv` block in `STConv` receieves spatially transformed feature maps. Every localization network consists of `Conv` block, `MaxPool2D` with (2,2) kernel and stride 2, `Conv` block, `GlobalAveragePooling` and `Linear`, whose size depends on the number of transform parameters.

All group equivariant models start with `LiftingConv` layer, followed by a variable number of `GroupConv` blocks. Between every two `GroupConv` blocks, there is `GroupSpatialMaxPool`, with (2,2) kernel and stride 1. After the final `GroupConv` block, there is `GlobalAveragePooling` over group and spatial dimensions. Final layer is `Linear`, whose output size is the number of classes. `LiftingConv` block consists of lifting convolution, `LayerNorm` and `ReLU`. `GroupConv` block consists of group convolution, `LayerNorm` and `ReLU`.

All models are trained for at most 20 epochs using AdamW (Loshchilov & Hutter, 2017), with a learning rate $1e-2$ and weight decay $1e-4$. No other regularization is used. Test accuracy is evaluated on models selected by early stopping criterion based on validation accuracy, with patience of 5 epochs. Models were trained on NVIDIA A100 or V100.

7. Conclusion and Outlook

As in both training settings group equivariant networks outperformed spatial transformers of comparable size and a wide range of model configurations was considered, this study strongly supports the hypothesis that group equivariant models outperform spatial transformers, on tasks which require rotation invariance. Since qualitative embedding space and feature map analysis resulted in observations compatible with findings of Finnveden et al., there is evidence to support the hypothesis that group equivariant networks provide more robust invariance to rotations, while spatial transformers possess only approximate invariance. Although the study provides evidence in favour of the hypothesis, it was conducted on just one, relatively simple dataset. To reach more conclusive statements, a similar study over a wider range of datasets, tasks and model configurations is necessary. Since, due to limited computational resources, experiment results were gathered from just three independent runs, the study is not entirely statistically rigorous.

The study covered only regular group equivariant networks and it would be interesting to know whether similar conclusions can be drawn for steerable GCNs (Cohen & Welling, 2016) and harmonic networks (Worrall et al., 2017).

8. Appendix

8.1. Group Convolutions are all you need

Theorem 8.1. *Let X, Y be homogeneous spaces on which Lie group G acts transitively, suppose μ_X is a Radon measure on X. Let $K : \mathcal{L}^2(X) \to \mathcal{L}^2(Y)$ be a bounded linear operator, which is G-equivariant. Then for some fixed $y_0 \in Y$, K is a group convolution with kernel k,*

$$[Kf](y) = \int_X f(x) k(g_y^{-1} \cdot x) \frac{\mathrm{d}\mu_X(g_y^{-1} \cdot x)}{\mathrm{d}\mu_X(x)} \, \mathrm{d}\mu_X(x), \tag{1}$$

where $g_y \in G$ is any $g \in G$ such that $y = g \cdot y_0$. Moreover, for every $h \in \mathrm{Stab}_G(y_0)$, for every $x \in X$,

$$k(x) = \frac{\mathrm{d}\mu_X(h^{-1} \cdot x)}{\mathrm{d}\mu_X(x)} k(h^{-1} \cdot x), \ \mu_X - a.e. \tag{2}$$

Proof. By Theorem 1 (Duits, 2005) and the following discussion, K is an integral transform, with kernel $\widetilde{k}$. Since K is G-equivariant, for every $g \in G, f \in \mathcal{L}^2(X), y \in Y$,

$$([K \circ \rho_{G \to \mathcal{L}^2(X)}(g)](f))(y) = (\rho_{G \to \mathcal{L}^2(Y)}(g) \circ [Kf])(y)$$
$$\int_X \widetilde{k}(y, x) f(g^{-1} \cdot x) \, \mathrm{d}\mu_X(x) = \int_X \widetilde{k}(g^{-1} \cdot y, x) f(x) \, \mathrm{d}\mu_X(x).$$

Applying the change of variables with $x = g^{-1} \cdot z$ to the right hand side of equation above yields

$$\int_X \widetilde{k}(y, x) f(g^{-1} \cdot x) \, \mathrm{d}\mu_X(x) =$$
$$\int_X \widetilde{k}(g^{-1} \cdot y, g^{-1} \cdot x) f(g^{-1} \cdot x) \frac{\mathrm{d}\mu_X(g^{-1} \cdot x)}{\mathrm{d}\mu_X(x)} \, \mathrm{d}\mu_X(x),$$

which implies that

$$\int_X \left(\widetilde{k}(y, x) - \widetilde{k}(g^{-1} \cdot y, g^{-1} \cdot x) \frac{\mathrm{d}\mu_X(g^{-1} \cdot x)}{\mathrm{d}\mu_X} \right) f(g^{-1} \cdot x) \, \mathrm{d}\mu_X(x) = 0.$$

For g, y, set $f(x) = \widetilde{k}(y, g \cdot x) - \widetilde{k}(g^{-1} \cdot y, x) \frac{\mathrm{d}\mu_X(x)}{\mathrm{d}\mu_X(g \cdot x)}$. Applying the result above to f yields

$$\int_X \left| \widetilde{k}(y, x) - \widetilde{k}(g^{-1} \cdot y, g^{-1} \cdot x) \frac{\mathrm{d}\mu_X(g^{-1} \cdot x)}{\mathrm{d}\mu_X(x)} \right|^2 \mathrm{d}\mu_X(x) = 0.$$

Since the integrand is nonnegative, for every $g \in G$,

$$\widetilde{k}(y, x) = \widetilde{k}(g^{-1} \cdot y, g^{-1} \cdot x) \frac{\mathrm{d}\mu_X(g^{-1} \cdot x)}{\mathrm{d}\mu_X(x)}, \ \mu_X - \text{a.e.} \tag{3}$$

Since G acts transitively on Y, there exists $y_0 \in G$ such that for every $y \in Y$, there is $g_y \in G$ such that $g_y \cdot y_0 = y$. Applying 3 to x, y, g_y yields

$$\begin{aligned}\widetilde{k}(y,x) &= \widetilde{k}(g_y \cdot y_0, x) \\ &= \widetilde{k}(y_0, g_y^{-1} \cdot x)\frac{\mathrm{d}\mu_X(g_y^{-1}\cdot x)}{\mathrm{d}\mu_X(x)}.\end{aligned} \tag{4}$$

Since y_0 is fixed, we define single argument kernel k,

$$k(x) := \widetilde{k}(y_0, x). \tag{5}$$

Substituting Equation 4 into the form of K, yields

$$\begin{aligned}[Kf](y) &= \int_X \widetilde{k}(y,x) f(x)\,\mathrm{d}\mu_X(x) \\ &= \int_X f(x)\widetilde{k}(y_0, g_y^{-1}\cdot x)\frac{\mathrm{d}\mu_X(g_y^{-1}\cdot x)}{\mathrm{d}\mu_X(x)}\,\mathrm{d}\mu_X(x) \\ &= \int_X f(x) k(g_y^{-1}\cdot x)\frac{\mathrm{d}\mu_X(g_y^{-1}\cdot x)}{\mathrm{d}\mu_X(x)}\,\mathrm{d}\mu_X(x),\end{aligned}$$

by Equation 5. This proves Equation 1 in the claim. To prove the second claim, suppose $h \in \mathrm{Stab}_G(y_0)$. Since $h \in \mathrm{Stab}_G(y_0)$, $h \cdot y_0 = y_0$, so

$$\begin{aligned}k(x) &:= \widetilde{k}(y_0, x) && \text{by Equation 5} \\ &= \widetilde{k}(h\cdot y_0, x) && \text{since } h\cdot y_0 = y_0 \\ &= \widetilde{k}(y_0, h^{-1}\cdot x)\frac{\mathrm{d}\mu_X(h^{-1}\cdot x)}{\mathrm{d}\mu_X(x)} && \text{by Equation 4} \\ &= k(h^{-1}\cdot x)\frac{\mathrm{d}\mu_X(h^{-1}\cdot x)}{\mathrm{d}\mu_X(x)} && \text{by Equation 5,}\end{aligned}$$

where the equality above should be understood μ_X almost everywhere. This proves Equation 2. □

8.2. On Extension of Theorem 8.1

Theorem 8.1 is a well-known result in the field, whose variants were stated and proved in Bekkers 2020, Kondor & Trivedi 2018, Cohen et al. 2018. In my opinion, the simplest argument is from Bekkers 2020. However, in Bekkers 2020, the theorem is stated in full generality, but in the paper, it is proved only for $G = \mathbb{R}^n \rtimes H$. See Appendix A in the paper for more information. Apart from that, in the paper, Bekkers 2020 concludes that Equation 14 directly implies Equation 15. In my opinion, this is not immediately obvious, so the elaboration is provided. I also think that the kernel symmetry constraint (Equation 15 in the paper, Equation 2 in this paper) holds almost everywhere, not necessarily everywhere. The proof presented in this paper is for the claim in full generality, with further elaboration and "almost everywhere" note on the kernel symmetry constraint (Equation 2). It is based on the proof from Bekkers 2020.

References

Bekkers, E. B-spline cnns on lie groups, 03 2020. URL https://arxiv.org/pdf/1909.12057.pdf.

Bronstein, M. M., Bruna, J., Cohen, T., and Veličković, P. Geometric deep learning: Grids, groups, graphs, geodesics, and gauges. *arXiv:2104.13478 [cs, stat]*, 05 2021. URL https://arxiv.org/abs/2104.13478.

Cohen, T. and Welling, M. Steerable cnns, 12 2016. URL https://arxiv.org/pdf/1612.08498.pdf.

Cohen, T., Geiger, M., and Weiler, M. A general theory of equivariant cnns on homogeneous spaces, 11 2018. URL https://arxiv.org/abs/1811.02017.

Duits, R. *Perceptual organization in image analysis : a mathematical approach based on scale, orientation and curvature*. PhD thesis, Biomedical Engineering, 2005.

Finnveden, L., Jansson, Y., and Lindeberg, T. Understanding when spatial transformer networks do not support invariance, and what to do about it, 05 2021. URL https://arxiv.org/pdf/2004.11678.pdf.

Jaderberg, M., Simonyan, K., Zisserman, A., and Kavukcuoglu, K. Spatial transformer networks, 02 2016. URL https://arxiv.org/pdf/1506.02025.pdf.

Kondor, R. and Trivedi, S. On the generalization of equivariance and convolution in neural networks to the action of compact groups, 2018. URL https://arxiv.org/pdf/1802.03690.pdf.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. *arxiv.org*, 11 2017. URL https://arxiv.org/abs/1711.05101.

Worrall, D., Garbin, S., Turmukhambetov, D., and Brostow, G. Harmonic networks: Deep translation and rotation equivariance, 04 2017. URL https://arxiv.org/pdf/1612.04642.pdf.