Group Equivariant Neural Networks for Spectropolarimetric Inversions in Solar Astronomy

Anonymous authors
Paper under double-blind review

Abstract

The upcoming *Daniel K. Inouye Solar Telescope (DKIST)* will produce unprecedented high-cadence, high-resolution, and multi-line spectropolarimetric observations of the Sun. New computational techniques are needed to infer the state of the Sun's atmosphere from these observations. Deep learning is a promising approach to this spectropolarimetric inversion problem that can both provide *real-time* visualizations to astronomers and potentially improve upon existing algorithms by combining spatial, temporal, and multi-spectral information. Here we investigate group equivariant deep learning as a method for inferring the three-dimensional photospheric structures, training on magnetohydrodynamic (MHD) simulations of two types of solar features: sunspots and active regions. Our results demonstrate that including multiple lines improves the mean relative error from 18.6% to 14.4%, averaged over all MHD state variables, and that using group equivariant convolution architectures further improves the mean relative error to 12.5%.

1 Introduction

The solar photosphere hosts a variety of interacting physical phenomena including magnetic fields, convection, and energy transfer/conversion. These phenomena are generally well-described by the MHD equations for the evolution of the photospheric *state variables*: the magnetic field $\boldsymbol{B}$, velocity field v, density ρ, and pressure p throughout the 3d volume. Because these quantities determine how photons are emitted, absorbed, and re-emitted, solar physicists infer them from distinct spectral bands and polarization patterns (Stokes profiles) observed by telescopes. Thus, interpreting solar telescope data is an *inverse* problem (Figure 1).

Current spectropolarimetric inversion techniques are extremely computationally demanding. These inversion algorithms generally start with an initial guess of the solar state variables, and a physics model is used to calculate the resulting Stokes profiles that would be observed on Earth. These profiles are compared to observation data, and the error is used to update the estimate of the solar state variables —iterating until convergence. This inversion is essentially a non-convex optimization problem with convergence difficulties, ambiguities due to multiple local minima, and computationally expensive evaluations of the forward model. Thus, these models rely on a number of simplifications to speed up calculation (del Toro Iniesta & Ruiz Cobo, 2016), and processing cannot keep up with the *DKIST* data stream (20 TB per day on average) so astronomers cannot view solar conditions in real time. New computational methods are needed to meet the demands of modern solar astronomy.

As in many other inverse problems in physics (Sadowski & Baldi, 2018; Brehmer et al., 2020), deep learning is a promising approach because simulations can be used for training data, and approximate inference with neural networks is computationally efficient. MHD simulations are now advanced enough to faithfully reproduce many solar features from first principles (Rempel, 2012; Cheung et al., 2010; 2019) and provide the ground truth labels for training a neural network. The observations (inputs) are the observed Stokes parameters that would be observed on Earth, while the outputs are the hidden MHD variables that describe the state of the Sun's atmosphere.

The feasibility of deep learning for spectropolarimetric inversion was first demonstrated by the pioneering work of Ramos & Baso (2019) using a standard convolutional neural network. In this work,

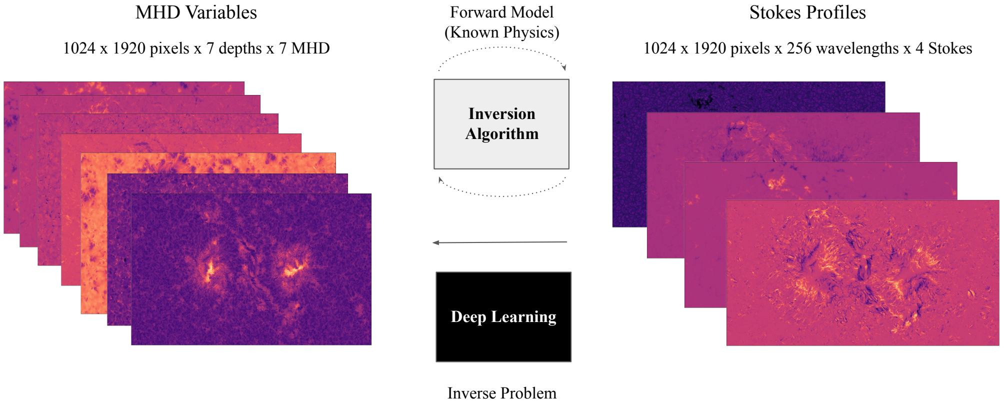

Figure 1: Current Stokes inversion techniques randomly initialize the hidden MHD state variables (left) and make iterative updates by repeatedly simulating the forward physics to predict observed Stokes profiles (right). This is done independently for each pixel, without taking advantage of spatial correlations. Instead, we use a deep G-CNN to predict all the MHD variables in a 3-d volume directly from the Stokes profiles in a single step. In our setup, the input to the G-CNN is a 4-d tensor of shape (16, 16, 272, 4) with two spatial dimensions of size 16, 272 wavelengths, and 4 stokes parameters. The output is a 4-d tensor with three spatial dimensions and 7 MHD variables.

we show that performance is improved by accounting for rotational symmetry with specialized neural network architectures. Enforcing this symmetry makes a model less likely to overfit (Gens & Domingos, 2014; Dieleman et al., 2016; Cohen & Welling, 2016; Cohen et al., 2019), with many applications in medical imaging (Winkels & Cohen, 2018; Bekkers et al., 2018), remote sensing, and astronomy (Dieleman et al., 2015). In particular, we use the implementation of *group equivariant* convolution networks (G-CNNs) proposed by Cohen & Welling (2016) to build models equivariant under 90° rotations and reflection transformations (the Dihedral group D_4). In a D_4 G-CNN, feature detectors at the initial convolutional layer are rotated and reflected, such that a rotation or reflection applied to the input image results in a permutation of the feature maps in the first hidden layer. Permutations of feature maps in the first hidden layer result in similar permutations in subsequent layers, until they are aggregated at the top, resulting in an image-to-image model that is globally-equivariant — or an image classifier that is invariant — to rotations and reflections.

In experiments, we compare G-CNNs against standard CNNs using two simulation datasets which capture solar features: a sunspot and an active region. Moreover, we show for the first time that performance improves if we include multiple spectral lines (more wavelengths) as inputs to the model. These spectral lines probe different heights in the photosphere, which provide additional information that can be exploited by our deep learning approach. Together, our experiments demonstrate that deep learning is a promising approach to the Stokes inversion problem, and that applications of deep learning to astronomy can benefit by designing architectures that capture symmetries of the data.

2 Experimental Results

2.1 Datasets and Evaluation

Two separate models were trained and evaluated on the simulations of an active region (Rempel, 2012) and a sunspot (Cheung et al., 2010). Stokes observations of both pairs of Fe I 630.2 and Fe I 1565 nm lines were synthesized using the software framework of Ramos & Baso (2019). These simulations both describe the state of the Sun's atmosphere as a sequence of 3d snapshots of seven variables. Each snapshot was split into non-overlapping 16 × 16 pixel patches along the latitude and longitude dimensions, resulting in 76,800 different 16×16×7×7 tensors for the active region and 9,216 tensors for the sunspot. For each tensor, we simulated the forward physics to obtain the Stokes profiles — the data observed by the telescope — using the SIR algorithm (Ruiz Cobo & del Toro Iniesta, 2012). This produced a labelled dataset for supervised learning, where the inputs are

4-d tensors of shape (16, 16, 4, 272) with two spatial dimensions each of size 16 and four Stokes parameters (I, Q, U, V) at 272 wavelengths (depending on which set of lines are used). The outputs are 4-d tensors of shape (16, 16, 7, 7) with three spatial dimensions and 7 MHD variables

A problem that arises when predicting magnetic fields is that there are ambiguous solutions: the same Stokes parameters can correspond to opposing magnetic fields, so it may be impossible for the model to predict the true direction. We remove this ambiguity using the coordinate transformation described in Ramos & Baso (2019) that maps the original field (B_x, B_y, B_z) to field (B_Q, B_U, B_V):

$$B_Q = \text{sign}(B_x^2 - B_y^2)|B_x^2 - B_y^2|^{\frac{1}{2}} \tag{1}$$

$$B_U = \text{sign}(B_x B_y)|B_x B_y|^{\frac{1}{2}} \tag{2}$$

$$B_V = B_z. \tag{3}$$

The input data was then standardized by subtracting the mean and dividing by the standard deviation across heights, widths, time steps, and wavelengths. Outputs were standardized by subtracting the mean and dividing by the standard deviation across heights, widths, and time steps.

Each dataset was split into 60/20/20% train/validation/test. For the sunspot, splits were made chronologically, keeping all patches within the same snapshot in the same split. Models were optimized with Adam using a batch size of 32 (Kingma & Ba, 2014). The initial learning rate was $1e^{-4}$ and was halved after 10 epochs if no improvement was observed in the validation loss. After 15 epochs of no improvement on the validation loss, training was stopped. Hyperparameter optimization was conducted with SHERPA to minimize the validation set loss (Hertel et al., 2020). Data augmentation of rotations and mirror reflections were used during evaluation on the test set, and we report the median relative error for each variable across all heights, widths, and depths. These choices were held constant for each model.

2.2 Do Multi-line Observations Improve Performance?

DKIST's multi-line observations should enable us to cover a wider range of physical heights and better constrain the vertical gradient of MHD variables. The latter will improve our understanding of several important questions, for example, the horizontal electric current density and the degree of force-freeness of the photosphere (Puschmann et al., 2010; Metcalf et al., 1995). Initial line candidates include the well studied Fe I 630.2 and Fe I 1565 nm, which will be simultaneously observed with DL-NIRSP, with possible extension to the He I/Si I lines near 1083 nm.

We confirm that observing multiple lines improves performance in the deep learning approach. A fully convolutional neural network (CNN) was trained with two different sets of inputs: (1) only the Fe I 1565 nm lines (160 wavelengths); (2) both the Fe I 630.2 and Fe I 1565 nm lines (272 wavelengths). The results show clear performance improvements using both sets of lines (Table 1).

		Active Region		Sunspot Region	
Property		Single line	Multi-line	Single line	Multi-line
Height	km	0.19%	**0.17%**	1.28%	**1.23%**
Temperature	K	1.27%	**0.87%**	1.46%	**1.16%**
Pressure	Pa	4.43%	**3.77%**	6.78%	**6.36%**
Velocity Z	m/s	27.38%	**20.93%**	49.18%	**41.66%**
Magnetic field Q	$gauss$	31.56%	**24.27%**	32.95%	**26.23%**
Magnetic field U	$gauss$	32.53%	**24.66%**	33.07%	**25.00%**
Magnetic field V	$gauss$	33.08%	**26.03%**	30.19%	**25.31%**
Average	NA	18.63 %	**14.39%**	22.13%	**18.14%**

Table 1: Relative error on the test set of two CNN inversion models trained using single (Fe I 1565nm) and multiple lines (Fe I 630.2nm and Fe I 1565nm). The multi-line model outperforms the single-line model for each property.

2.3 Do Group-Equivariant CNNs Improve Performance?

Enormous progress has been made in computer vision by leveraging two key properties in CNN architectures: *equivariance* and *invariance* (Goodfellow et al., 2016). Convolutional layers provide translational equivariance: when the image is translated, the activations of the convolutional layer are translated too. When stacked with pooling layers, the CNNs can achieve translational invariance: when the image is translated, the output does not change. Invariance to translations constrains the learning problem and enables CNNs to learn from fewer examples. However, many applications in medical imaging, remote sensing, and astronomy also have *rotational* equivariance or invariance, which is harder to capture in neural network architectures. The G-CNN architecture introduced by Cohen & Welling (2016) provides a computationally-efficient architecture that is provably equivariant to 90° rotations and reflection transformations (the Dihedral group, D_4).

In G-CNNs, feature detectors at the initial convolutional layer are rotated and reflected. Thus, a rotation or reflection applied to the input image simply results in a reordering of the output feature maps at the first layer. At intermediate layers, the order of feature detectors are permuted such that permutations of feature maps in one layer result in permutations of feature maps in subsequent layers. Lastly, pooling at the output layer results in locally invariant and globally equivariant representations.

We construct a D_4-CNN model, a G-CNN on the D_4 group, by stacking group convolution (G-convolution) layers in D_4 along with group pooling layers (G-pooling) after each G-convolution layer. Next, we evaluate performance on a standard CNN as a baseline. We also evaluate performance on a CNN where the input-output pairs are augmented by random 90° rotations. The results show that rotation equivariance is a useful inductive bias that cannot be obtained from only data augmentation. The D_4-CNN overfits less to the training set, and the results on the test set show clear performance improvements across each MHD variable (Table 2). The inversion examples in Figure 2 also show that the D_4-CNN can accurately infer the features of the MHD variables.

		Active Region			Sunspot Region		
Property		CNN	CNN +R	D_4-CNN	CNN	CNN +R	D_4-CNN
Height	km	0.17%	0.17%	**0.14%**	1.23%	1.25%	**1.14%**
Temperature	K	0.87%	0.87%	**0.70%**	1.16%	1.19%	**1.00%**
Pressure	Pa	3.77%	3.77%	**3.20%**	6.36%	6.50%	**6.15%**
Velocity Z	m/s	20.93%	20.96%	**17.97%**	41.66%	42.74%	**38.31%**
Mag. field Q	$gauss$	24.27%	24.30%	**21.35%**	26.23%	26.95%	**23.61%**
Mag. field U	$gauss$	24.66%	24.65%	**22.03%**	25.00%	25.47%	**22.06%**
Mag. field V	$gauss$	26.03%	26.20%	**22.16%**	25.31%	26.21%	**22.74%**
Average	NA	14.39%	14.42 %	**12.51%**	18.14%	18.62%	**16.43%**

Table 2: Relative error on the test set for a CNN, a CNN with random rotations of $90°, 180°$, or $270°$ (CNN+R), and a D_4-CNN. Each model was trained on the multi-line set of inputs, with an equivalent number of parameters. The D_4-CNN outperforms both standard CNNs for every property

3 Conclusion

DKIST will be the center of ground-based solar observation for several decades. The *DKIST* Level-2 effort, being implemented by the National Solar Observatory (NSO) for the community, will initially concentrate on providing routine inversion data products for a fraction of *DKIST* observations focused on single snapshots in time, rather than at the high time cadence afforded by photospheric use cases. This deep learning approach could accelerate scientific progress by providing faster, cheaper, larger and more accurate data products to the scientific community.

Here we have introduced a deep learning model equivariant to rotations and reflections for the spectropolarimetric inversion problem, and showed through experiments that this is a useful inductive bias, improving performance by 13%. Moreover, we demonstrate that deep learning is able to successfully integrate information from multiple spectral lines to improve performance, whereas iterative approaches are only able to invert lines independently. Once trained, D_4-CNN inversions are

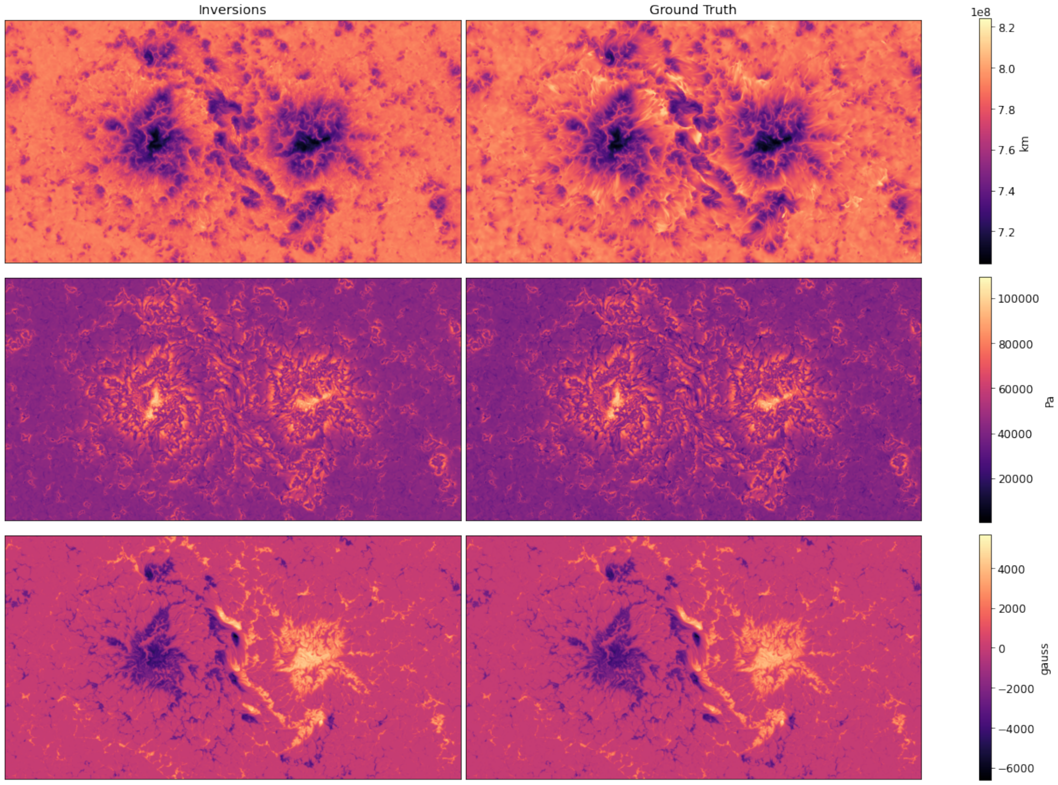

Figure 2: Stokes inversions for the active region using a D_4-CNN (left) vs. ground truth (right) on the test set. The model simultaneously predicts seven different properties at seven depths. Shown here from top to bottom are surface height, pressure, and magnetic field V.

orders of magnitude faster ($\approx 7.4 \times 10^4$pixel/sec) in comparison to current inversion methods such as VFSIV ($\approx 6 \times 10^2$pixel/sec) (Borrero et al., 2011).

Our collaborators are currently running large-scale MHD simulations at the NCAR supercomputing facility to provide training data for this project. Future work will leverage the insights from this work for much larger models.

Acknowledgments

We thank Curt Dodds, Sarah Jaeggli, Matthias Rempel, Lucas Tarr, and Tom Schad for valuable discussions. This material is based upon work supported by the National Science Foundation under Grants No. 1848250 and 2008344. The work was made possible by hardware grants from NVIDIA to PS, along with technical support and advanced computing resources from the University of Hawai'i Information Technology Services Cyberinfrastructure group.

References

Erik J Bekkers, Maxime W Lafarge, Mitko Veta, Koen AJ Eppenhof, Josien PW Pluim, and Remco Duits. Roto-translation covariant convolutional networks for medical image analysis. In *International conference on medical image computing and computer-assisted intervention*, pp. 440–448. Springer, 2018.

JM Borrero, S Tomczyk, M Kubo, H Socas-Navarro, J Schou, S Couvidat, and R Bogart. Vfisv: Very fast inversion of the stokes vector for the helioseismic and magnetic imager. *Solar Physics*, 273(1):267–293, 2011.

Johann Brehmer, Gilles Louppe, Juan Pavez, and Kyle Cranmer. Mining gold from implicit models to improve likelihood-free inference. *Proceedings of the National Academy of Sciences*, 117(10): 5242–5249, 2020.

M. C. M. Cheung, M. Rempel, A. M. Title, and M. Schüssler. Simulation of the Formation of a Solar Active Region. *Astrophys. J.*, 720(1):233–244, Sep 2010. doi: 10.1088/0004-637X/720/1/233.

M. C. M. Cheung, M. Rempel, G. Chintzoglou, F. Chen, P. Testa, J. Martínez-Sykora, A. Sainz Dalda, M. L. DeRosa, A. Malanushenko, V. Hansteen, B. De Pontieu, M. Carlsson, B. Gudiksen, and S. W. McIntosh. A comprehensive three-dimensional radiative magnetohydrodynamic simulation of a solar flare. *Nat. Astron.*, 3:160–166, Jan 2019. doi: 10.1038/s41550-018-0629-3.

MCM Cheung, M Rempel, M Schüssler, et al. Simulation of the formation of a solar active region. *The Astrophysical Journal*, 720(1):233, 2010.

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

Taco Cohen, Maurice Weiler, Berkay Kicanaoglu, and Max Welling. Gauge equivariant convolutional networks and the icosahedral cnn. In *International Conference on Machine Learning*, pp. 1321–1330. PMLR, 2019.

Jose Carlos del Toro Iniesta and Basilio Ruiz Cobo. Inversion of the radiative transfer equation for polarized light. *Living Reviews in Solar Physics*, 13(1):4, Nov 2016. ISSN 1614-4961. doi: 10.1007/s41116-016-0005-2. URL https://doi.org/10.1007/s41116-016-0005-2.

Sander Dieleman, Kyle W Willett, and Joni Dambre. Rotation-invariant convolutional neural networks for galaxy morphology prediction. *Monthly notices of the royal astronomical society*, 450 (2):1441–1459, 2015.

Sander Dieleman, Jeffrey De Fauw, and Koray Kavukcuoglu. Exploiting cyclic symmetry in convolutional neural networks. In *International conference on machine learning*, pp. 1889–1898. PMLR, 2016.

Robert Gens and Pedro M Domingos. Deep symmetry networks. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger (eds.), *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014. URL https://proceedings.neurips.cc/paper/2014/file/f9be311e65d81a9ad8150a60844bb94c-Paper.pdf.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

Lars Hertel, Julian Collado, Peter Sadowski, Jordan Ott, and Pierre Baldi. Sherpa: Robust hyperparameter optimization for machine learning. *SoftwareX*, 12:100591, 2020. ISSN 2352-7110. doi: https://doi.org/10.1016/j.softx.2020.100591. URL https://www.sciencedirect.com/science/article/pii/S2352711020303046. arXiv:2005.04048 [cs.LG].

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

Thomas R Metcalf, Litao Jiao, Alexander N McClymont, Richard C Canfield, and Han Uitenbroek. Is the solar chromospheric magnetic field force-free? *The Astrophysical Journal*, 439:474–481, 1995.

KG Puschmann, B Ruiz Cobo, and V Martínez Pillet. The electrical current density vector in the inner penumbra of a sunspot. *The Astrophysical Journal Letters*, 721(1):L58, 2010.

A Asensio Ramos and CJ Díaz Baso. Stokes inversion based on convolutional neural networks. *Astronomy & Astrophysics*, 626:A102, 2019.

Matthias Rempel. Numerical sunspot models: robustness of photospheric velocity and magnetic field structure. *The Astrophysical Journal*, 750(1):62, 2012.

B Ruiz Cobo and JC del Toro Iniesta. Sir: Stokes inversion based on response functions. *Astrophysics Source Code Library*, pp. ascl–1212, 2012.

Peter Sadowski and Pierre Baldi. Deep learning in the natural sciences: applications to physics. In *Braverman Readings in Machine Learning. Key Ideas from Inception to Current State*, pp. 269–297. Springer, 2018.

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

A Appendix

A.1 Architecture Ablation Study

We perform an ablation study on the D_4-CNN trained on the sunspot region to measure the relative contributions of the different components of the model (Table 3). We focus on the contributions from G-convolution on different groups (D_4 or C_4) and G-pooling layers. A G-convolution layer simply results in kernels and feature maps coming in groups of 4 or 8 corresponding to either the 4 rotations in C_4, the Cyclic group, or 8 rotations and reflections in D_4 (Figure 3). A G-pooling layer pools over all rotated and reflected versions of feature maps from the same kernel.

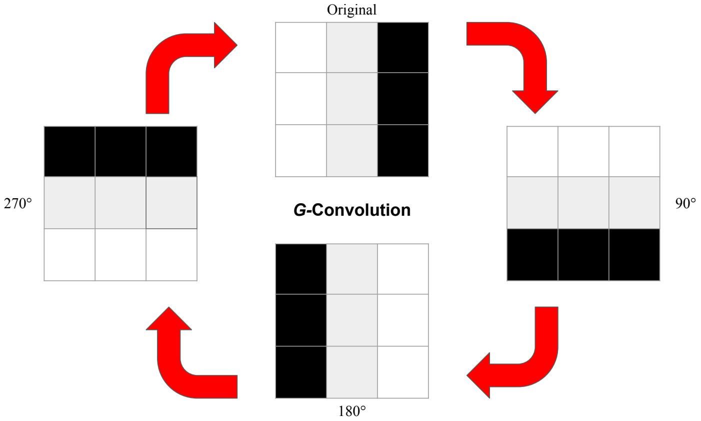

Figure 3: An example 3 × 3 kernel at the initial layer and its rotation in G-convolution on the C_4 group.

First, we replace all G-convolution layers equivariant in D_4 with G-convolution layers equivariant in C_4. This results in the model being equivariant to rotations, but *not* reflections. Next, we remove all G-pooling layers except for one at the last layer — this results in equivariant but non-invariant intermediate layers, despite the model as a whole retaining equivariance. We test this for the full D_4 model as well. Lastly, we remove all G-pooling layers completely, resulting in the model *not* retaining equivariance.

		Architectures				
Property		D_4-CNN	D_4-CNN $-$P	C_4-CNN	C_4-CNN $-$P	C_4-CNN $-$PP
Height	km	1.14%	1.14%	1.12%	1.13%	1.13%
Temperature	K	1.00%	1.00%	1.00%	0.99%	0.95%
Pressure	Pa	6.15%	6.00%	6.06%	5.89%	5.90%
Velocity Z	m/s	38.31%	38.60%	37.40%	37.11%	37.05%
Mag. field Q	$gauss$	23.61%	25.78%	23.30%	24.54%	25.89%
Mag. field U	$gauss$	22.06%	24.28%	21.56%	23.21%	24.40%
Mag. field V	$gauss$	22.74%	24.51%	22.43%	23.19%	23.95%
Average	NA	16.43%	17.33%	16.12%	16.54%	17.04%

Table 3: Relative error on the sunspot region testing set comparing a full D_4-CNN with group pooling at each layer, a D_4-CNN with group pooling only at the output layer indicated by $-$P, a C_4-CNN with group pooling at each layer, a C_4-CNN with group pooling only at the output layer, and a C_4-CNN without group pooling indicated by $-$PP.

When removing G-pooling at intermediate layers we need to account for the difference in trainable parameters between models. Compared to G-convolution in D_4 using G-pooling, G-convolution layers without G-pooling result in feature maps having 8 times the number of input channels. Thus, keeping the number of unique filters the same between layers results in G-convolution layers without G-pooling having 8 times the number of trainable parameters. To make a fair comparison, we need to divide the number of filters in each of the G-convolution layers without G-pooling by the square root of the group size as suggested in Cohen & Welling (2016). The result is that there are $1/\sqrt{8}$ times the number of filters in G-convolution layers without pooling and therefore $1/\sqrt{8}$ times the number of output feature maps. Composing layers in this fashion results in $1/8$ times the number of trainable parameters in each G-convolution layer without pooling. Thus, when G-pooling is not used the number of trainable parameters between models remains constant, allowing for fair comparisons.

The results indicate that the largest performance improvement over standard CNNs come from using rotation-equivariant features in the C_4-CNN. The use of final pooling layers in C_4-CNN $-$P and the use of intermediate pooling layers in D_4-CNN and C_4-CNN improve performance to a lesser extent in comparison to their non-pooling counterparts. We expected the premature invariance achieved by intermediate G-pooling layers to reduce performance as found in Cohen & Welling (2016). However, the observed improvement may be due to the reduction in the number of filters in models not using G-pooling at intermediate layers. Surprisingly, enforcing equivariance in D_4 rather than C_4 slightly decreases performance; we suspect this is not significant and the result of using a small training set.