HEAL-SWIN and equivariant non-linear maps

2025-08-29, PhD defence

Oscar Carlsson, Department of Mathematical Sciences

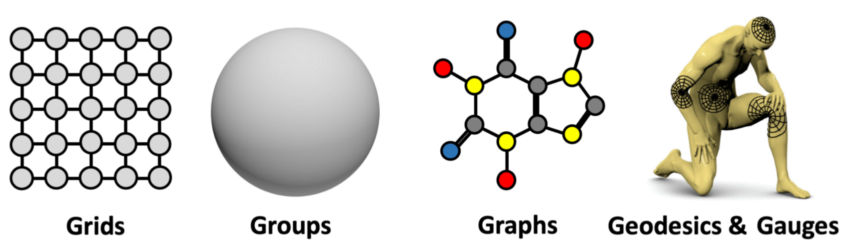
WASP | WALLENBERG AI, AUTONOMOUS SYSTEMS AND SOFTWARE PROGRAM

CHALMERS UNIVERSITY OF TECHNOLOGY | UNIVERSITY OF GOTHENBURG

My work

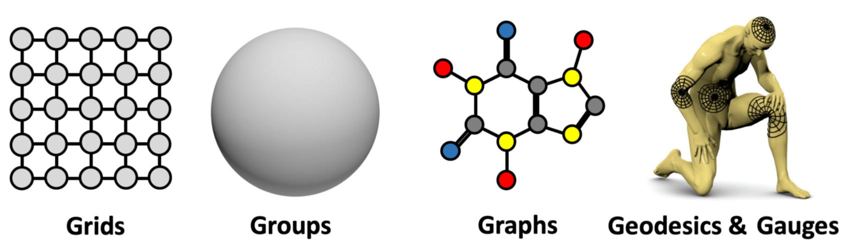
- **Paper I.**: Jan E. Gerken, Jimmy Aronsson*, **Oscar Carlsson***, Hampus Linander, Fredrik Ohlsson, Christoffer Petersson, and Daniel Persson "Geometric deep learning and equivariant neural networks". In: *Artificial Intelligence Review* (June 2023)
- **Paper II.**: Jan Gerken, **Oscar Carlsson**, Hampus Linander, Fredrik Ohlsson, Christoffer Petersson, and Daniel Persson "Equivariance versus Augmentation for Spherical Images". In: *Proceedings of the 39th International Conference on Machine Learning* (June 2022), pp. 7404-7421
- **Paper III.**: **Oscar Carlsson***, Jan E. Gerken*, Hampus Linander, Heiner Spieß, Fredrik Ohlsson, Christoffer Petersson, and Daniel Persson Daniel "HEAL-SWIN: A Vision Transformer on the Sphere". In: *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (June 2024), pp. 6067-6077
- **Paper IV.**: Elias Nyholm*, **Oscar Carlsson***, Maurice Weiler, and Daniel Persson "Equivariant non-linear maps for neural networks on homogeneous spaces". *Submitted* (April 2025)

Purpose of GDL



(“Geometric Deep Learning: Grids, Groups, Graphs, Geodesics, and Gauges”, Bronstein et al. 2021)

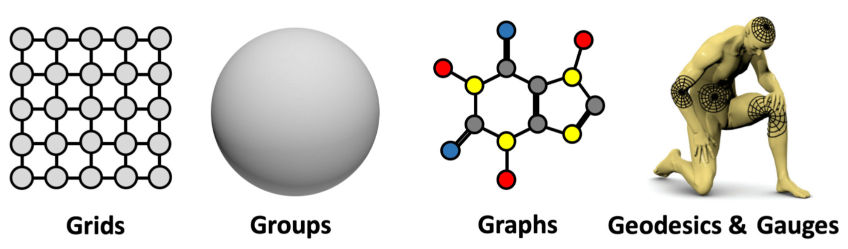
Purpose of GDL



("Geometric Deep Learning: Grids, Groups, Graphs, Geodesics, and Gauges", Bronstein et al. 2021)

Paper III: images on a sphere $S^2 = \mathrm{SO}(3)/\mathrm{SO}(2)$

Purpose of GDL



("Geometric Deep Learning: Grids, Groups, Graphs, Geodesics, and Gauges", Bronstein et al. 2021)

Paper III: images on a sphere $S^2 = \mathrm{SO}(3)/\mathrm{SO}(2)$

Paper IV: data on general homogenous space $X \cong G/H$

Paper III:
HEAL-SWIN

Material

HEAL-SWIN: A Vision Transformer On The Sphere

Oscar Carlsson[*ab] Jan E. Gerken[*a] Hampus Linander[a] Heiner Spieß[c] Fredrik Ohlsson[d]
Christoffer Petersson[ea] Daniel Persson[a]

Abstract

High-resolution wide-angle fisheye images are becoming more and more important for robotics applications such as autonomous driving. However, using ordinary convolutional neural networks or vision transformers on this data is problematic due to projection and distortion losses introduced when projecting to a rectangular grid on the plane. We introduce the HEAL-SWIN transformer, which combines the highly uniform Hierarchical Equal Area iso-Latitude Pixelation (HEALPix) grid used in astrophysics and cosmology with the Hierarchical Shifted-Window (SWIN) transformer to yield an efficient and flexible model capable of training on high-resolution, distortion-free spherical data. In HEAL-SWIN, the nested structure of the HEALPix grid is used to perform the patching and windowing operations of the SWIN transformer, enabling the network to process spherical representations with minimal computational overhead. We demonstrate the superior performance of our model on both synthetic and real automotive datasets, as well as a selection of other image datasets, for semantic segmentation, depth regression and classification tasks. Our code is publicly available[1].

1. Introduction

High-resolution fisheye cameras are among the most common and important sensors in modern intelligent vehicles [42]. Due to their non-rectilinear mapping functions and large field of view, fisheye images are highly distorted. Moreover, the most commonly used large-scale computer vision and autonomous driving datasets do not contain fisheye images. For these reasons, fisheye images have received much less attention than rectilinear images in the literature.

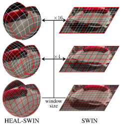


Figure 1. Our HEAL-SWIN model uses the nested structure of the HEALPix grid to lift the windowed self-attention of the SWIN model onto the sphere.

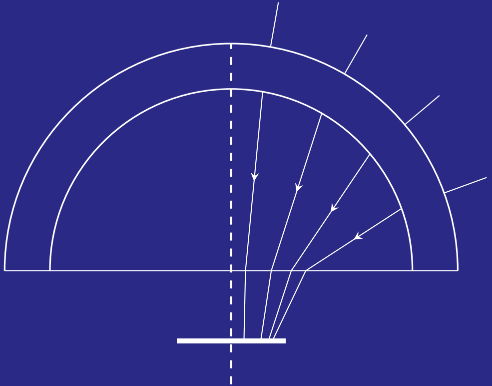
Despite the distortions introduced by the mapping function, the traditional approach for dealing with this kind of data is to use standard (flat) convolutional neural networks which are adjusted to the distortions and either preprocess the data [16, 17, 26, 36, 46, 50] or deform the convolution kernels [47]. However, these approaches struggle to capture the inherent spherical geometry of the images since they operate on a flat approximation of the sphere. Errors and artifacts arising from handling the strong and spatially inhomogeneous distortions are particularly problematic in safety-critical applications such as autonomous driving.

Utilizing spherical representations is an approach taken by some models [8, 10, 19] which lift convolutions to the sphere. These models rely on a rectangular grid in spherical

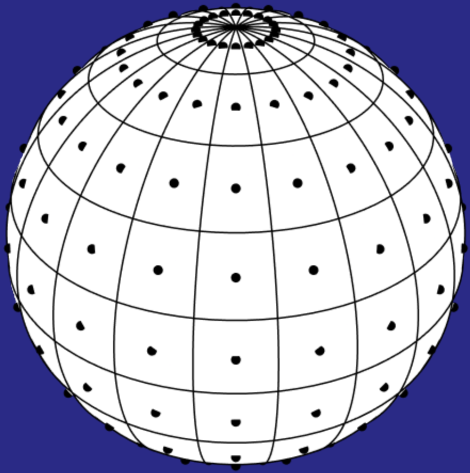
[*] Equal contribution
[a] Department of Mathematical Sciences, Chalmers University of Technology, University of Gothenburg, SE-412 96 Gothenburg, Sweden
[b] Corresponding author, email: osccarls@chalmers.se
[c] Neural Information Processing, Science of Intelligence, Technical University Berlin, DE-10623 Berlin, Germany
[d] Department of Mathematics and Mathematical Statistics, Umeå University, SE-901 87 Umeå, Sweden
[e] Zenseact, SE-417 56 Gothenburg, Sweden
[1] https://github.com/JanEGerken/HEAL-SWIN

6067

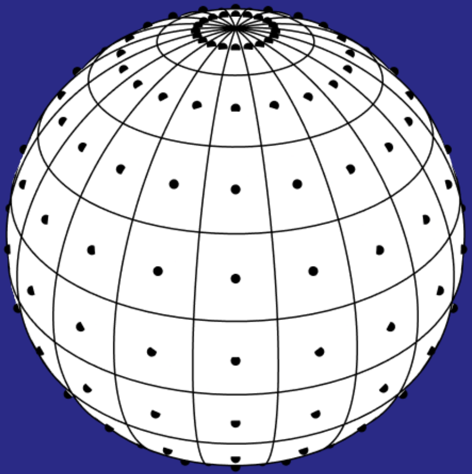
Motivation: Large FOV images are curved

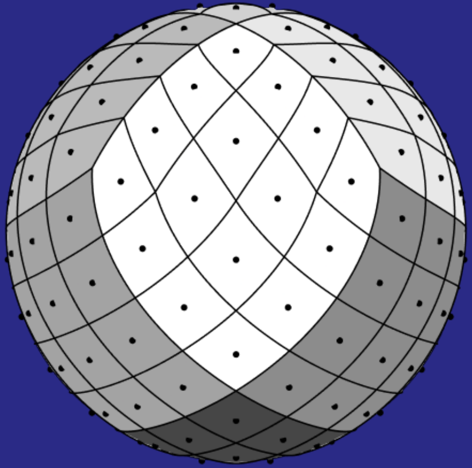


Sampling on the sphere: Driscoll-Healy



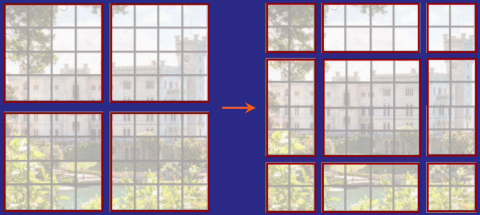
Sampling on the sphere: Driscoll-Healy vs HEALPix



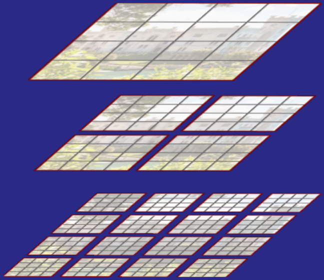


SWIN

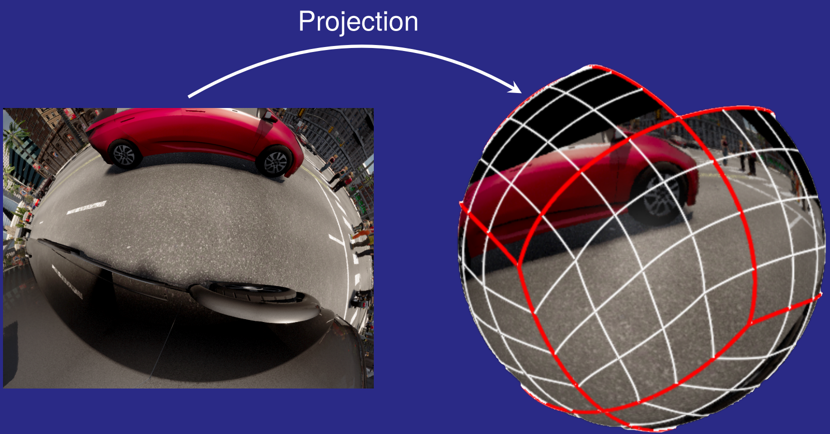
Shifting attention windows



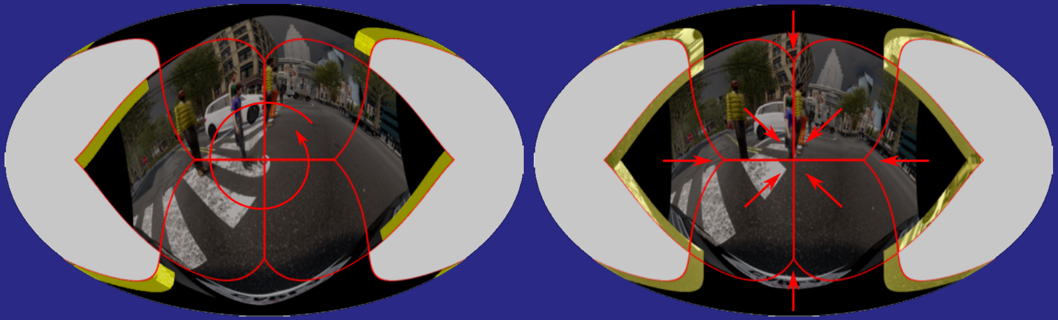
Patch merging



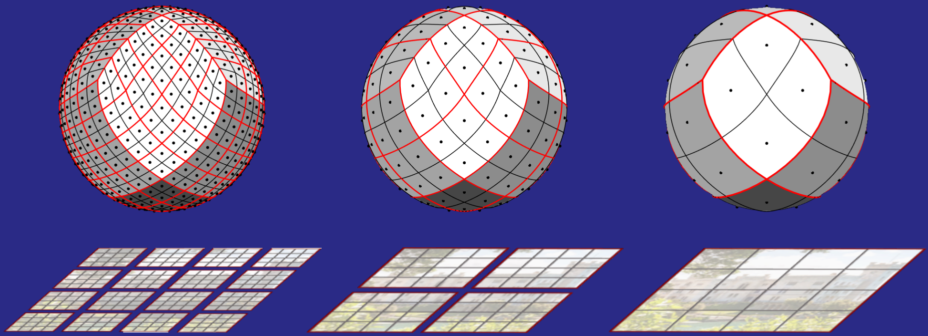
HEAL-SWIN



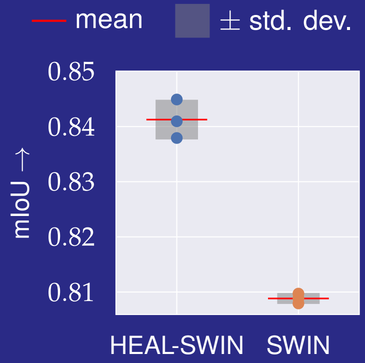
Shifting

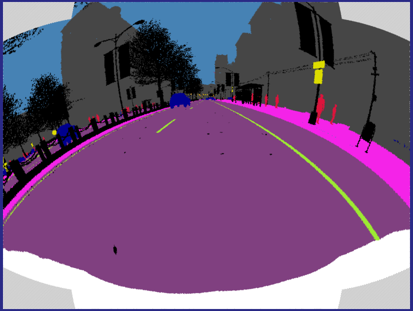


Patch merging

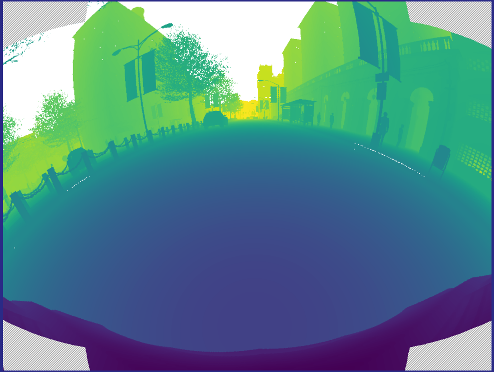


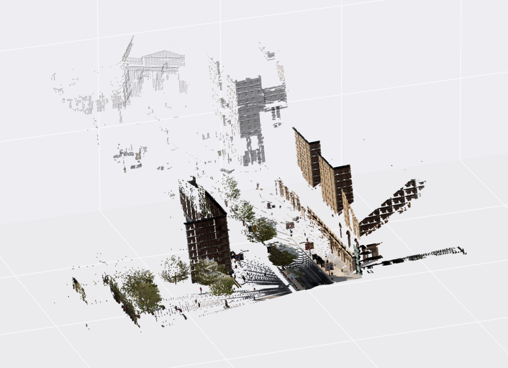
Results: Semantic segmentation on SynWoodScape



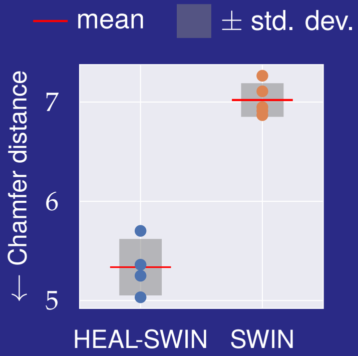


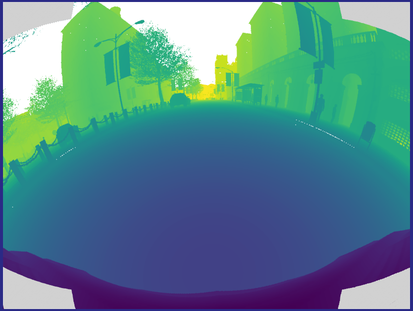
Experiment: Depth estimation





Results: Depth estimation on SynWoodScape





Outlook

- Use HEAL-SWIN, or inspired methods, on large FOV in vehicles/drones

Outlook

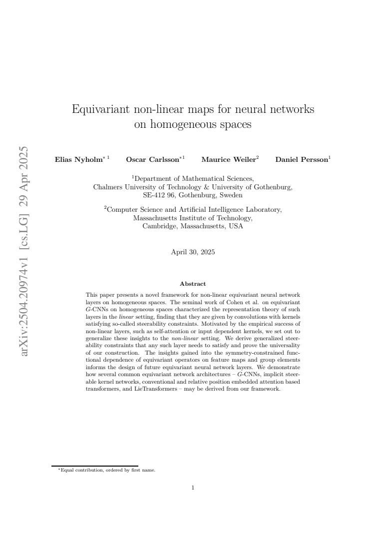
- Use HEAL-SWIN, or inspired methods, on large FOV in vehicles/drones
- Other spherical data (e.g. CMB, weather data (PEAR))

Outlook

- Use HEAL-SWIN, or inspired methods, on large FOV in vehicles/drones
- Other spherical data (e.g. CMB, weather data (PEAR))
- Try to extend this to a rotationally equivariant structure

Paper IV:
Framework for equivariant non-linear maps

Material: Paper IV



arXiv:2504.20974v1 [cs.LG] 29 Apr 2025

Equivariant non-linear maps for neural networks on homogeneous spaces

Elias Nyholm[*1] **Oscar Carlsson**[*1] **Maurice Weiler**[2] **Daniel Persson**[1]

[1]Department of Mathematical Sciences,
Chalmers University of Technology & University of Gothenburg,
SE-412 96, Gothenburg, Sweden

[2]Computer Science and Artificial Intelligence Laboratory,
Massachusetts Institute of Technology,
Cambridge, Massachusetts, USA

April 30, 2025

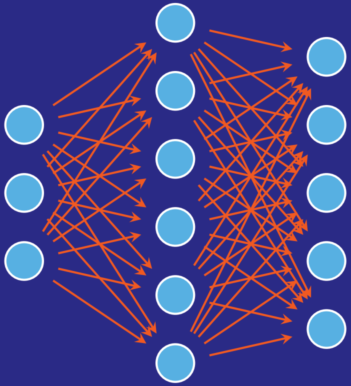
Abstract

This paper presents a novel framework for non-linear equivariant neural network layers on homogeneous spaces. The seminal work of Cohen et al. on equivariant *G*-CNNs on homogeneous spaces characterized the representation theory of such layers in the *linear* setting, finding that they are given by convolutions with kernels satisfying so-called steerability constraints. Motivated by the empirical success of non-linear layers, such as self-attention or input dependent kernels, we set out to generalize these insights to the *non-linear* setting. We derive generalized steerability constraints that any such layer needs to satisfy and prove the universality of our construction. The insights gained into the symmetry-constrained functional dependence of equivariant operators on feature maps and group elements informs the design of future equivariant neural network layers. We demonstrate how several common equivariant network architectures – *G*-CNNs, implicit steerable kernel networks, conventional and relative position embedded attention based transformers, and LieTransformers – may be derived from our framework.

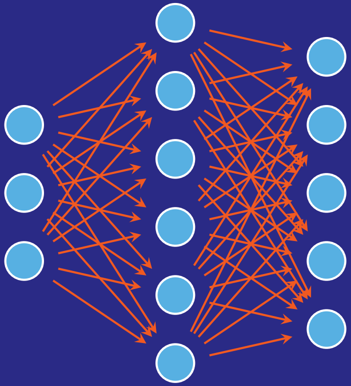
*Equal contribution, ordered by first name.

1

Machine learning models: Neural networks

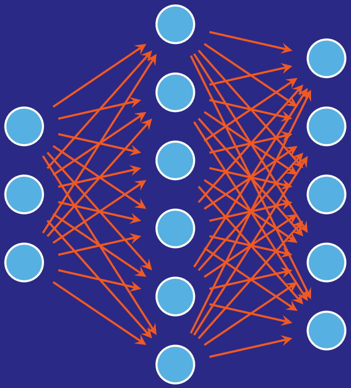


Machine learning models: Neural networks



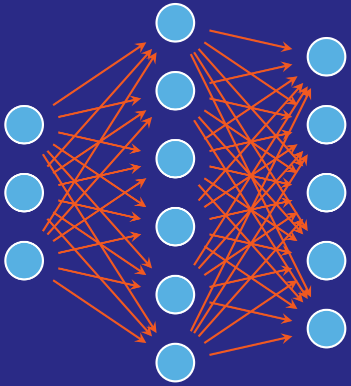
- Linear layers + non-linear activation functions

Machine learning models: Neural networks



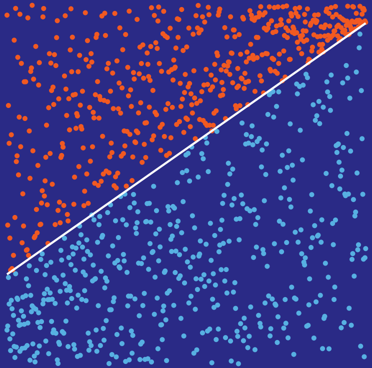
- Linear layers + non-linear activation functions
- Non-linear layers

Machine learning models: Neural networks

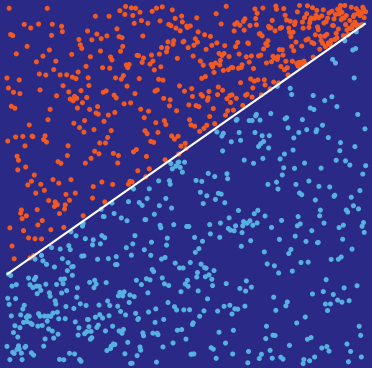


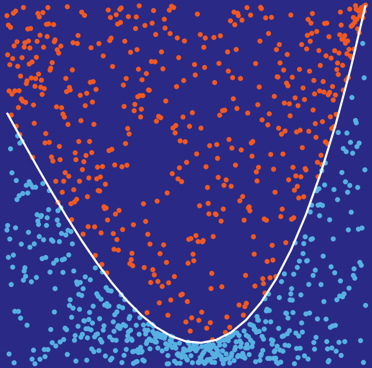
- Linear layers + non-linear activation functions
- Non-linear layers
- Combination of these

Why are non-linear maps important in NNs?

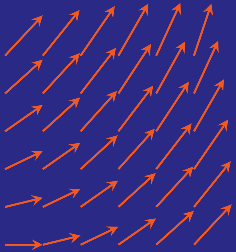


Why are non-linear maps important in NNs?



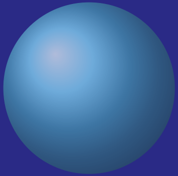


Example of data on the plane

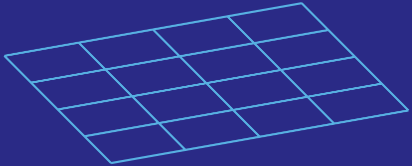


$$f(x) \in \mathbb{R}^2$$

Homogeneous spaces: G/H



$\mathrm{SO}(3)/\mathrm{SO}(2)$



$\mathbb{R}^2 \cong \mathrm{SE}(2)/\mathrm{SO}(2)$

Lifted Feature maps: the induced representation

$$\mathrm{Ind}_H^G \rho_{\mathrm{in}} = \left\{ f : G \to V_{\mathrm{in}} \mid f(gh) = \rho_{\mathrm{in}}(h^{-1}) f(g),\ \forall h \in H \right\}$$

Lifted Feature maps: the induced representation

$$\mathrm{Ind}_H^G \rho_{\mathrm{in}} = \left\{ f : G \to V_{\mathrm{in}} \mid f(gh) = \rho_{\mathrm{in}}(h^{-1}) f(g),\ \forall h \in H \right\} =: \mathcal{I}_{\rho_{\mathrm{in}}}$$

Lifted Feature maps: the induced representation

$$\mathrm{Ind}_H^G \rho_{\mathrm{in}} = \left\{ f : G \to V_{\mathrm{in}} \mid f(gh) = \rho_{\mathrm{in}}(h^{-1}) f(g),\ \forall h \in H \right\} =: \mathcal{I}_{\rho_{\mathrm{in}}}$$

$$[\rho_{\mathcal{I}_{\mathrm{in}}}(k) \triangleright f](g) = f(k^{-1}g)$$

Inspiration: CNNs and transformers

$$[\phi f](g) = \int_G \kappa(g^{-1}g')f(g')\mathrm{d}g'$$

G-Equivariant CNN on homogeneous space
(Cohen et al. 2019)

$$[\phi f](p) = \int_{\|u\|<R} \alpha(f)(p, q_u)V_u(f'(q_u))\mathrm{d}u$$

Gauge equivariant transformer

(He et al. 2021)

$$[\phi f](g) = \int_G \omega_f(g, g')f(g')\mathrm{d}g'$$

LieTransformer
(Hutchinson, Le Lan, Zaidi et al. 2020)

Inspiration: CNNs and transformers

$$[\phi f](g) = \int_G \kappa(g^{-1}g') f(g') \mathrm{d}g'$$

G-Equivariant CNN on homogeneous space
(Cohen et al. 2019)

$$[\phi f](p) = \int_{\|u\|<R} \alpha(f)(p, q_u) V_u(f'(q_u)) \mathrm{d}u$$

Gauge equivariant transformer

(He et al. 2021)

$$[\phi f](g) = \int_G \omega_f(g, g') f(g') \mathrm{d}g'$$

LieTransformer
(Hutchinson, Le Lan, Zaidi et al. 2020)

Inspiration: CNNs and transformers

$$[\phi f](g) = \int_G \kappa(g^{-1}g')f(g')\mathrm{d}g'$$

G-Equivariant CNN on homogeneous space
(Cohen et al. 2019)

$$[\phi f](p) = \int_{\|u\|<R} \alpha(f)(p, q_u)V_u(f'(q_u))\mathrm{d}u$$

Gauge equivariant transformer

(He et al. 2021)

$$[\phi f](g) = \int_G \omega_f(g, g')f(g')\mathrm{d}g'$$

LieTransformer
(Hutchinson, Le Lan, Zaidi et al. 2020)

Inspiration: CNNs and transformers

$$[\phi f](g) = \int_G \kappa(g^{-1}g')f(g')\mathrm{d}g'$$

G-Equivariant CNN on homogeneous space
(Cohen et al. 2019)

$$[\phi f](p) = \int_{\|u\|<R} \alpha(f)(p, q_u)V_u(f'(q_u))\mathrm{d}u$$

Gauge equivariant transformer

(He et al. 2021)

$$[\phi f](g) = \int_G \omega_f(g, g')f(g')\mathrm{d}g'$$

LieTransformer
(Hutchinson, Le Lan, Zaidi et al. 2020)

Join these into single integrand

Starting point

Want $\phi : \mathcal{I}_{\rho_{\text{in}}} \to \mathcal{I}_{\rho_{\text{out}}}$:

$$[\phi f](g) = \int_G \omega(f, g, g')\mathrm{d}g'$$

$$\omega : \mathcal{I}_{\rho_{\text{in}}} \times G \times G \to V_{\text{out}}$$

such that $[\phi f]$ is a feature map (Mackey-condition):

$$[\phi f](gh) = \rho_{\text{out}}(h^{-1})[\phi f](g)$$

for all $h \in H$.

Goal

Equivariance:

$$[\rho_{\mathcal{I}_{\text{out}}}(k) \triangleright [\phi f]](g) = [\phi[\rho_{\mathcal{I}_{\text{in}}}(k) \triangleright f]](g)$$

for all $k \in G$.

Whilst $[\phi f] \in \mathcal{I}_{\rho_{\text{out}}}$

Result

Theorem (Theorem 4.10 in Paper IV)

If ω satisfies $\omega(f,g,g') = \omega(\rho_{\mathcal{I}_{\text{in}}}(k) \triangleright f, kg, g')$ for all $k \in G$ then ϕ is equivariant and ω can be reduced to a two-argument map $\widehat{\omega}(f,g')$ satisfying the Mackey constraint

$$\widehat{\omega}(\rho_{\mathcal{I}_{\text{in}}}(h) \triangleright f, g') = \rho_{\text{out}}(h)\widehat{\omega}(f,g'), \quad \forall h \in H.$$

Hence ϕ can be formulated as

$$[\phi f](g) = \int_G \omega(f,g,g')\mathrm{d}g' = \int_G \widehat{\omega}(\rho_{\mathcal{I}_{\text{in}}}(g^{-1}) \triangleright f, g')\mathrm{d}g'.$$

Universality?

Does there for each equivariant map $\lambda : \mathcal{I}_{\rho_{\text{in}}} \to \mathcal{I}_{\rho_{\text{out}}}$ exist an $\widehat{\omega}$ such that

$$[\phi f](g) = \lambda[f](g)?$$

$$[\phi f](g) = \int_G \widehat{\omega}(\rho_{\mathcal{I}_{\text{in}}}(g^{-1}) \triangleright f, g')\mathrm{d}g'$$

Universality?

Does there for each equivariant map $\lambda : \mathcal{I}_{\rho_{\text{in}}} \to \mathcal{I}_{\rho_{\text{out}}}$ exist an $\widehat{\omega}$ such that

$$[\phi f](g) = \lambda[f](g)?$$

$$[\phi f](g) = \int_G \widehat{\omega}(\rho_{\mathcal{I}_{\text{in}}}(g^{-1}) \triangleright f, g')\mathrm{d}g'$$

Integration domain G compact

$$\widehat{\omega}(\rho_{\mathcal{I}_{\text{in}}}(g^{-1}) \triangleright f, g') = \frac{1}{\mathrm{vol}(G)}\lambda[\rho_{\mathcal{I}_{\text{in}}}(g^{-1}) \triangleright f](e)$$

Universality?

Does there for each equivariant map $\lambda : \mathcal{I}_{\rho_{\text{in}}} \to \mathcal{I}_{\rho_{\text{out}}}$ exist an $\widehat{\omega}$ such that

$$[\phi f](g) = \lambda[f](g)?$$

$$[\phi f](g) = \int_G \widehat{\omega}(\rho_{\mathcal{I}_{\text{in}}}(g^{-1}) \triangleright f, g')\mathrm{d}g'$$

Integration domain G compact

$$\widehat{\omega}(\rho_{\mathcal{I}_{\text{in}}}(g^{-1}) \triangleright f, g') = \frac{1}{\text{vol}(G)}\lambda[\rho_{\mathcal{I}_{\text{in}}}(g^{-1}) \triangleright f](e)$$

Have access to a δ-distribution or similar

$$\widehat{\omega}(\rho_{\mathcal{I}_{\text{in}}}(g^{-1}) \triangleright f, g') = \delta(g')\lambda[g^{-1}f](e)$$

Constructing new layers

1. Determine the homogeneous space G/H

Constructing new layers

1. Determine the homogeneous space G/H
2. From the arguments of $\widehat{\omega}(\rho_{\mathcal{I}_{\text{in}}}(g^{-1}) \triangleright f, g')$ isolate and construct $\widehat{\omega}(\rho_{\mathcal{I}_{\text{in}}}(g^{-1}) \triangleright f, g')$ from

Constructing new layers

1. Determine the homogeneous space G/H
2. From the arguments of $\widehat{\omega}(\rho_{\mathcal{I}_{\text{in}}}(g^{-1}) \triangleright f, g')$ isolate and construct $\widehat{\omega}(\rho_{\mathcal{I}_{\text{in}}}(g^{-1}) \triangleright f, g')$ from
   - $[\rho_{\mathcal{I}_{\text{in}}}(g^{-1}) \triangleright f](e) = f(g)$

Constructing new layers

1. Determine the homogeneous space G/H
2. From the arguments of $\widehat{\omega}(\rho_{\mathcal{I}_{\text{in}}}(g^{-1}) \triangleright f, g')$ isolate and construct $\widehat{\omega}(\rho_{\mathcal{I}_{\text{in}}}(g^{-1}) \triangleright f, g')$ from
   - $[\rho_{\mathcal{I}_{\text{in}}}(g^{-1}) \triangleright f](e) = f(g)$
   - $[\rho_{\mathcal{I}_{\text{in}}}(g^{-1}) \triangleright f](g') = f(gg')$

Constructing new layers

1. Determine the homogeneous space G/H
2. From the arguments of $\widehat{\omega}(\rho_{\mathcal{I}_{\text{in}}}(g^{-1}) \triangleright f, g')$ isolate and construct $\widehat{\omega}(\rho_{\mathcal{I}_{\text{in}}}(g^{-1}) \triangleright f, g')$ from
   - $[\rho_{\mathcal{I}_{\text{in}}}(g^{-1}) \triangleright f](e) = f(g)$
   - $[\rho_{\mathcal{I}_{\text{in}}}(g^{-1}) \triangleright f](g') = f(gg')$
   - $\psi(g')$

   such that

$$\widehat{\omega}(\rho_{\mathcal{I}_{\text{in}}}(hg^{-1}) \triangleright f, g') = \rho_{\text{out}}(h)\widehat{\omega}(\rho_{\text{Ind}_{\text{in}}}(g^{-1}) \triangleright f, g'), \ \forall h \in H.$$

Special instances of this framework

Convolution based

G-CNN:

$$\hat{\kappa}(g')[g^{-1}f](g')$$

Implicit steerable CNN:

$$\kappa\big(g', [g^{-1}f](e), [g^{-1}f](g')\big)\,[g^{-1}f](g')$$

General framework

$$\omega(f, g, g')$$

Equivariance

$$\hat{\omega}(g^{-1}f, g')$$

Attention based

Self-attention:

$$\mathrm{Softmax}\left\{\frac{[g^{-1}f](e)^\top W_Q^\top W_K\,[g^{-1}f](g')}{\sqrt{d}}\right\} \cdot W_V\,[g^{-1}f](g')$$

Relative Position Bias self-attention:

$$\mathrm{Softmax}\left\{\frac{[g^{-1}f](e)^\top W_Q^\top W_K\,[g^{-1}f](g')}{\sqrt{d}} + \psi(g')\right\} \cdot W_V\,[g^{-1}f](g')$$

LieTransformer:

$$\mathrm{norm}\{\alpha([g^{-1}f](e), [g^{-1}f](g'), g')\} \cdot W_V\,[g^{-1}f](g')$$

Outlook

- Investigate how this works for data on general manifolds

Outlook

- Investigate how this works for data on general manifolds
- Design new equivariant layers using this framework

Outlook

- Investigate how this works for data on general manifolds
- Design new equivariant layers using this framework
- Classify layers based on the construction of $\widehat{\omega}$

Thanks!