Geometric deep learning for data on manifolds, and spherical images

Half-way seminar,

Oscar Carlsson, Department of Mathematical Sciences

For the Geometric Deep Learning seminar series on 2023-02-09, at the Department of Mathematical Sciences

Material

This presentation is based on the following papers:

- "**Geometric Deep Learning and Equivariant Neural Networks**" by Jan E. Gerken, Jimmy Aronsson, Oscar Carlsson, Hampus Linander, Fredrik Ohlsson, Christoffer Petersson, Daniel Persson. *Under review for AIRE.*
- "**Equivariance versus Augmentation for Spherical Images**" by Jan E. Gerken, Oscar Carlsson, Hampus Linander, Fredrik Ohlsson, Christoffer Petersson, Daniel Persson. *Accepted for ICML22.*

Outline

Introduction
-Machine learning

Examples of machine learning applications

- Language models
- AlphaFold
- Generative models
- Image analysis
  - Object detection
  - Semantic segmentation
  - Depth estimation
  - Classification

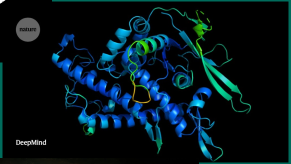
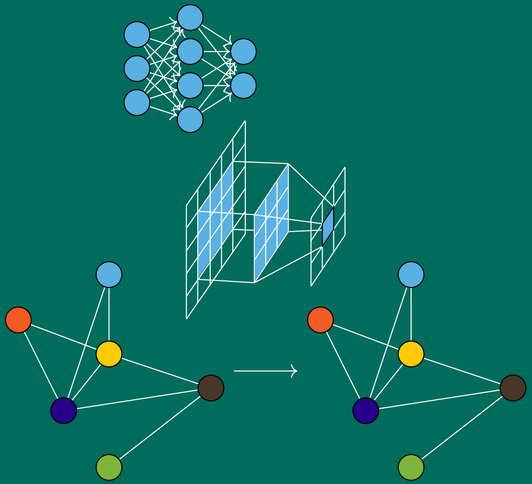


Image analysis example

From the WoodScape dataset (Yogamani et al. 2019).

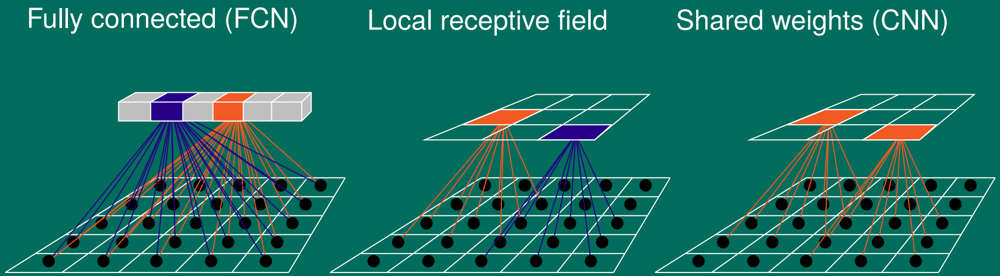
Some architecture examples

- Fully connected networks (FCN)
- Convolutional neural networks (CNN)
- Graph networks

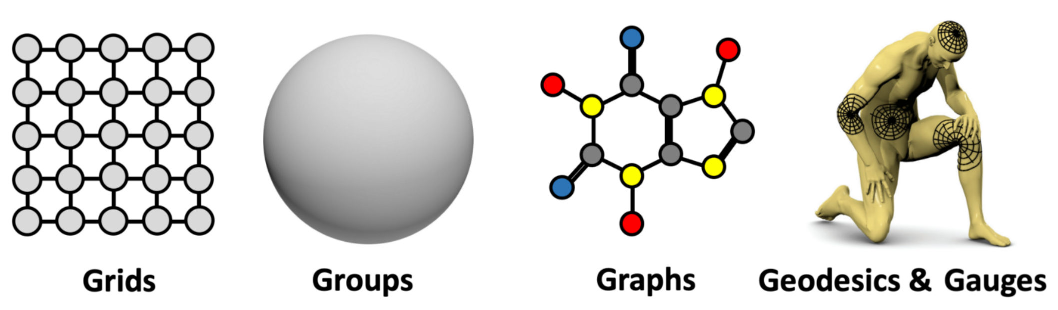


FCN vs CNN

Q: Why does FCN not work as well as CNN for image analysis?

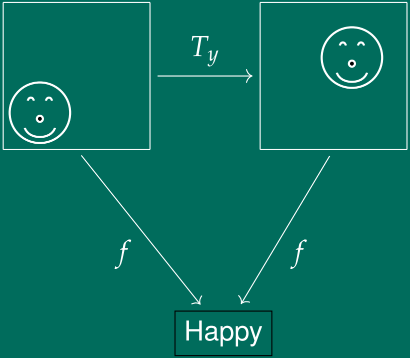


What is geometric deep learning?

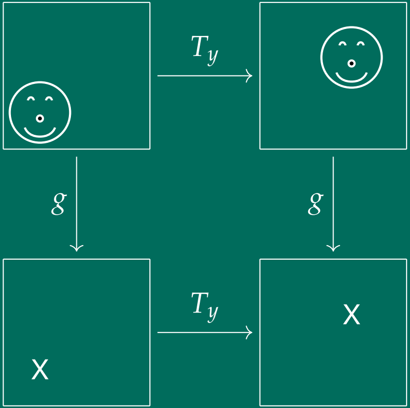


(Bronstein et al. 2021)

Invariance and equivariance



$$(T_y f)(x) = f(x)$$



$$(T_y g)(x) = g(T_y x)$$

Group actions and representations

Important viewpoint/assumption: data is some function on an underlying space. In this viewpoint *layers maps data functions to data functions*.

E.g. Viewing input colour image as a function $f : \mathbb{R}^2 \to \mathbb{R}^3$.

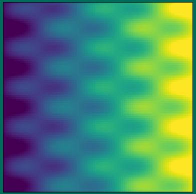
Group of transformations G acting on the input data, e.g.

$$(g \triangleright f)(x) = \rho(g) f(g^{-1} \triangleright x)$$

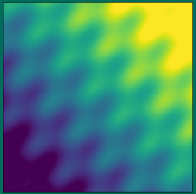
How the data function f transforms is given by ρ, sometimes this is referred to as different *data types*.

Example of data types: scalar

$\phi(x)$

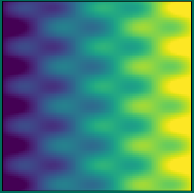


$(L_g\phi)(x) = \phi(g^{-1}x)$

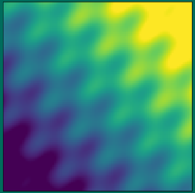


Example of data types: scalar and vector

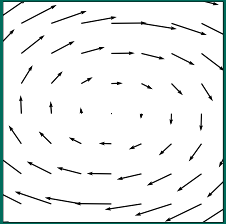
$\phi(x)$



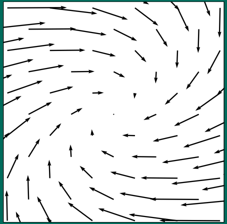
$(L_g\phi)(x) = \phi(g^{-1}x)$



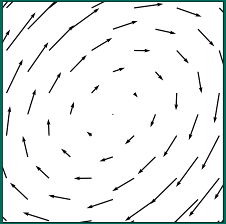
$f(x)$



$f(g^{-1}x)$

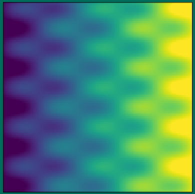


$(L_g f)(x)$

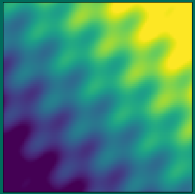


Example of data types: scalar and vector

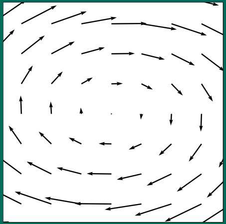
$\phi(x)$



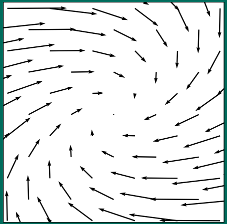
$(L_g\phi)(x) = \phi(g^{-1}x)$



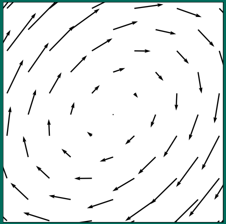
$f(x)$



$f(g^{-1}x)$

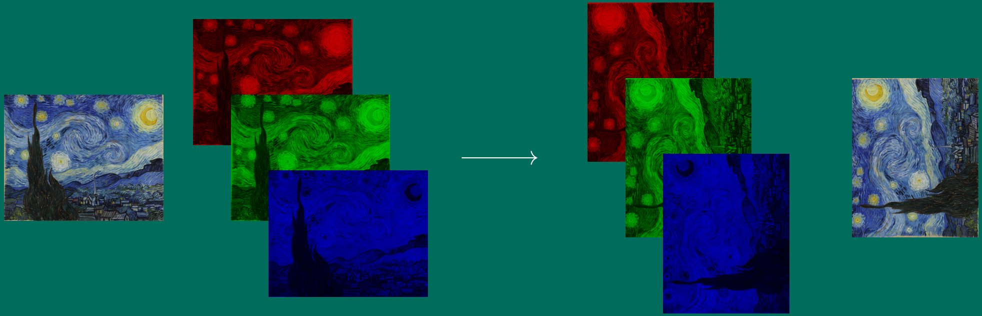


$(L_gf)(x) = \rho(g)f(g^{-1}x)$



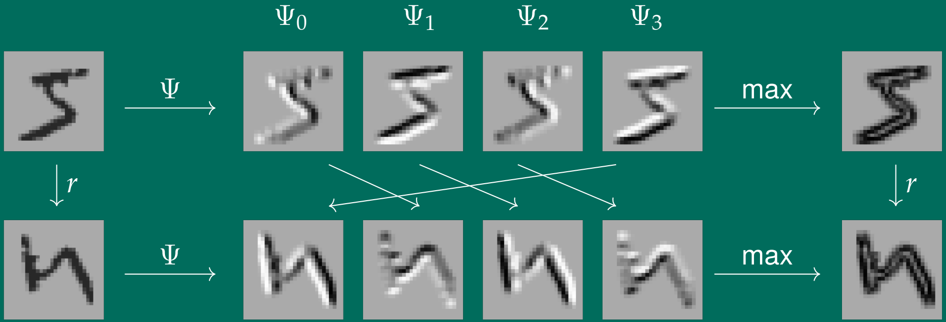
Example in practice: ordinary image

Group $G = \mathrm{SO}(2)$ acting as $(g \triangleright f)(x) = \rho(g)f(g^{-1} \triangleright x)$ where $\rho = \bigoplus_{i=1}^{3} \mathrm{id}$ where id is the trivial representation.



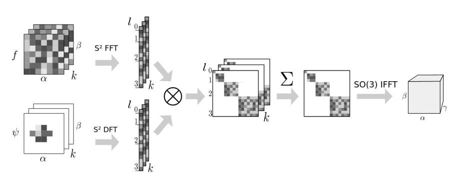
Regular representation approach

- Copy and rotate filters (Bekkers et al. 2018; T. S. Cohen and Welling 2016)



Continuous groups: spectral methods

- Spherical cnn: S2CNN (T. S. Cohen, Geiger et al. 2018)



$$[\psi \star f](R) = \langle L_R \psi, f \rangle = \int_{S^2} \sum_{k=1}^{K} \psi_k(R^{-1}x) f_k(x) dx, \quad R \in \mathrm{SO}(3)$$

Solving the kernel constraint

- Kernel constraint for steerable kernels $k : \mathbb{R}^2 \to \mathbb{R}^{\dim_{\text{out}} \times \dim_{\text{in}}}$

  $$k(gx) = \rho_{\text{out}}(g) k(x) \rho_{\text{in}}(g^{-1}) \,, \quad \forall g \in G$$

  change basis to irreps and redefine kernel k gives

  $$k(gx) = \left(\bigoplus_{i \in I_{\text{out}}} \psi_i(g) \right) k(x) \left(\bigoplus_{j \in I_{\text{in}}} \psi_j^{-1}(g) \right) .$$

  (Weiler and Cesa 2019)

The papers

- Paper I: Geometric Deep Learning and Equivariant Neural Networks (Gerken, Aronsson et al. 2021)
- Paper II: Equivariance versus Augmentation for Spherical Images (Gerken, Carlsson et al. 2022)

Paper I: Geometric Deep Learning and Equivariant Neural Networks
-Overview

Covers mathematical approaches to equivariant networks for data defined on a continuous (possibly discretised) spaces covering

- General (Riemannian) manifolds
- Meshes
- General Homogenous spaces
- Spectral networks on the sphere

Gauge equivariant convolutions on Riemannian manifolds

Lifting the framework introduced in T. Cohen et al. 2020 and refined in Aronsson 2021 to a general Riemannian manifold where the transformations are local coordinate changes.

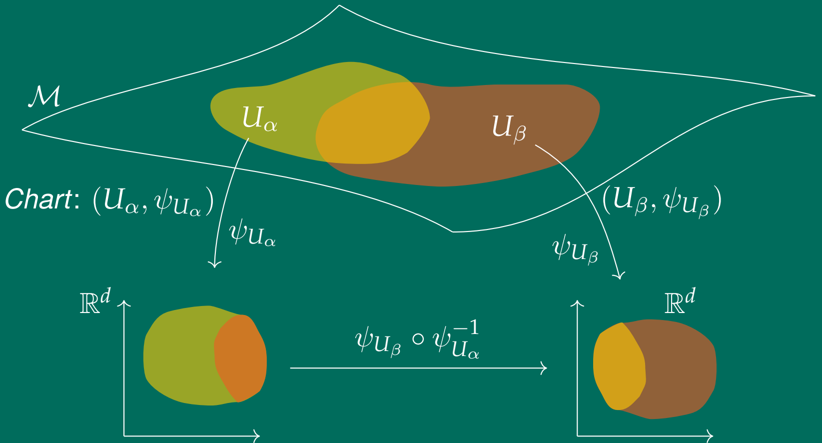
Necessary concepts:

- Manifolds
- Fibre bundles (vector bundles, principal bundles, and associated bundles)
- Gauge transformations
- Parallel transport

Gauge equivariant convolution $\phi : C_c(P, \rho) \to C_c(P, \eta)$ by

$$(\phi f)(p) = [\kappa \star f](p) = \int_{B_R} \kappa(X) f(\mathcal{T}_X p) \mathrm{vol}_{T_x \mathcal{M}} \, .$$

Recap on manifolds

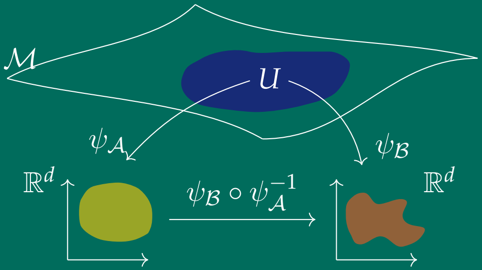


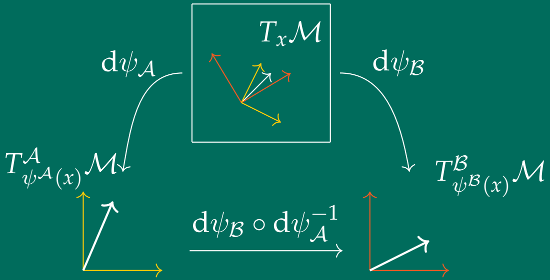
Atlas: $\{(U_\alpha, \psi_{U_\alpha}) : \alpha \in I\}$, transition maps $\tau_{\alpha,\beta} = \psi_{U_\beta} \circ \psi_{U_\alpha}^{-1}$ for $U_\alpha \cap U_\beta \neq \emptyset$

Gauge transformations (change of local coordinates)

A chart $(U, \psi_\mathcal{A})$ is an assignment of local coordinates to $U \subseteq \mathcal{M}$, choosing another chart $(U, \psi_\mathcal{B})$ gives a different set of local coordinates.

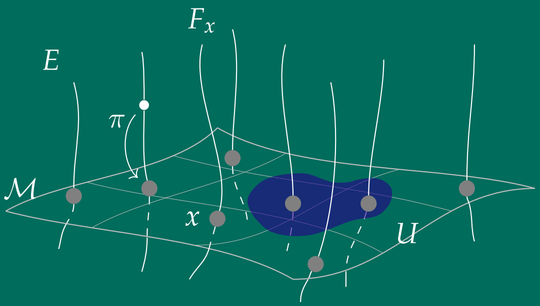
The map $\psi_\mathcal{B} \circ \psi_\mathcal{A}^{-1}$ induces a gauge transformation in the tangent spaces $T_x\mathcal{M} \cong \mathbb{R}^d$ for $x \in U$.





Recap on bundles

- *Bundle*: A bundle $(E, \pi, \mathcal{M})$, $E \xrightarrow{\pi} \mathcal{M}$, is a triple consisting of a (total) space E, a surjective continuous map $\pi : E \to \mathcal{M}$ (projection), and a base manifold $\mathcal{M}$.
- *Vector bundle*: Each fibre is a vector space V
  - Ex: Tangent bundle $T\mathcal{M}$, $T_x\mathcal{M} \cong \mathbb{R}^d$, $\forall x \in \mathcal{M}$
- *Principal* K-bundle: Each fibre is homeomorphic to the group K



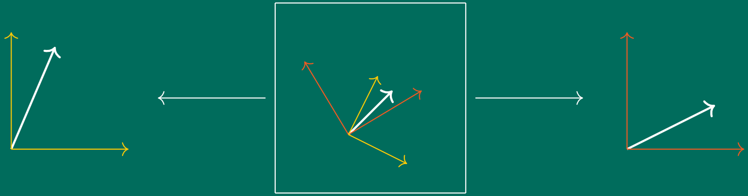
Associated vector bundles

Associated bundle: A bundle constructed from a principal K-bundle P and a representation (V, ρ), i.e. $k \triangleright v = \rho(k)v$, through the equivalence relation

$$(p, v) \sim_\rho (p \triangleleft k, k^{-1} \triangleright v)$$

on the space $P \times V$. The quotient $P \times V/\sim_\rho =: E_\rho$ with the projection $\pi([p, v]) = \pi_P(p)$ makes E_ρ to a vector bundle over the same base space as P.

Intuition: $[p, v]$ is a geometric (i.e. coordinate free) vector. A specific instance $(p, v) \in [p, v]$ expresses the geometric vector in a "frame"/basis p and components v.



Framework used in Aronsson 2021

- *Data point*: section $\sigma \in \Gamma(E_\rho)$ of associated bundle
- *Feature map*: map $f \in C_c(P, \rho)$ satisfying

$$k \triangleright f(p) = f(p \triangleleft k^{-1})$$

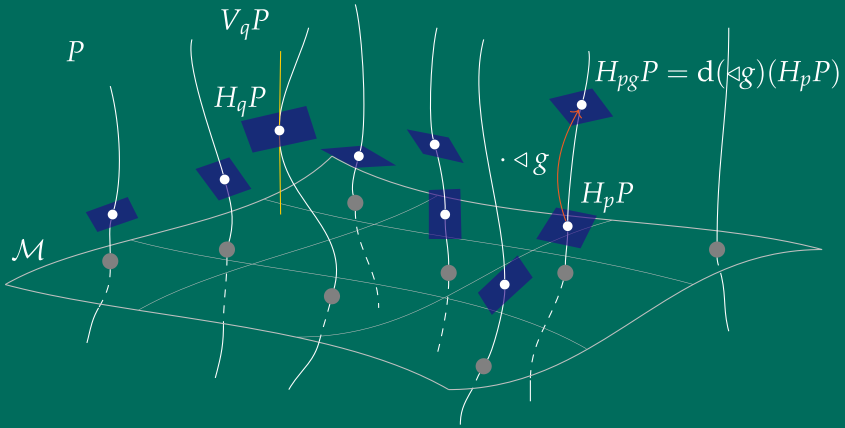
  or $k \circ f = f \circ k^{-1}$.
- **Theorem** $\varphi_\rho : C_c(P, \rho) \to \Gamma(E_\rho)$ acting by

$$f \mapsto \sigma_f = [\pi^{-1}, f \circ \pi^{-1}]$$

  is an isomorphism between $C_c(P, \rho)$ and $\Gamma(E_\rho)$ (Kolar et al. 1993).

Connection and parallel transport

Principal Ehresmann connection: a smooth subbundle H of TP such that $H \oplus V = TP$ where $V = \ker(\mathrm{d}\pi : TP \to T\mathcal{M})$. At each $p \in P$ then $T_pP = H_p \oplus V_p$



Connection and parallel transport

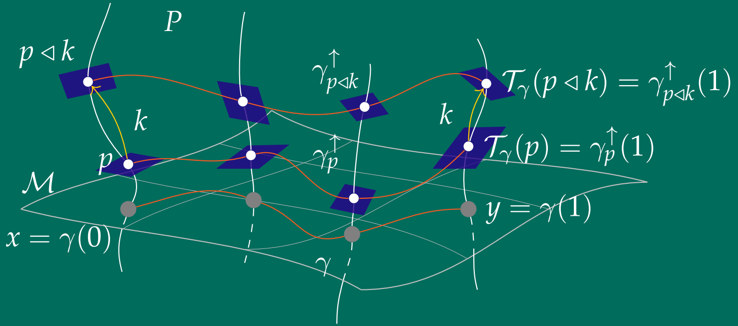
Horizontal lift $\gamma_p^{\uparrow} : [0,1] \to P$ of curve $\gamma : [0,1] \to \mathcal{M}$ s.t. $\gamma(0) = x$ and $\gamma(1) = y$.

Parallel transport in P by

$$\mathcal{T}_\gamma p = \gamma_p^{\uparrow}(1)\,,$$

which is equivariant to the principal action

$$\mathcal{T}_\gamma(p) \triangleleft k = \mathcal{T}_\gamma(p \triangleleft k)\,.$$



Gauge equivariant convolution

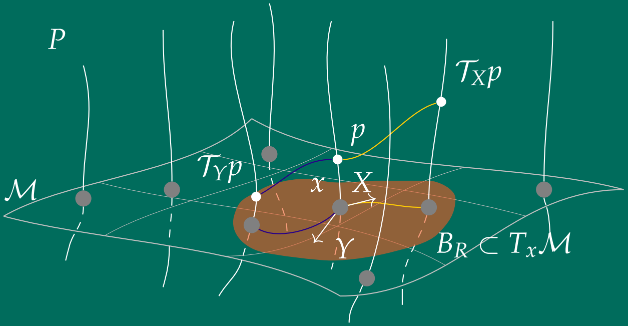
Kernel: $\kappa : T\mathcal{M} \to \mathrm{Hom}(E_\rho, E_\eta)$.
In local coordinates $\mathcal{A}$ we have

$$\kappa^{\mathcal{A}} : \mathbb{R}^d \to \mathbb{R}^{\dim(\eta)\times\dim(\rho)}$$

transforming as

$$\kappa^{\mathcal{A}}(X^{\mathcal{A}}) = \eta(k^{-1})\kappa^{\mathcal{B}}(X^{\mathcal{B}})\rho(k)$$

where $X^{\mathcal{B}} = k \triangleright X^{\mathcal{A}}$. Then $\phi : C_c(P,\rho) \to C_c(P,\eta)$ by



$$(\phi f)(p) = [\kappa \star f](p) = \int_{B_R} \kappa(X) f(\mathcal{T}_X p) \mathrm{vol}_{T_x\mathcal{M}} .$$

Calculations

Evaluate the convolution at a frame $p^{\mathcal{A}}$ and let $k \in K$ be s.t. $p^{\mathcal{B}} = p^{\mathcal{A}} \triangleleft k^{-1}$ then

$$
\begin{aligned}
k \triangleright (\phi f)(p^{\mathcal{A}}) = k \triangleright [\kappa^{\mathcal{A}} \star f](p^{\mathcal{A}}) = \eta(k) \int_{B_R^{\mathcal{A}}} \kappa^{\mathcal{A}}(X^{\mathcal{A}}) f(\mathcal{T}_X p^{\mathcal{A}}) \mathrm{vol}^{\mathcal{A}}_{T_x\mathcal{M}} \\
&= \int_{B_R^{\mathcal{A}}} \kappa^{\mathcal{B}}(X^{\mathcal{B}}) \rho(k) f(\mathcal{T}_X p^{\mathcal{A}}) \mathrm{vol}^{\mathcal{A}}_{T_x\mathcal{M}} = \left\{ \rho(k) f(p) = f(p \triangleleft k^{-1}) \right\} \\
&= \int_{B_R^{\mathcal{A}}} \kappa^{\mathcal{B}}(X^{\mathcal{B}}) f\left((\mathcal{T}_X p^{\mathcal{A}}) \triangleleft k^{-1} \right) \mathrm{vol}^{\mathcal{A}}_{T_x\mathcal{M}} = \{ \mathcal{T}_X(p) \triangleleft k = \mathcal{T}_X(p \triangleleft k) \} \\
&= \int_{B_R^{\mathcal{A}}} \kappa^{\mathcal{B}}(X^{\mathcal{B}}) f\left(\mathcal{T}_X(p^{\mathcal{A}} \triangleleft k^{-1}) \right) \mathrm{vol}^{\mathcal{A}}_{T_x\mathcal{M}} \\
&= \int_{B_R^{\mathcal{B}}} \kappa^{\mathcal{B}}(X^{\mathcal{B}}) f(\mathcal{T}_X p^{\mathcal{B}}) \mathrm{vol}^{\mathcal{B}}_{T_x\mathcal{M}} \\
&= (\phi f)(p^{\mathcal{B}}) = (\phi f)(p^{\mathcal{A}} \triangleleft k^{-1}) \quad \Rightarrow \quad \phi f \text{ is a feature map}
\end{aligned}
$$

$$
k \circ \phi \circ f = \phi \circ f \circ k^{-1} = \phi \circ k \circ f \quad \Rightarrow \quad k \circ \phi = \phi \circ k \quad \phi \text{ equivariant}
$$

Some details on the kernel

Defining the kernel as

$$\kappa : T\mathcal{M} \to \mathrm{Hom}(E_\rho, E_\eta)$$

allows for different kernels at different $x \in \mathcal{M}$, i.e. a *kernel field*.

Noting that in local a local coordinate chart $\mathcal{A}$ at $x \in \mathcal{M}$ this kernel becomes

$$\mathcal{K}^{\mathcal{A}} : \mathbb{R}^d \to \mathbb{R}^{\dim(\eta)\times\dim(\rho)}.$$

To get weight sharing over $x \in \mathcal{M}$ pick

$$K : \mathbb{R}^d \to \mathbb{R}^{\dim(\eta)\times\dim(\rho)}$$

and set

$$\mathcal{K}^{\mathcal{X}} = K\,,$$

for any gauge $\mathcal{X}$. This kernel can now be shared over different $x \in \mathcal{M}$

(Weiler, Forré et al. 2021)

Paper II: Equivariance versus Augmentation for Spherical Images
-Background

Data augmentation

- Simple to implement
- ”Generating more points on the data manifold“
- No guarantee for ”exact“ equivariance
- Increases training time

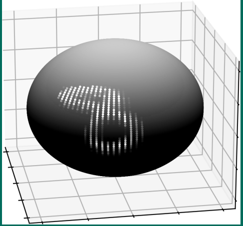
Equivariance

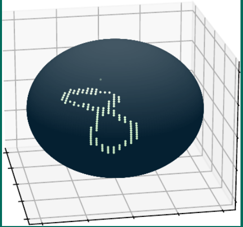
- Can be more complicated to code with more details in the model
- Theoretically more complicated
- Theoretical insurance that symmetry is respected

Often claimed: equivariant networks are better, but little is actually investigated; i.e. what is the limit of *geometrical data augmentation*?

Paper II: Setup

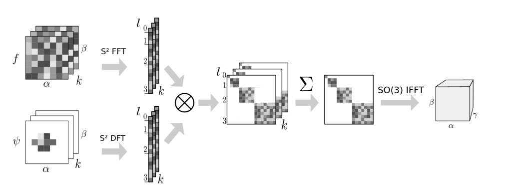
- Dataset: MNIST images projected to the sphere.
  - Data augmentation: pick random $g \in \mathrm{SO}(3)$ and act on an image
- Models:
  - S2CNN (T. S. Cohen, Geiger et al. 2018)
  - Ordinary CNN
- Tasks:
  - Classification (invariant)
  - Semantic segmentation (equivariant)



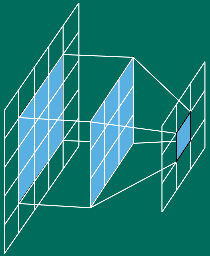


Paper II: Networks

S2CNN :



CNN:



Paper II: Network selection

20 random networks for each parameter range and type.

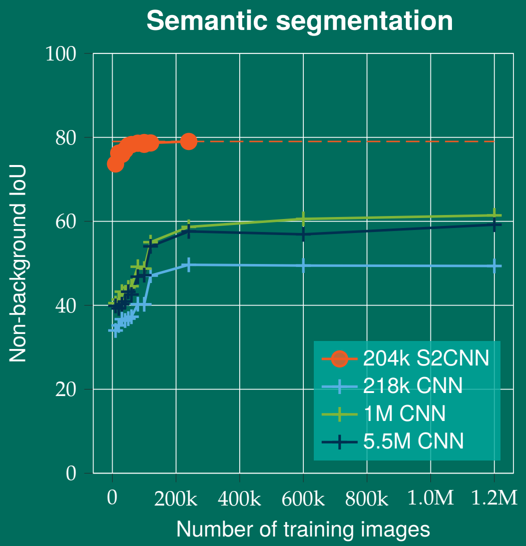
S2CNN

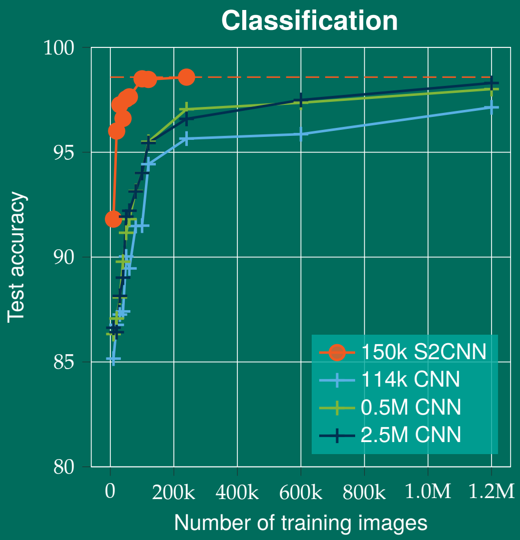
- Semantic segmentation
  - 204k
  - (820k)
- Classification
  - 150k

CNN

- Semantic segmentation
  - 218k
  - 1M
  - 5.5M
- Classification
  - 114k
  - 0.5M
  - 2.5M

Paper II: Some results





Paper II: Latency and training times

Latency

Model	Latency (ms)	Throughput (N/s)
204k S2CNN	111 ± 0.6	9.0 ± 0.04
200k CNN	5.93 ± 0.24	169 ± 5.8

Training time

Model	Accuracy	Training time
150k S2CNN	97.64%	15h
5M CNN	97.49%	26h

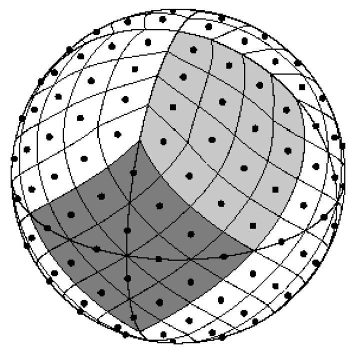
Paper II: Conclusions and takeaways

Whether equivariance or data augmentation is better for an application depends on several factors:

- Which symmetries exists in your problem?
- Is the symmetry “amenable” to your model?
- How much data do you have access to?
- Is training time important?
- Is throughput / latency important?

Future work

- Develop mathematical formalism for equivariant transformers
- Combining local and global symmetries
- Effects of sampling data with the HEALPix grid (Gorski et al. 2005)



Thank you

Thank you for listening!

References I

- ‘Homogeneous Vector Bundles and G-Equivariant Convolutional Neural Networks’. 11th May 2021. arXiv: 2105.05400 [cs, math, stat].
- ‘Roto-Translation Covariant Convolutional Networks for Medical Image Analysis’. 11th June 2018. arXiv: 1804.03393 [cs, math].
- ‘Geometric Deep Learning: Grids, Groups, Graphs, Geodesics, and Gauges’. 27th Apr. 2021. arXiv: 2104.13478 [cs, stat].
- ‘A General Theory of Equivariant CNNs on Homogeneous Spaces’. 9th Jan. 2020. arXiv: 1811.02017 [cs, stat].
- ‘Spherical CNNs’. 25th Feb. 2018. arXiv: 1801.10130 [cs, stat].
- ‘Group Equivariant Convolutional Networks’. 3rd June 2016. arXiv: 1602.07576 [cs, stat].

References II

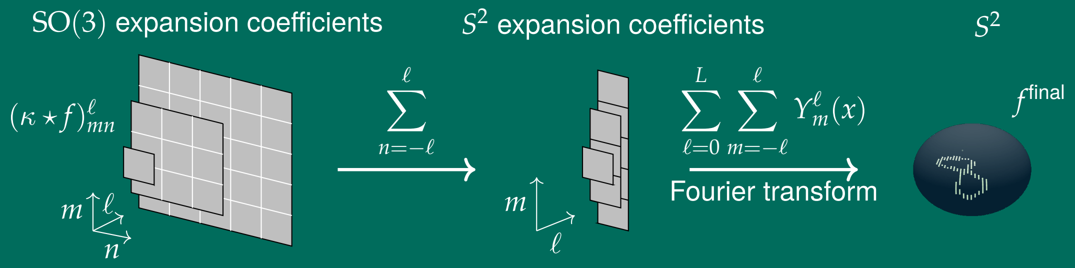
- 'Geometric Deep Learning and Equivariant Neural Networks'. 28th May 2021. arXiv: 2105.13926 [hep-th].
- *Equivariance versus Augmentation for Spherical Images*. 12th July 2022. DOI: 10.48550/arXiv.2202.03990. arXiv: 2202.03990 [cs].
- 'HEALPix – a Framework for High Resolution Discretization, and Fast Analysis of Data Distributed on the Sphere'. In: *The Astrophysical Journal* 622.2 (Apr. 2005), pp. 759–771. ISSN: 0004-637X, 1538-4357. DOI: 10.1086/427976. arXiv: astro-ph/0409513.
- *Natural Operations in Differential Geometry*. 1993rd edition. Berlin ; New York: Springer, 4th Feb. 1993. 440 pp. ISBN: 978-3-540-56235-1.
- 'General E(2)-Equivariant Steerable CNNs'. 19th Nov. 2019. arXiv: 1911.08251 [cs, eess].

References III

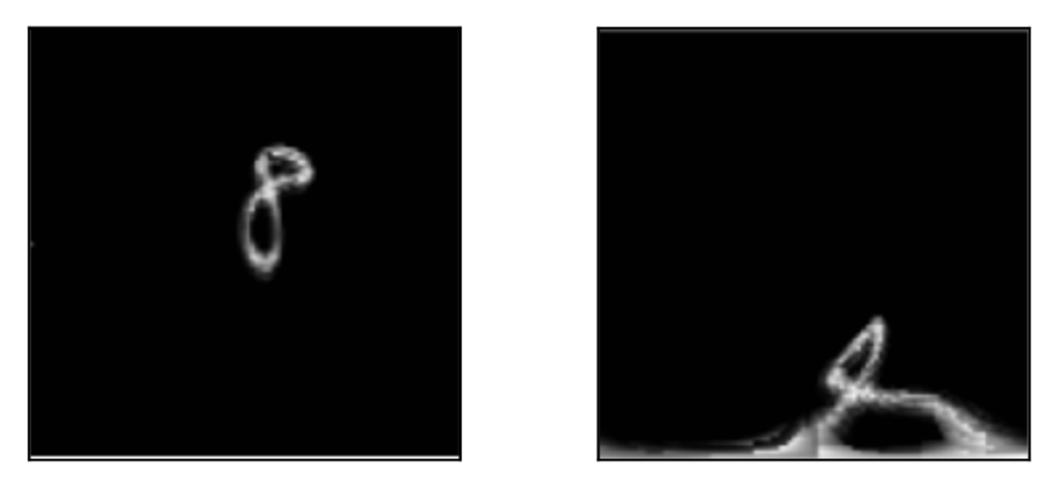
Coordinate Independent Convolutional Networks – Isometry and Gauge Equivariant Convolutions on Riemannian Manifolds. 10th June 2021. DOI: 10.48550/arXiv.2106.06020. arXiv: 2106.06020 [cs, stat].

'WoodScape: A Multi-Task, Multi-Camera Fisheye Dataset for Autonomous Driving'. 16th Aug. 2019. arXiv: 1905.01489 [cs, stat].

Paper II: S2CNN extension



Paper II: Rotated data points



Mathematical formalism for supervised learning

Data: from $\mathcal{X}$

Labels: from $\mathcal{Y}$

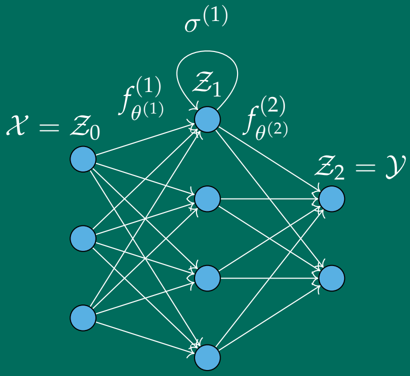
Layers: $f_{\theta^{(l)}}^{(l)} : \mathcal{Z}_{l-1} \to \mathcal{Z}_l$

Non-linearity: $\sigma^{(l)} : \mathcal{Z}_l \to \mathcal{Z}_l$

Model: $\mathcal{N}_\theta : \mathcal{X} \to \mathcal{Y}, \quad \mathcal{N}_\theta = \bigcirc_{l=1}^{N} \sigma^{(l)} \circ f_{\theta^{(l)}}^{(l)}$

Dataset: $D = \{(x_i, y_i)\}_{i=1\ldots N} \subseteq \mathcal{X} \times \mathcal{Y}$

Loss function: $\mathcal{L} : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$



Train the model by minimising the empirical risk through updating the parameters θ

$$\frac{1}{|D|} \sum_{(x_i, y_i) \in D} \mathcal{L}(\mathcal{N}_\theta(x_i), y_i)$$