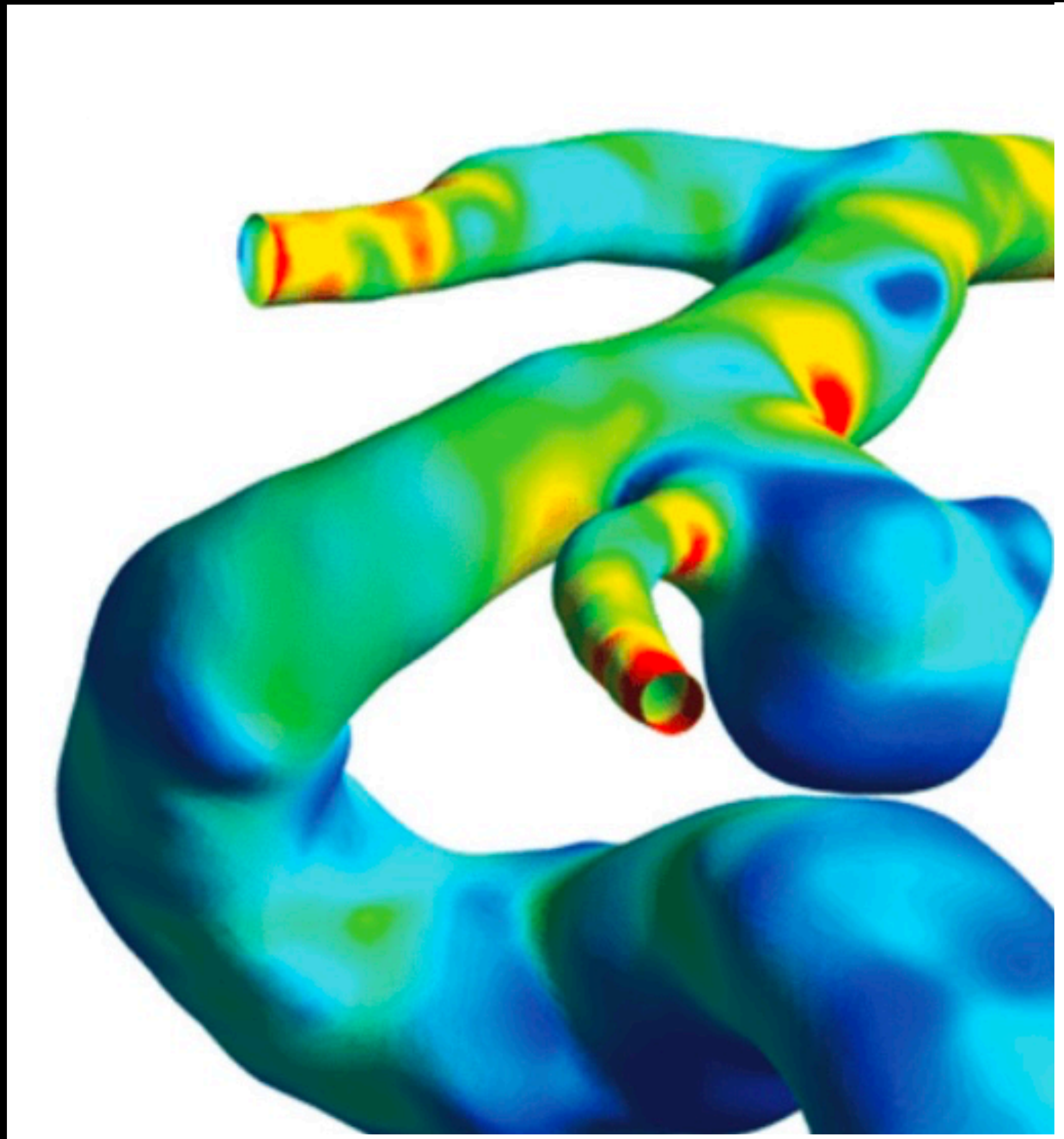
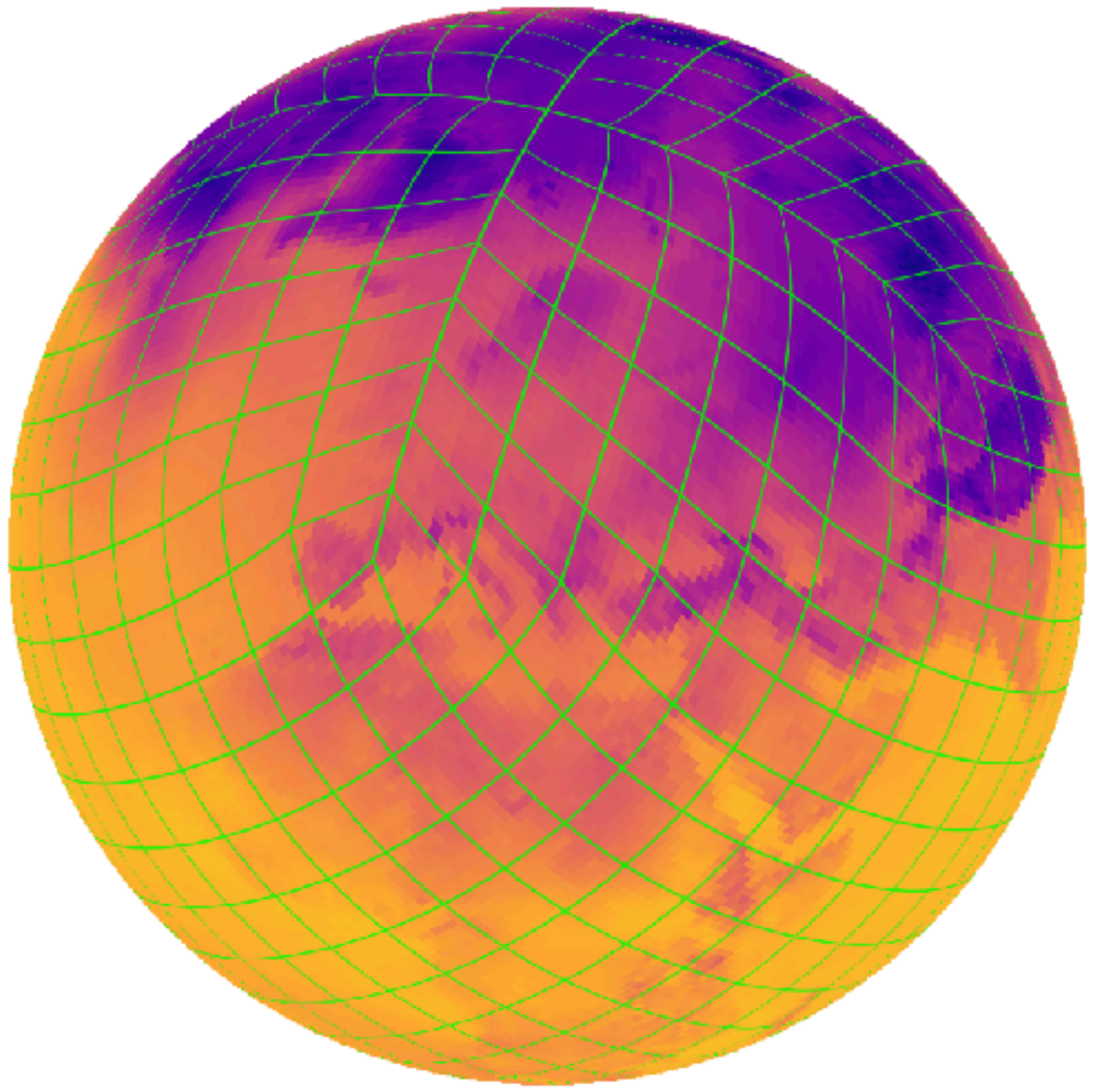
Geometric Deep Learning

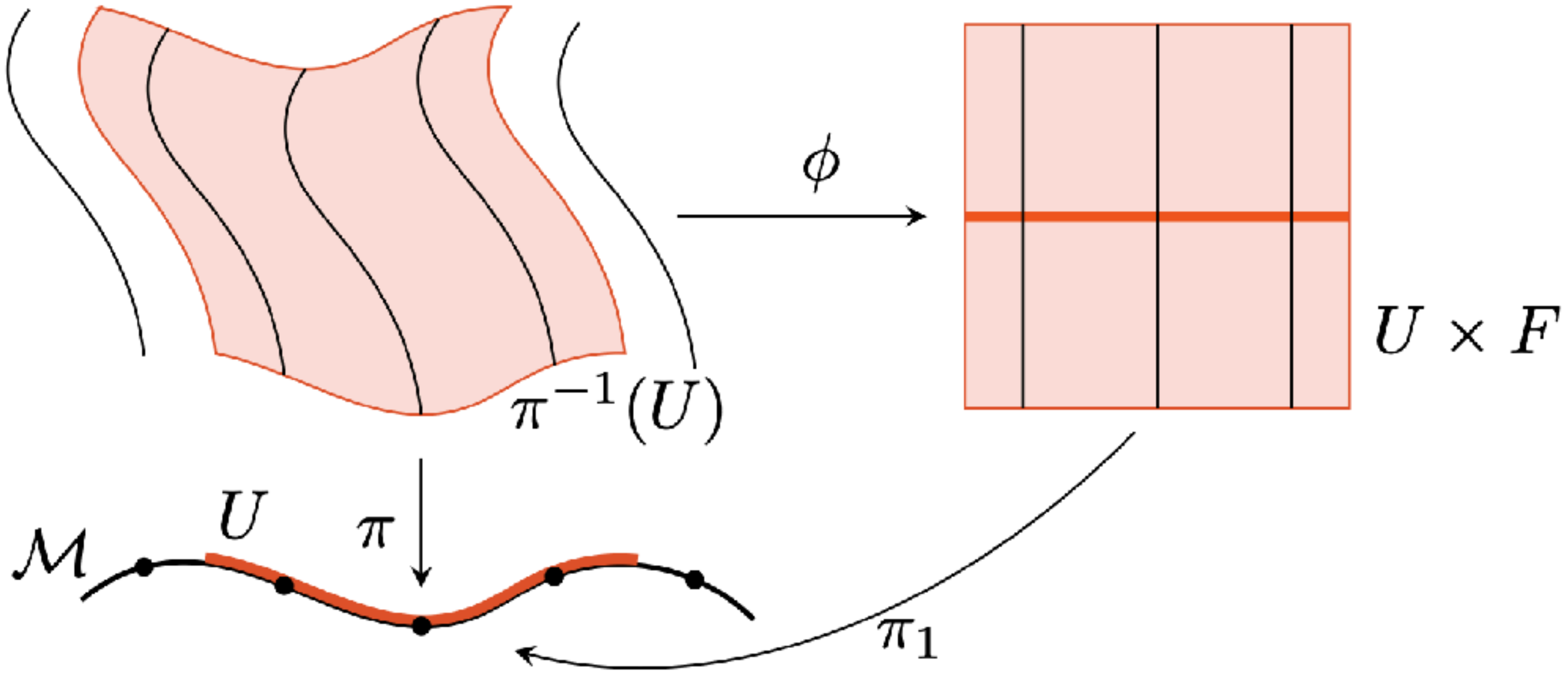
- From equivariance to weather predictions -

Daniel Persson
Department of Mathematical Sciences
Chalmers University of Technology
University of Gothenburg

AI4Physics Workshop
Uppsala University
April 17, 2026

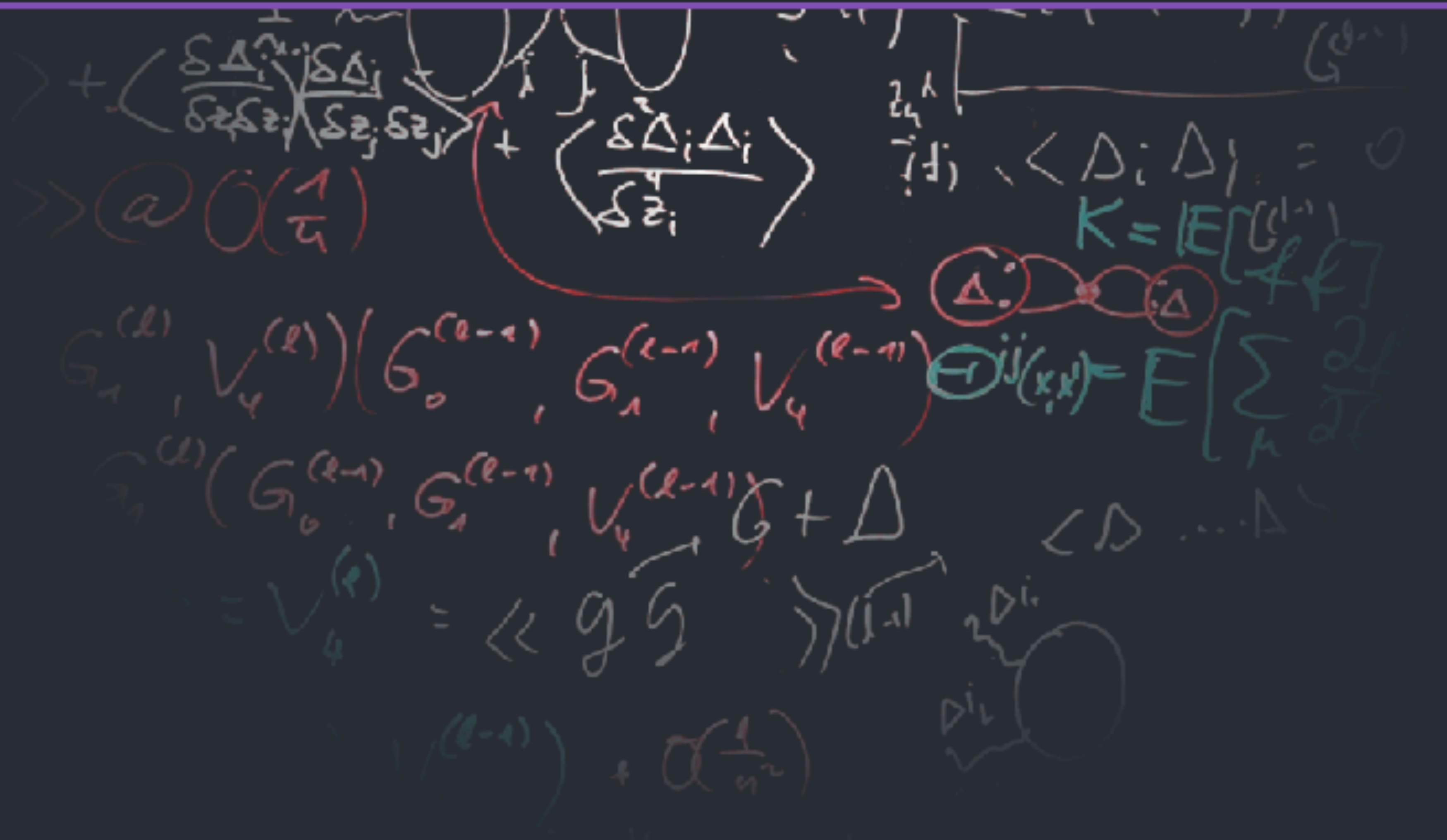






GAPinDNNs Home Members Research Output Seminar Teaching Positions

Geometry, Algebra and Physics in Deep Neural Networks



The research group on Geometry, Algebra and Physics in Deep Neural Networks (GAPinDNNs) is based at the Department for Mathematical Sciences at Chalmers University of Technology and the University of Gothenburg. Our vision is to develop a mathematical foundation for deep learning which elevates the field into a theoretically well-grounded science.

News

New Preprint on The Geometry of Polynomial Group Convolutional Neural Networks

31 Mar 2026

Daniel Persson, together with collaborators Yacoub Hendi and Magdalena Larfors, have published a new preprint on *The Geometry of Polynomial Group Convolutional Neural Networks*.

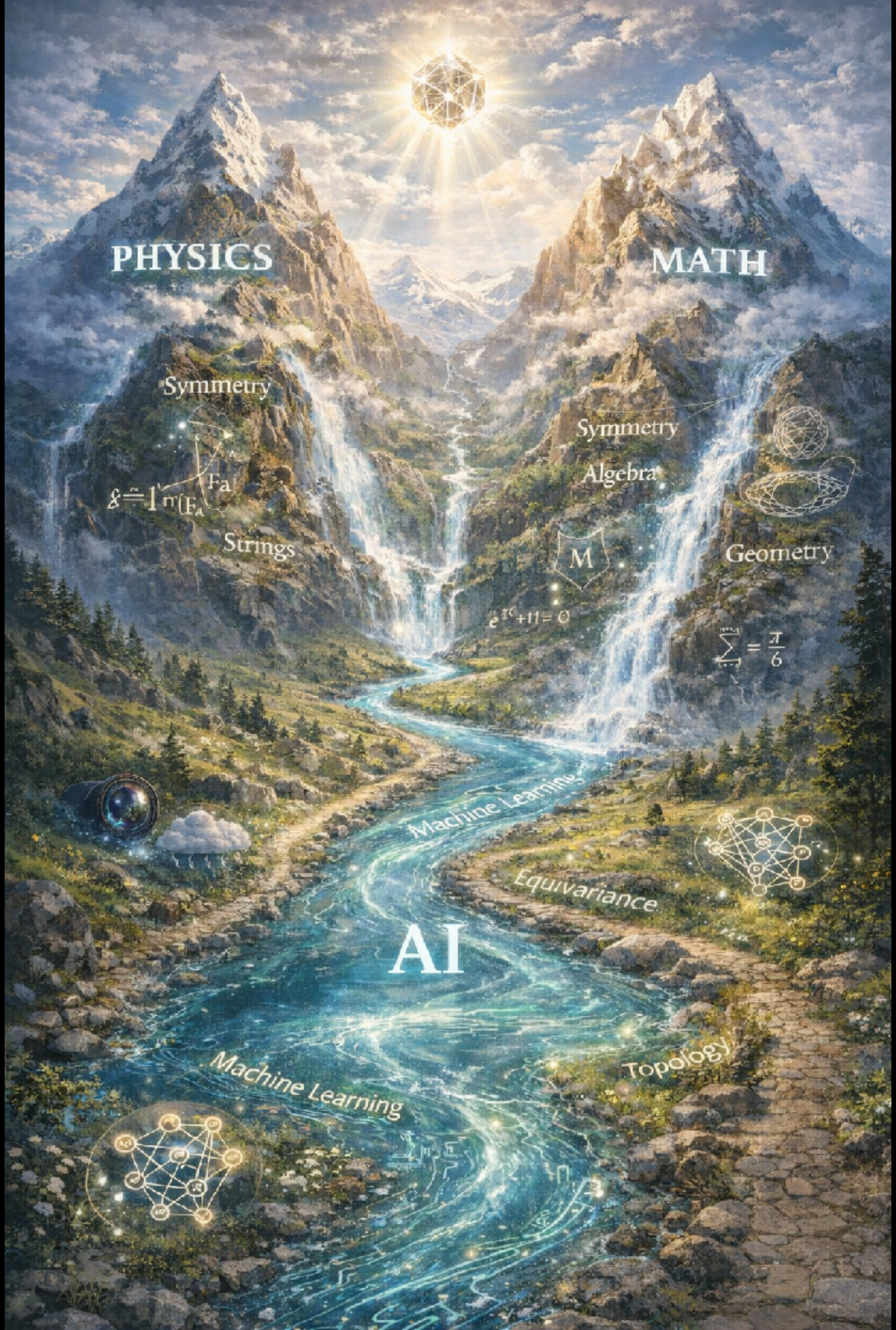
Read More

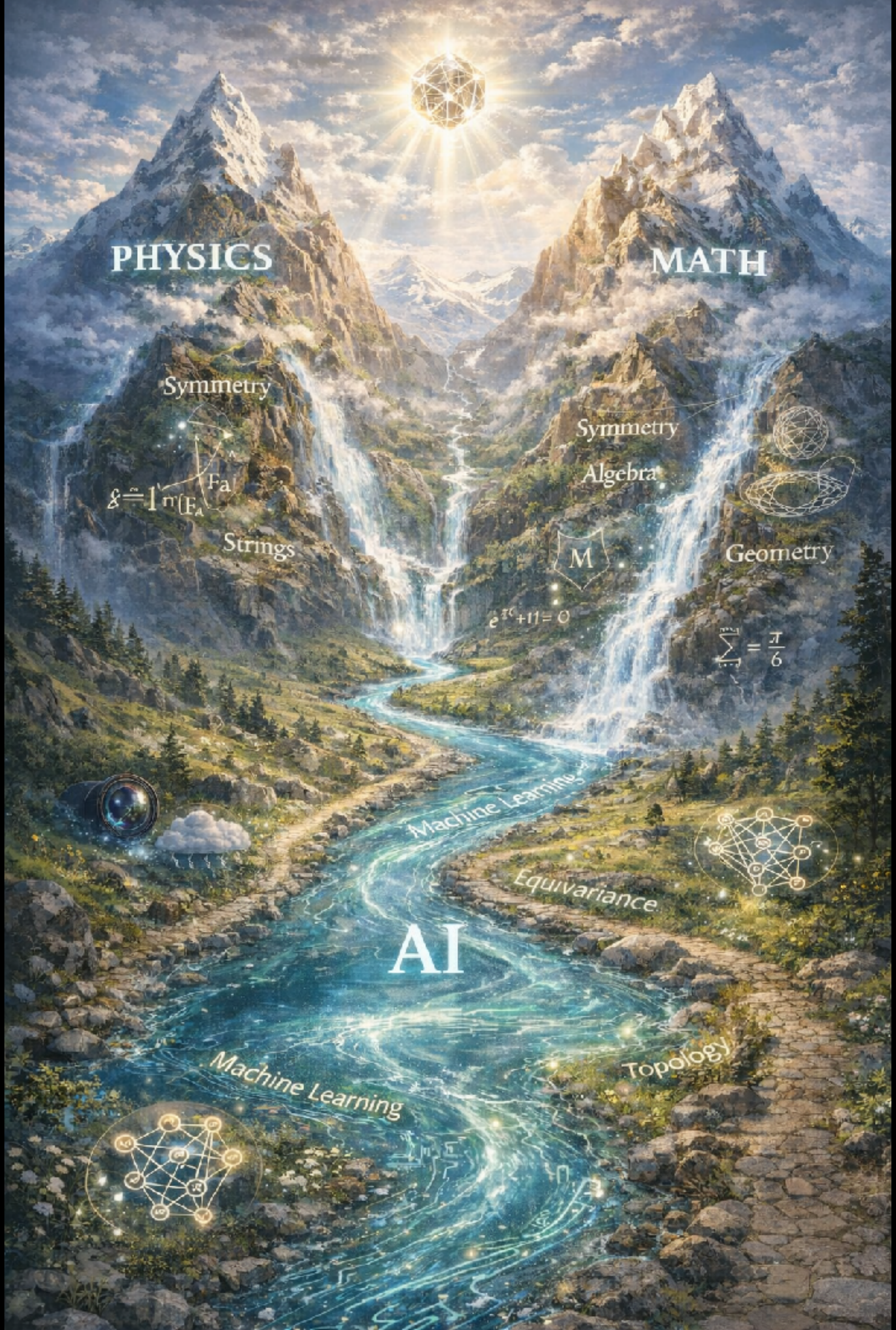
Elias in Boston

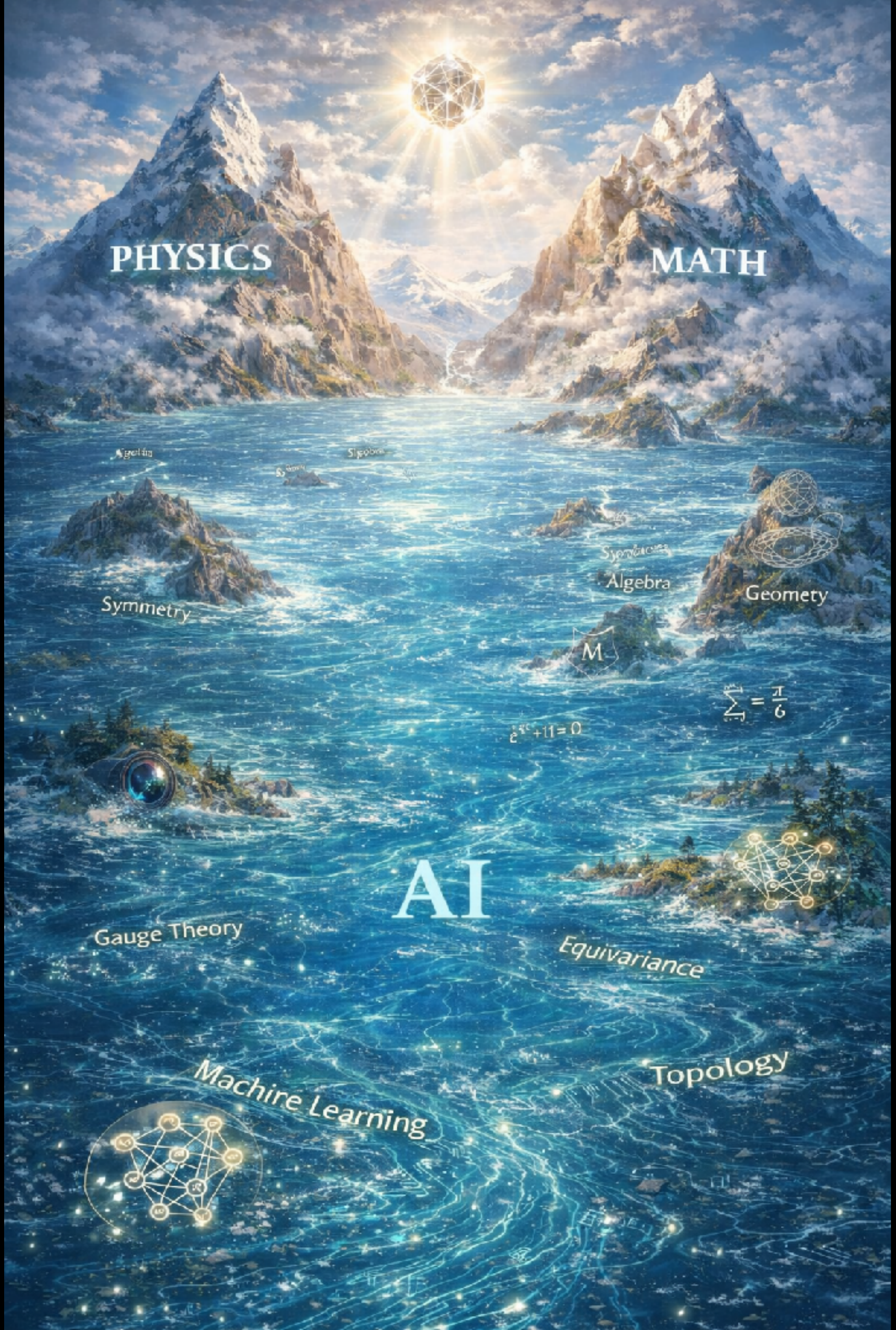
16 Jan 2026

Our PhD student Elias will spend the spring in Boston on a WASP-funded research visit, working with Maurice Weiler at MIT and Robin Walters at

Math/Physics4AI

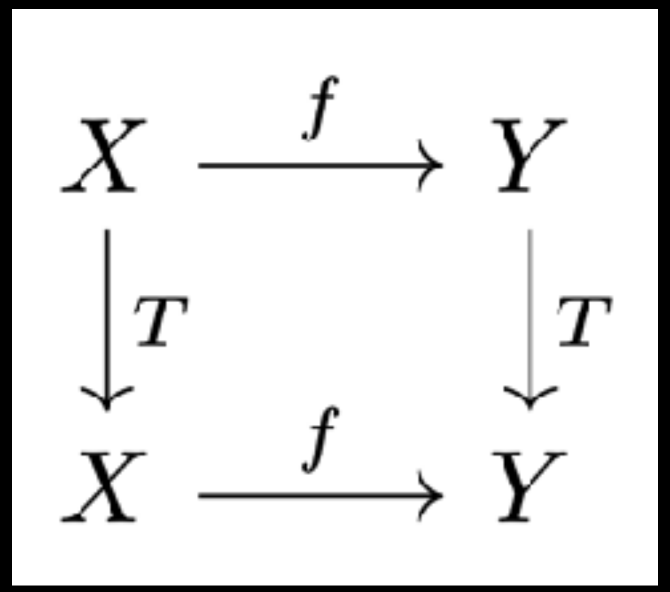



PHYSICS
MATH
Symmetry
Symmetry
Algebra
Strings
M
Geometry
Machine Learning
Equivariance
AI
Topology
Machine Learning


PHYSICS
MATH
Algebra
Symmetry
M
Geometry
AI
Gauge Theory
Equivariance
Machire Learning
Topology

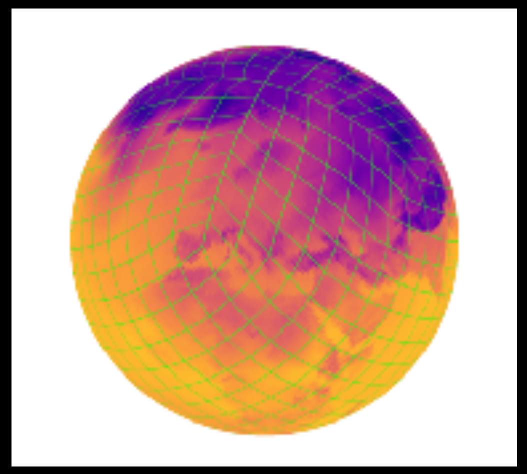
Outline

- Motivation from physics: Unification



- Symmetries in neural networks: Geometric deep learning

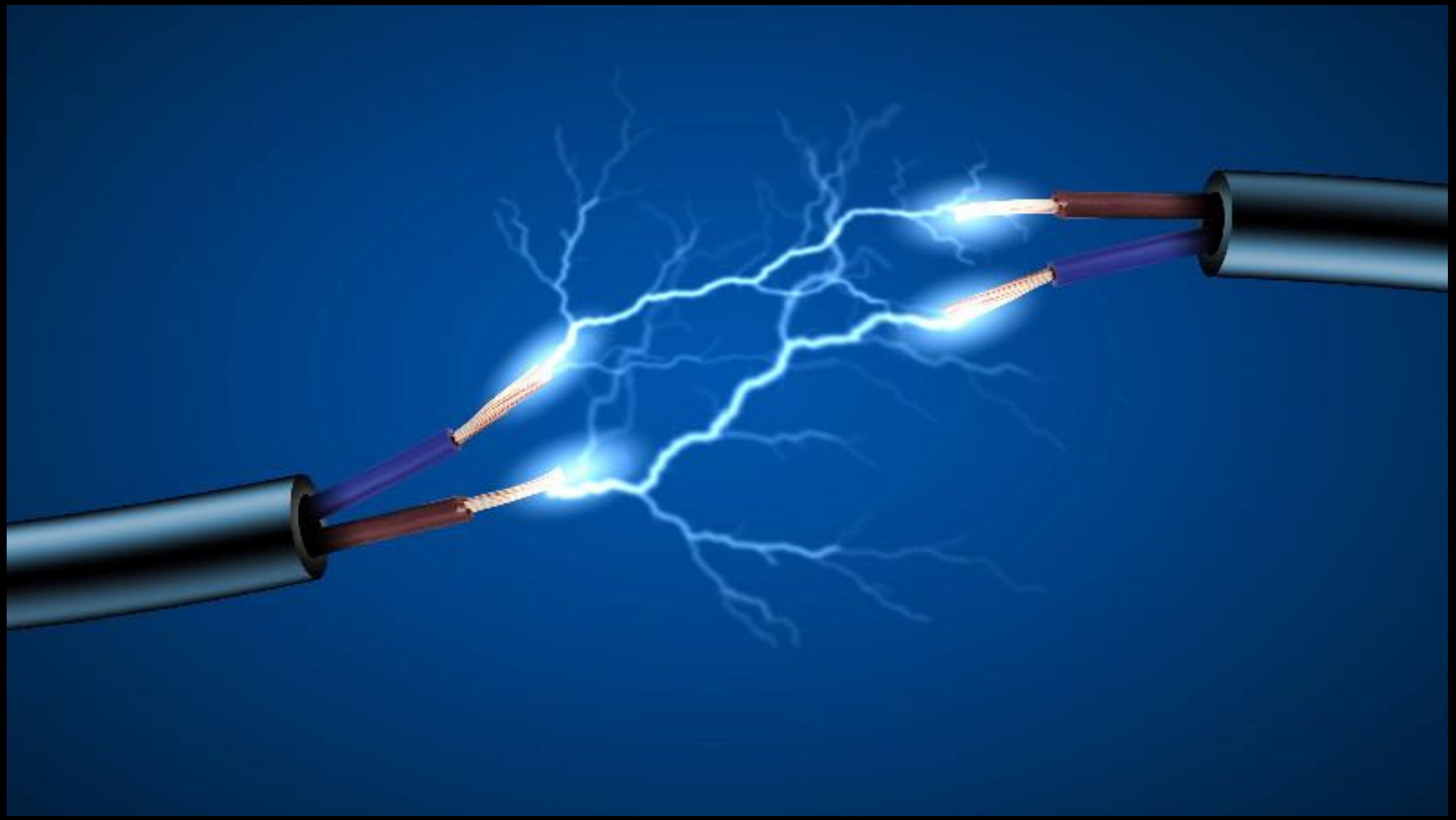
- Applications: AI4Physics



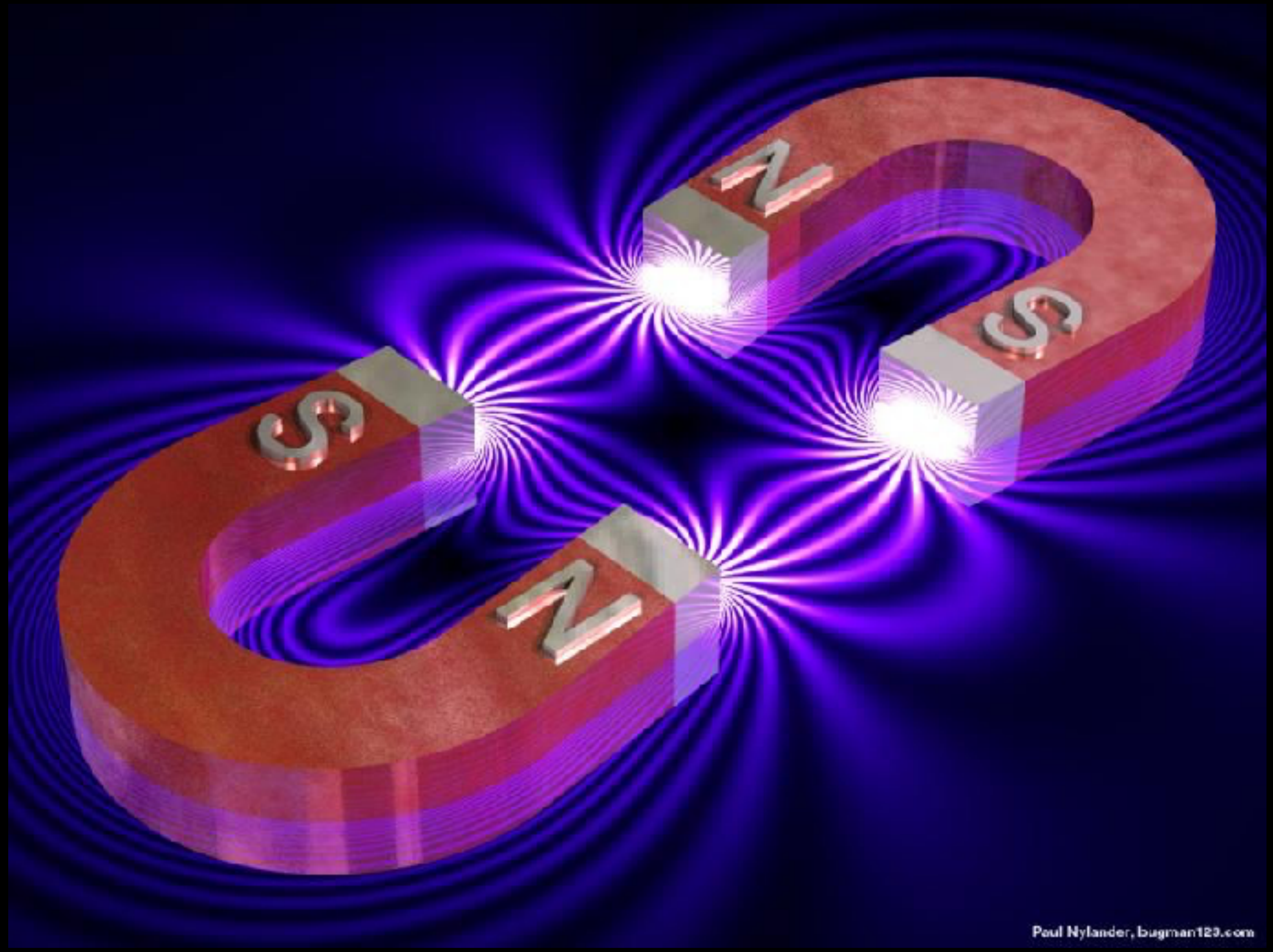
- Conclusions

Motivation from physics: Unification

Unification

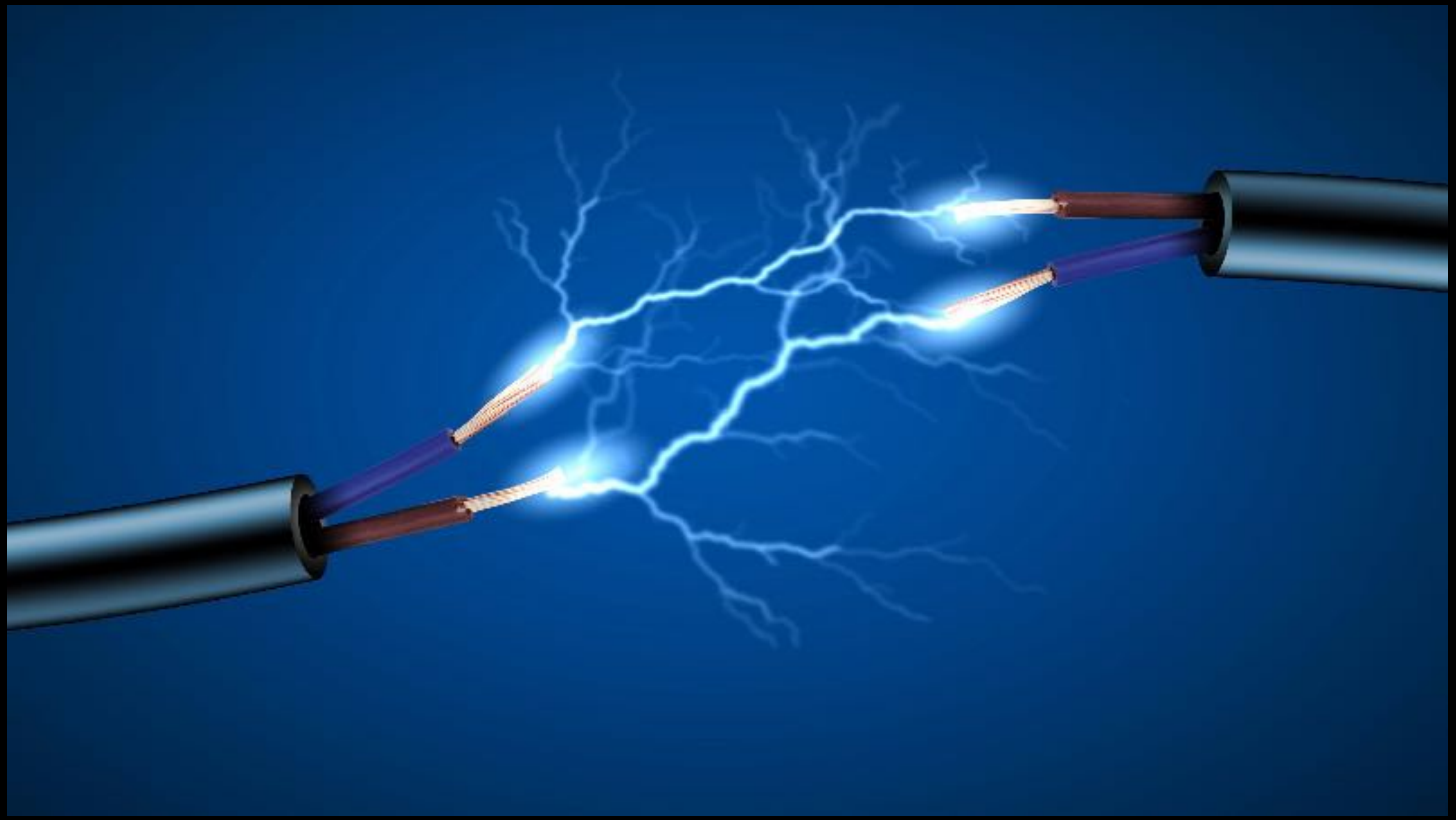


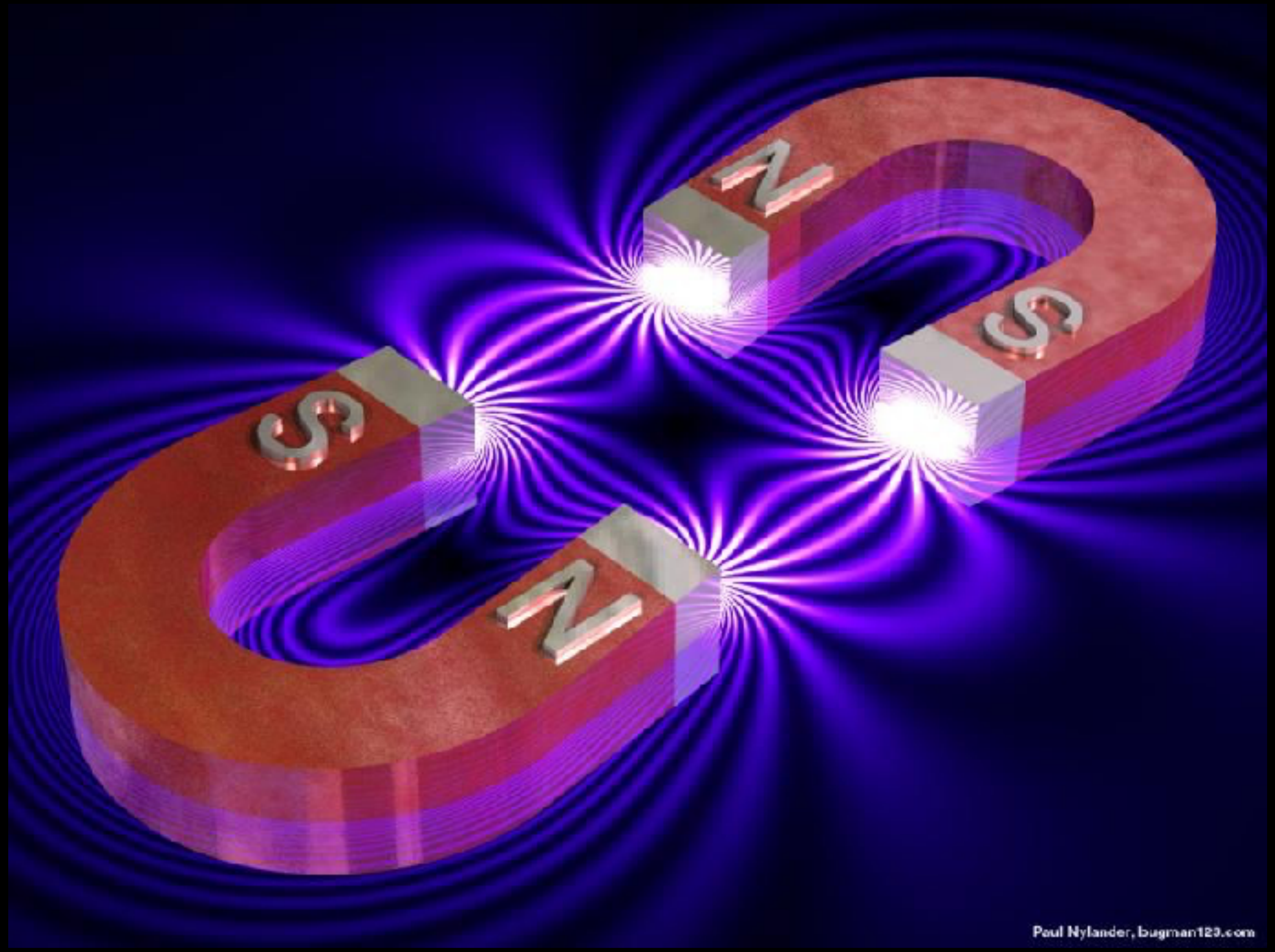
electricity

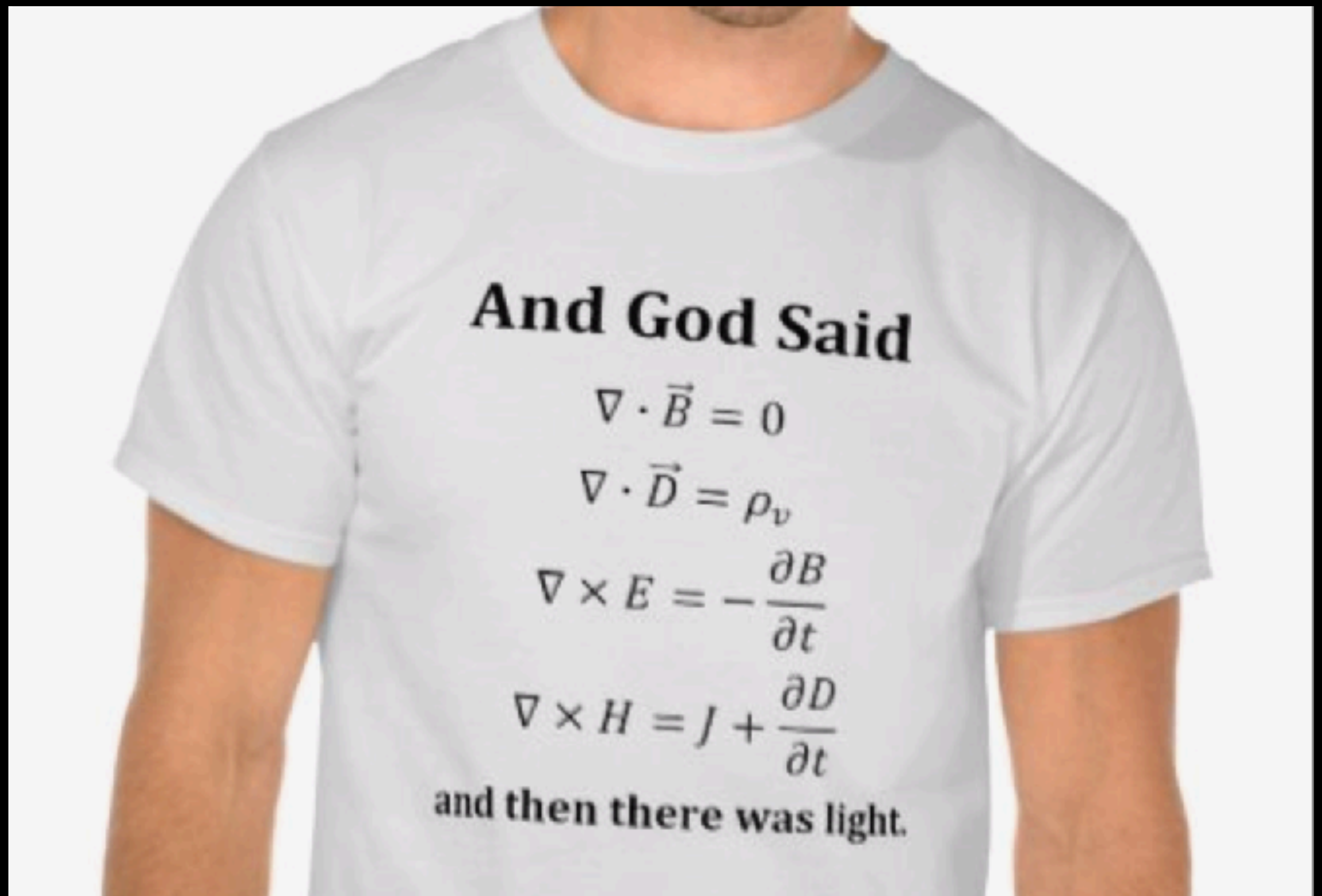


magnetism

Unification





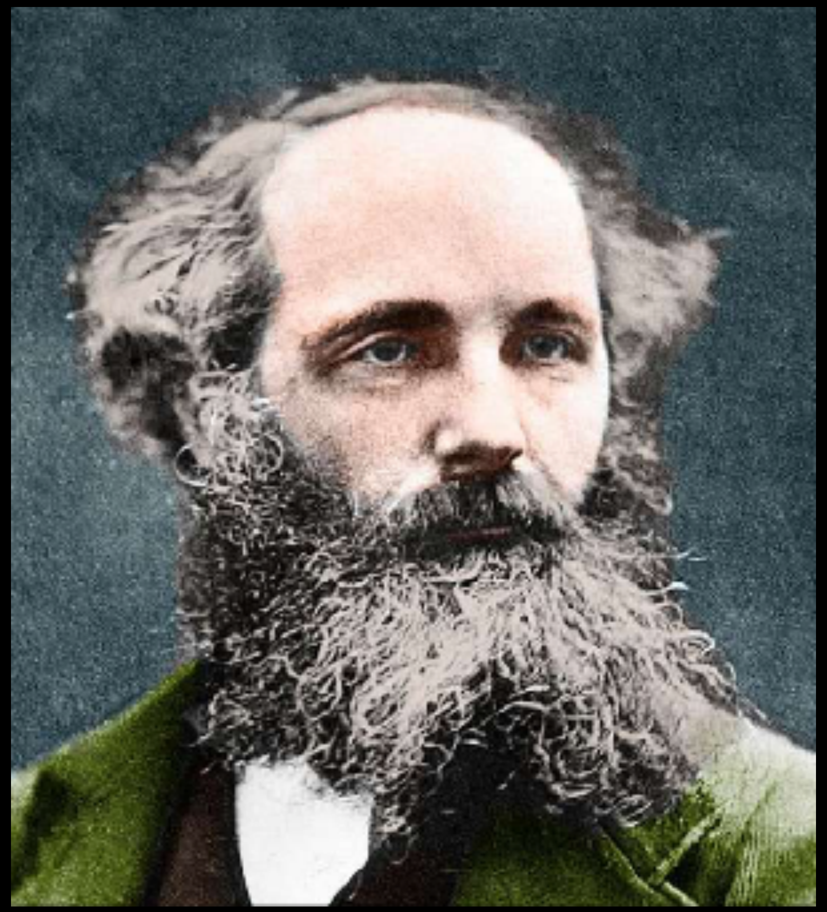


electromagnetism

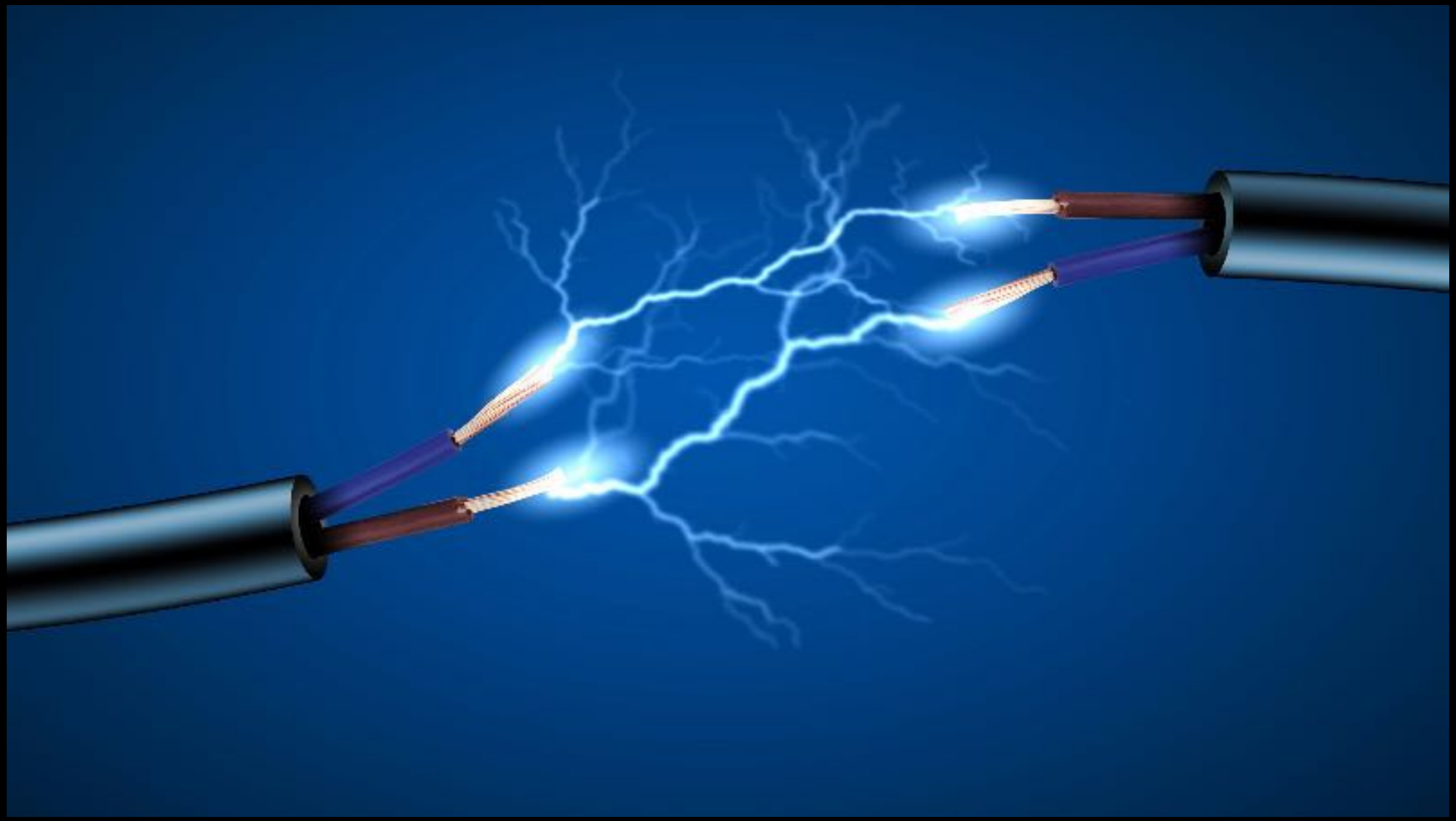
Maxwell's equations (1861)

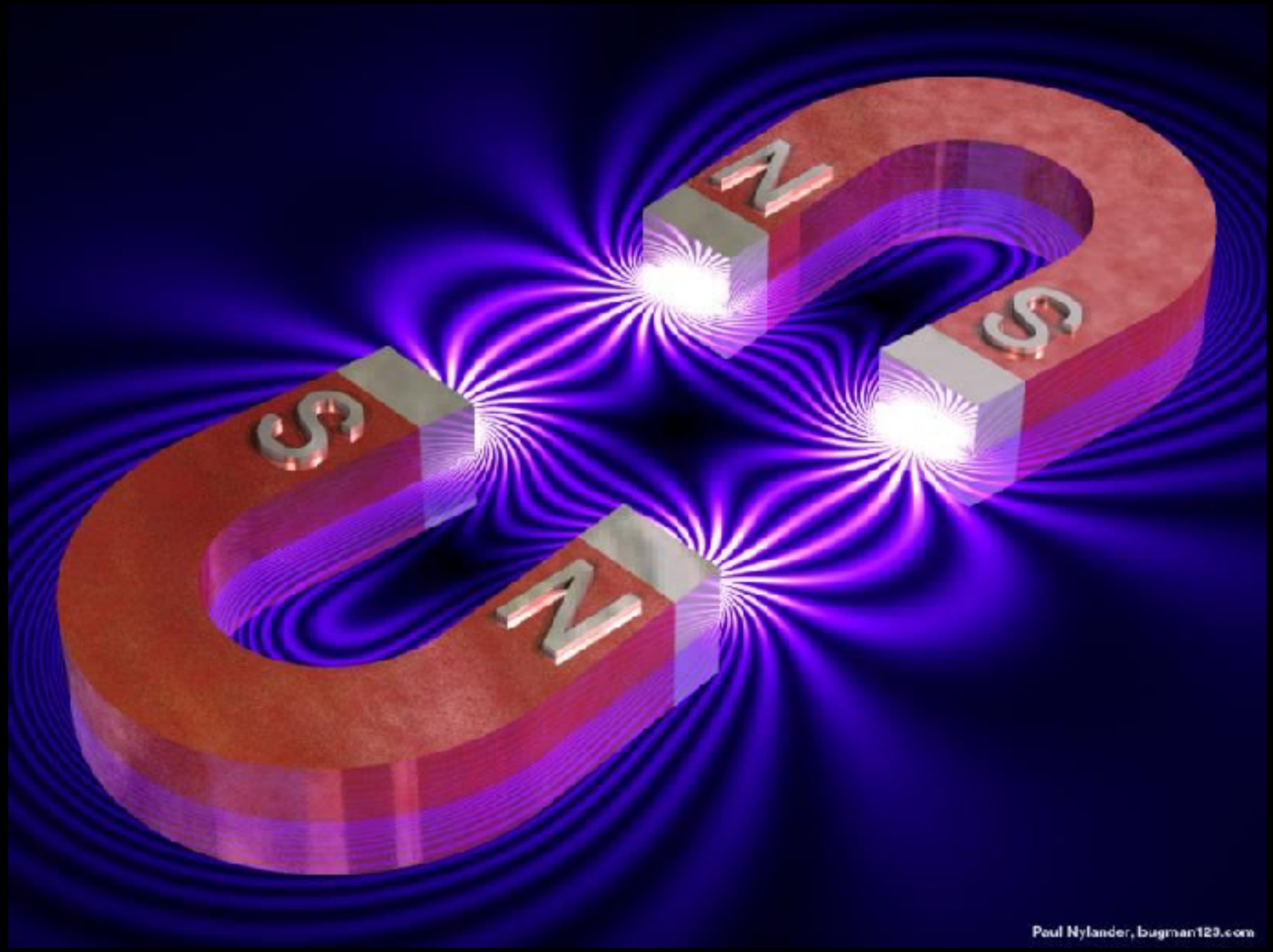
U(1) "gauge theory"

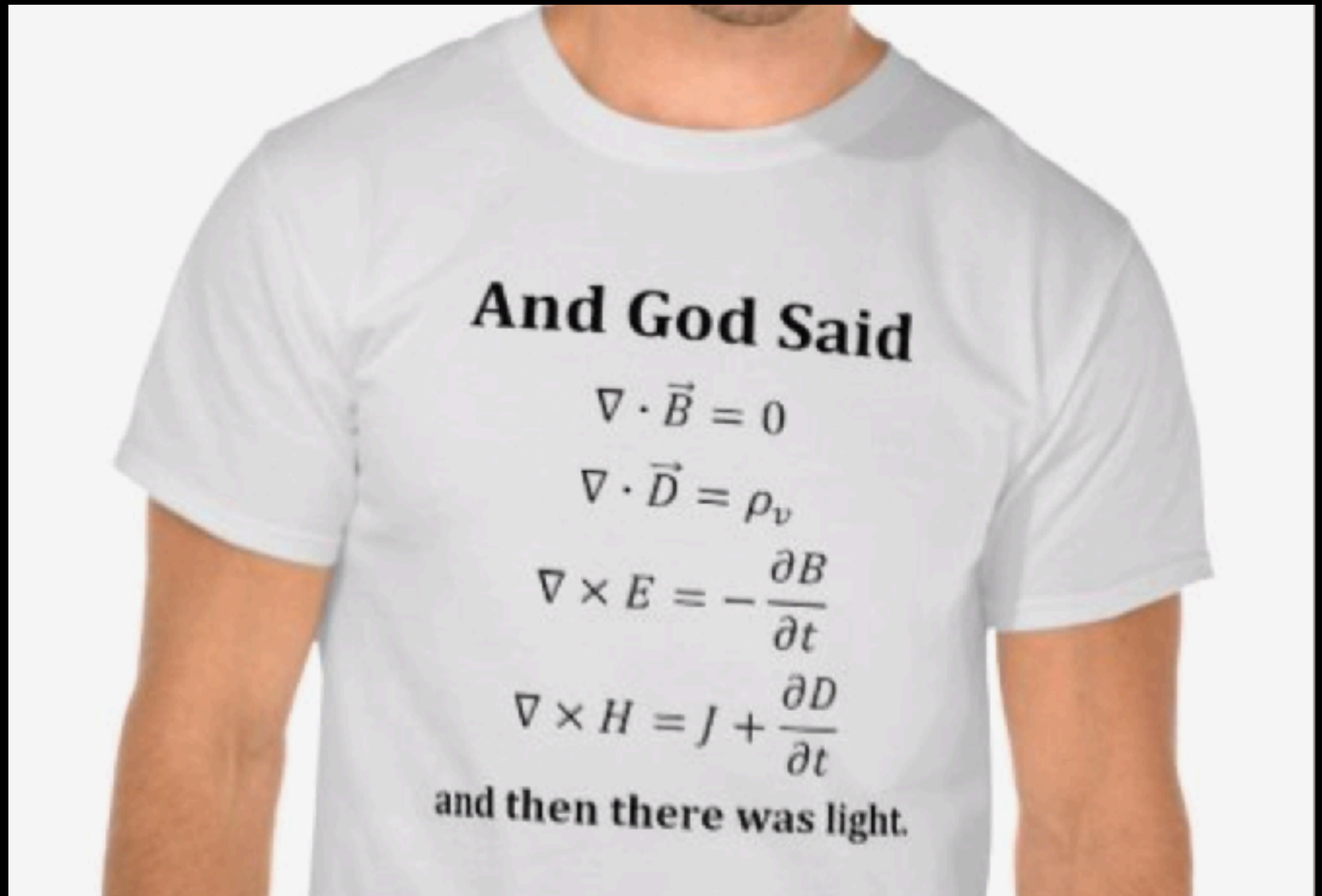
(local symmetry)



Unification





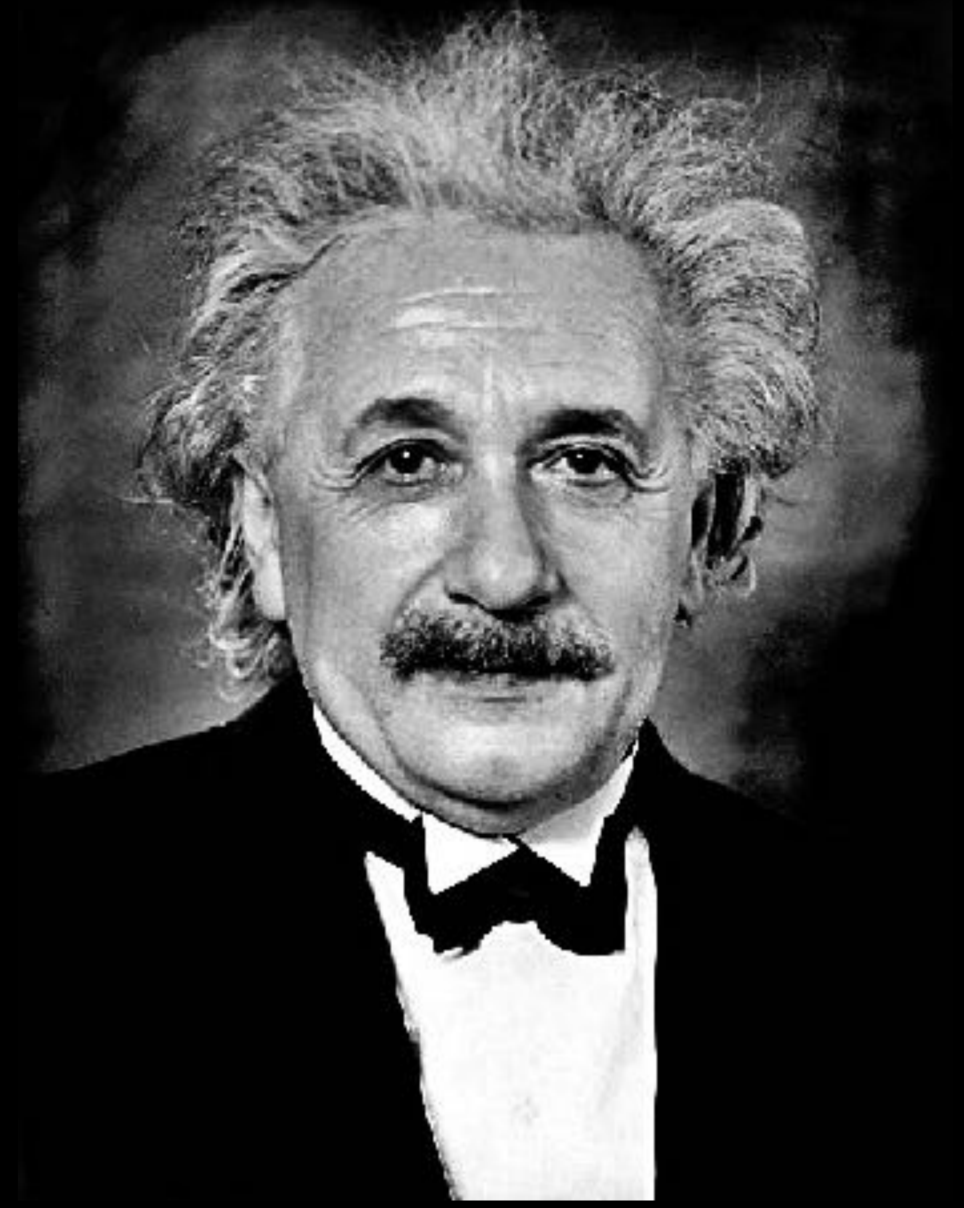


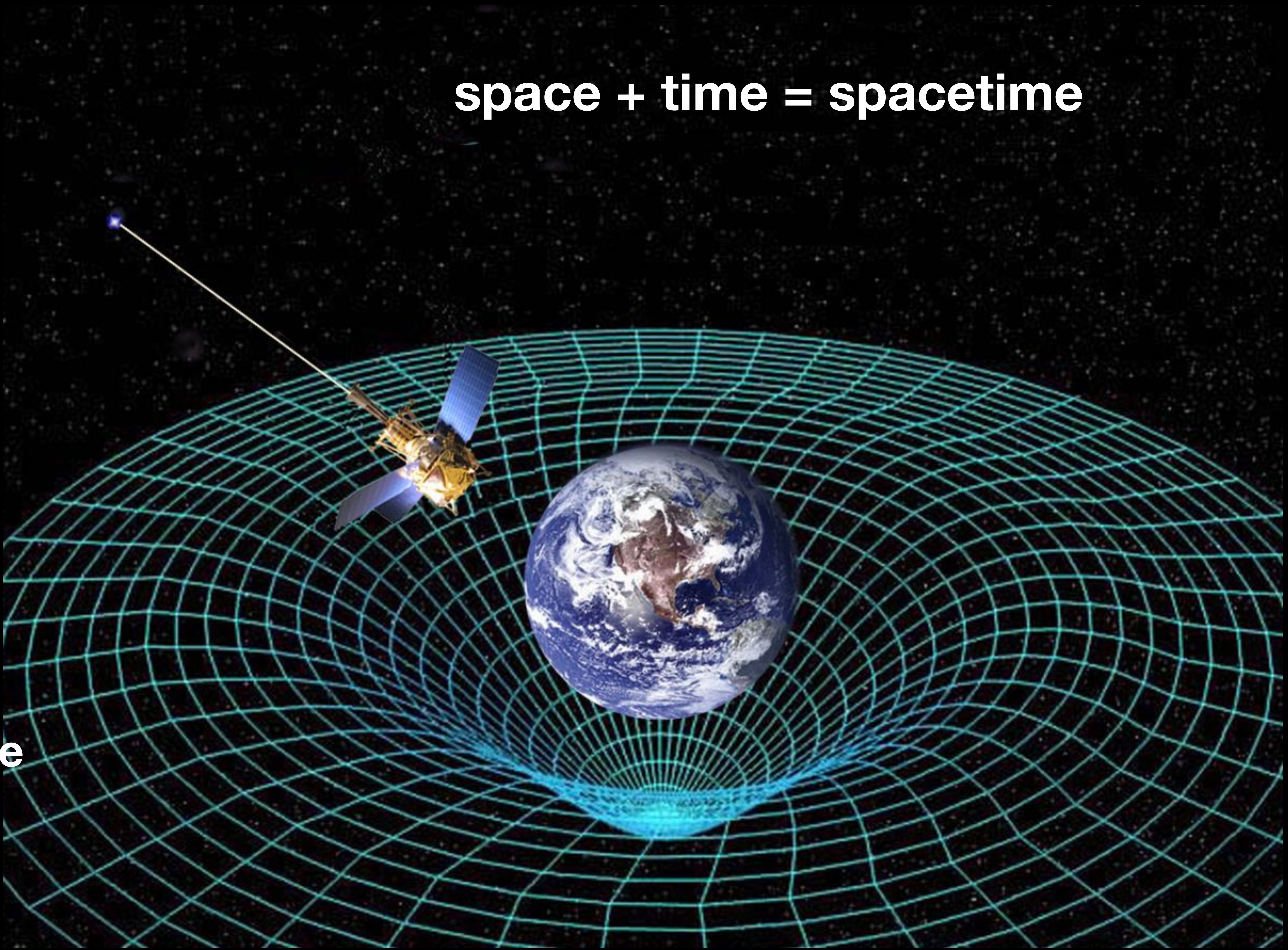
electromagnetism

Electric-magnetic duality

E ⟷ B

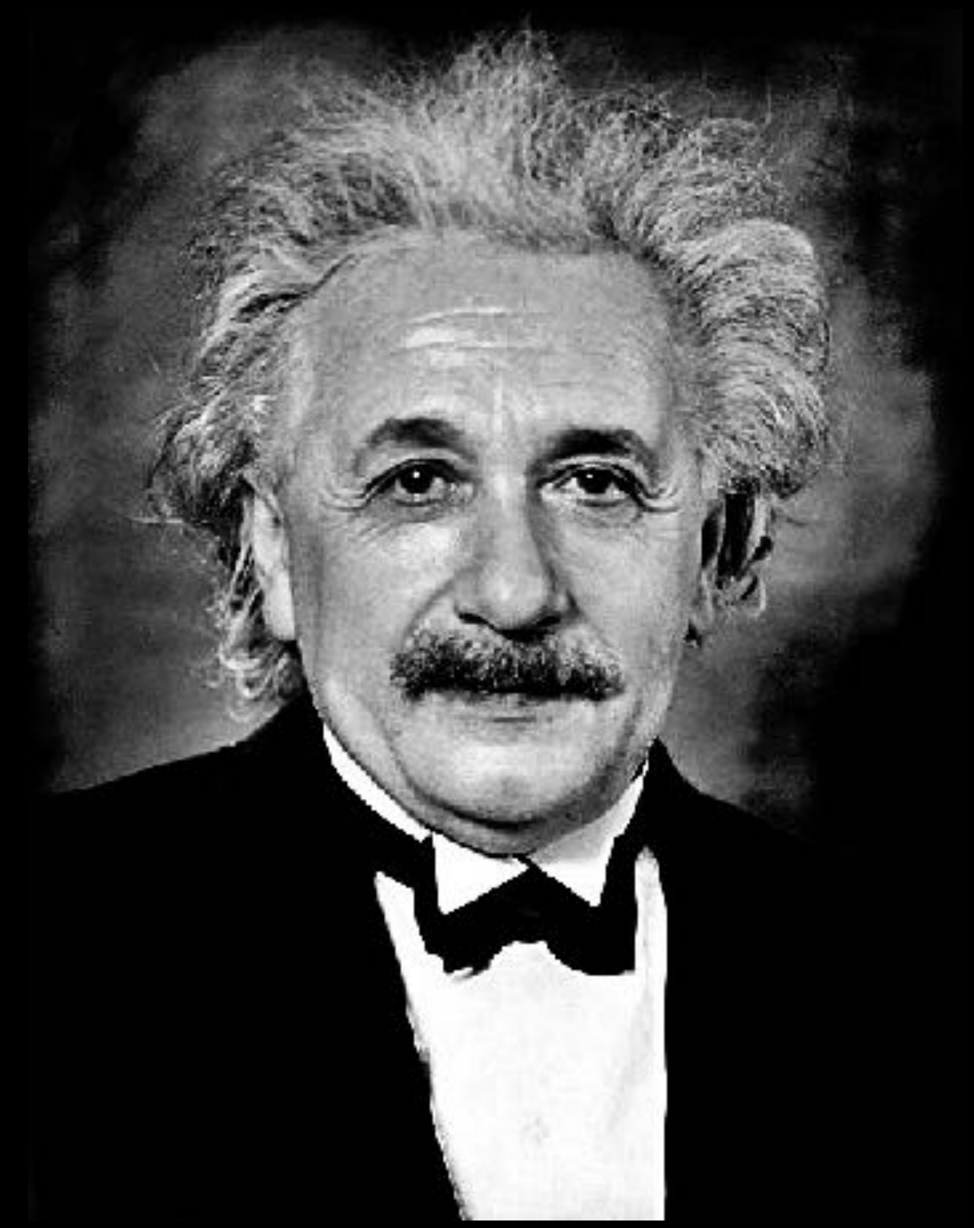
(global symmetry)





gravity:

mass curves spacetime

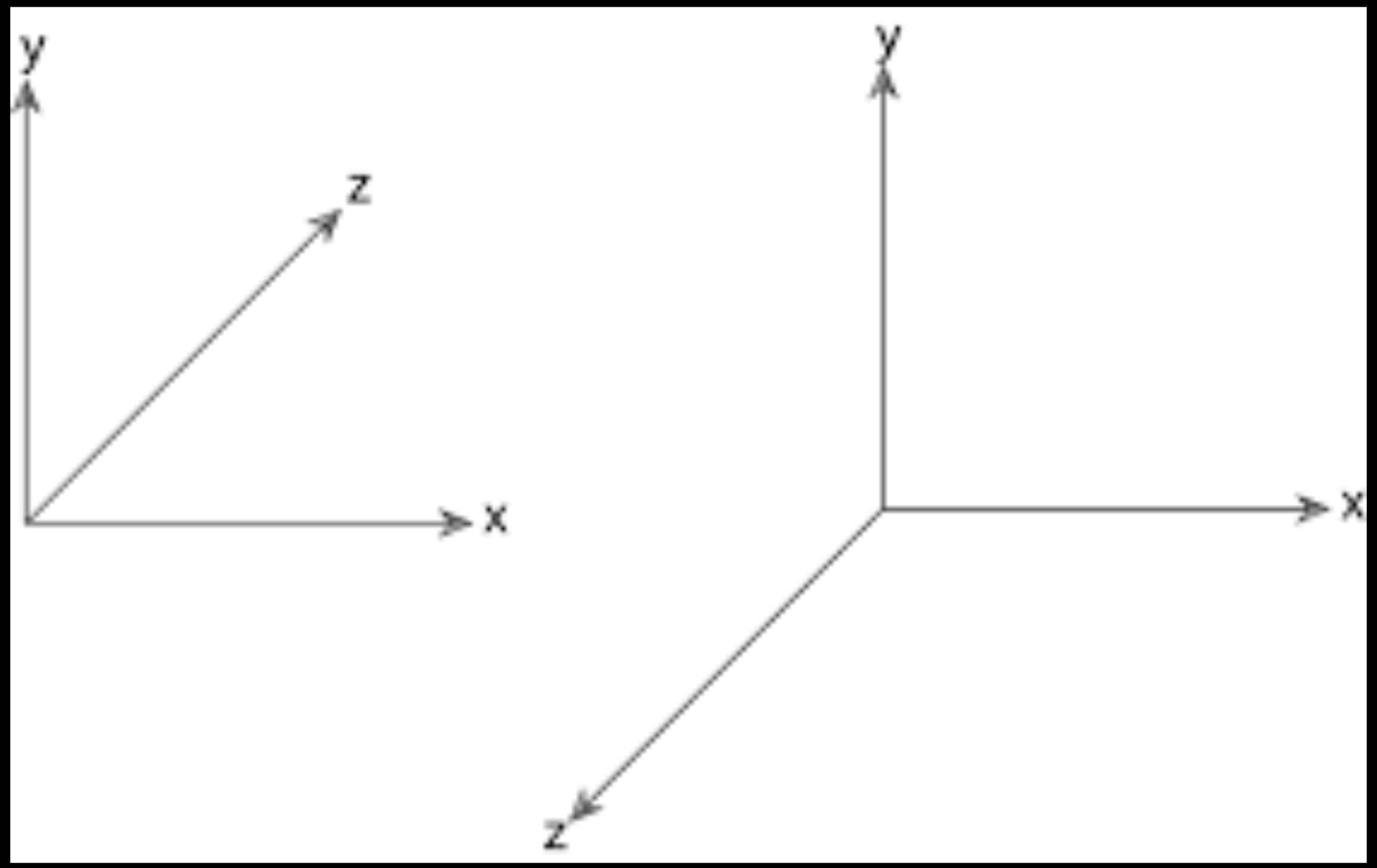


Einstein's general theory of relativity connected differential geometry and physics

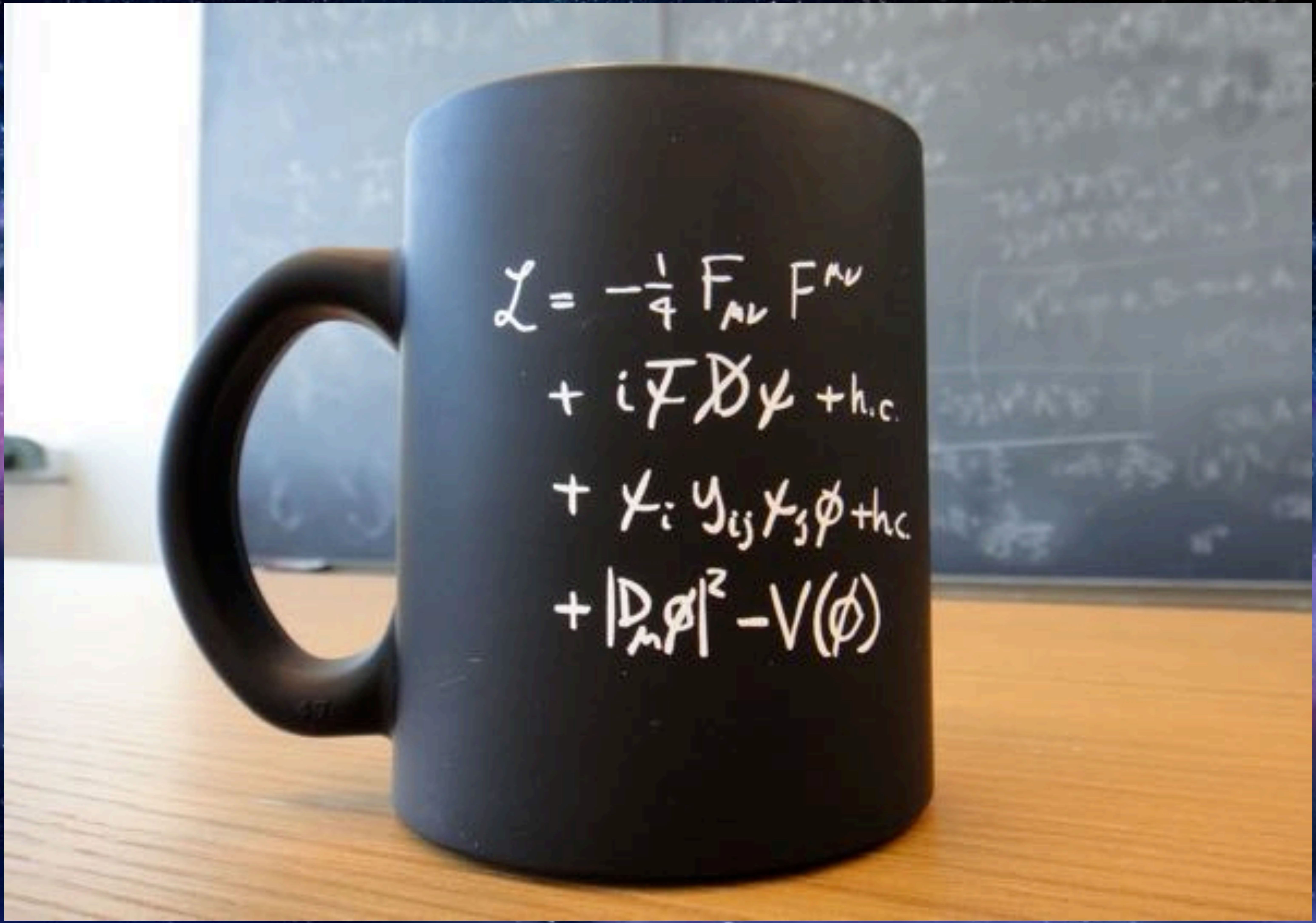
Principle of general covariance:

The laws of physics should take the same form independently of which coordinate system we use to represent them

$$R_{\mu\nu}(g) - \frac{1}{2}g_{\mu\nu}R(g) = T_{\mu\nu}(g, F, \ldots)$$




The fundamental laws of Nature are determined by symmetries!

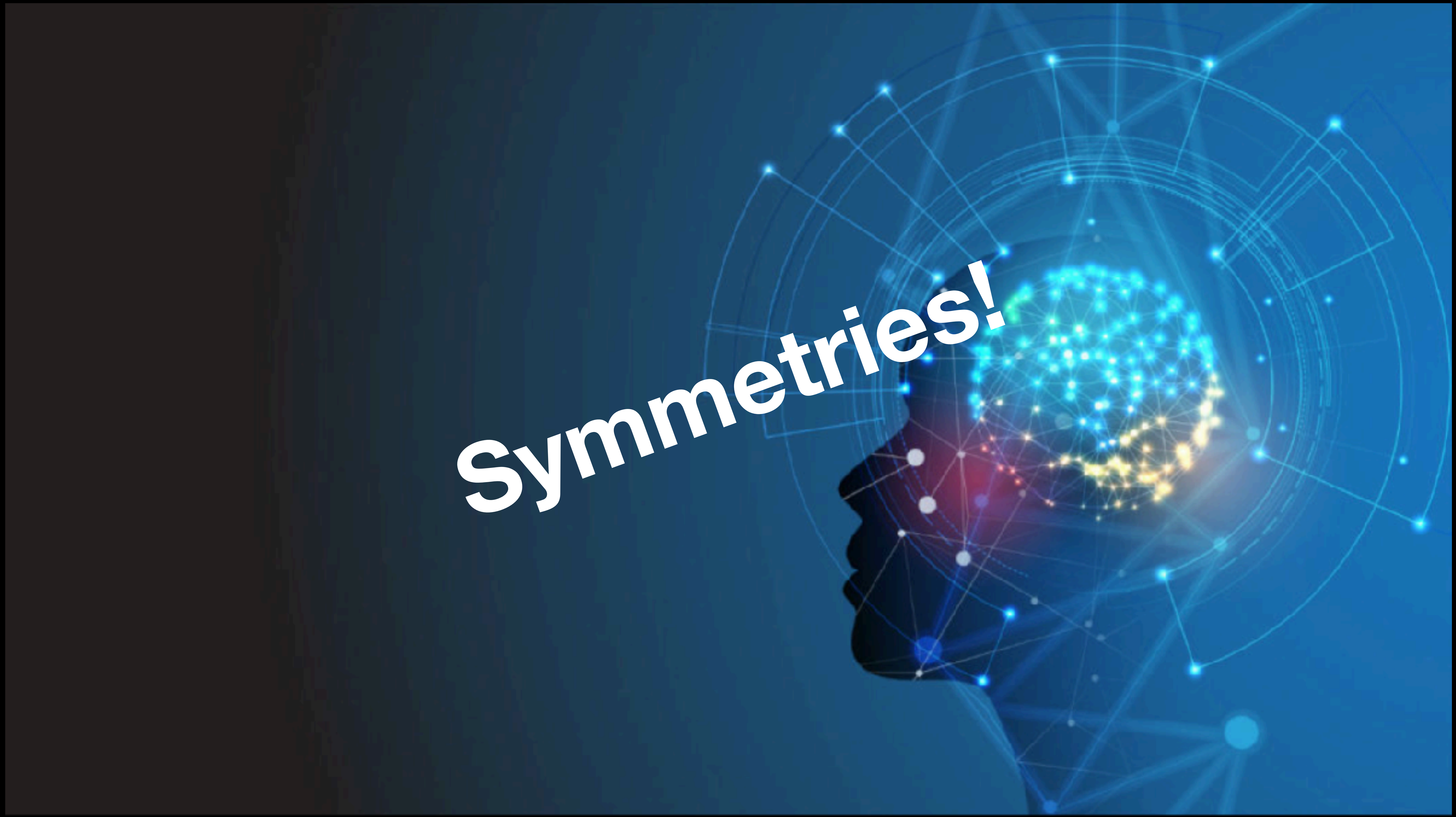


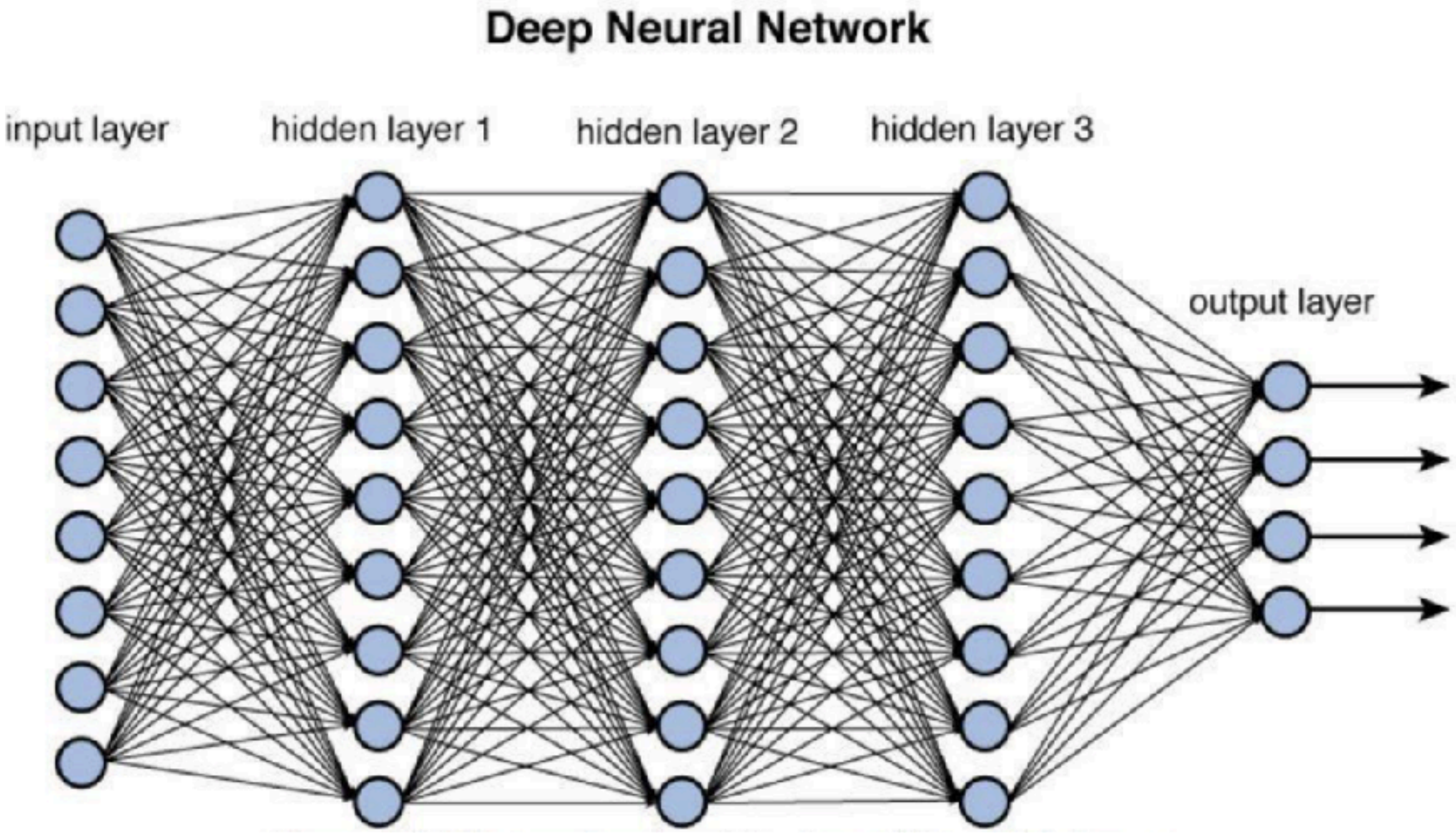
The fundamental laws of Nature are determined by symmetries!

Symmetries in neural networks: Geometric deep learning

What does all this have to do with AI?

What does all this have to do with AI?





Artificial neuron:

1. Input layer
2. Hidden layers
3. Output layer

Inductive bias

Incorporate known constraints into the architecture/training

symmetries

differential equations

geometry

Image classification

“skateboard”

“ribs”

“boxing gloves”

Reflection symmetry

reflect

Reflection symmetry

reflect

classify

classify

“boxing glove”

Reflection symmetry

reflect

classify

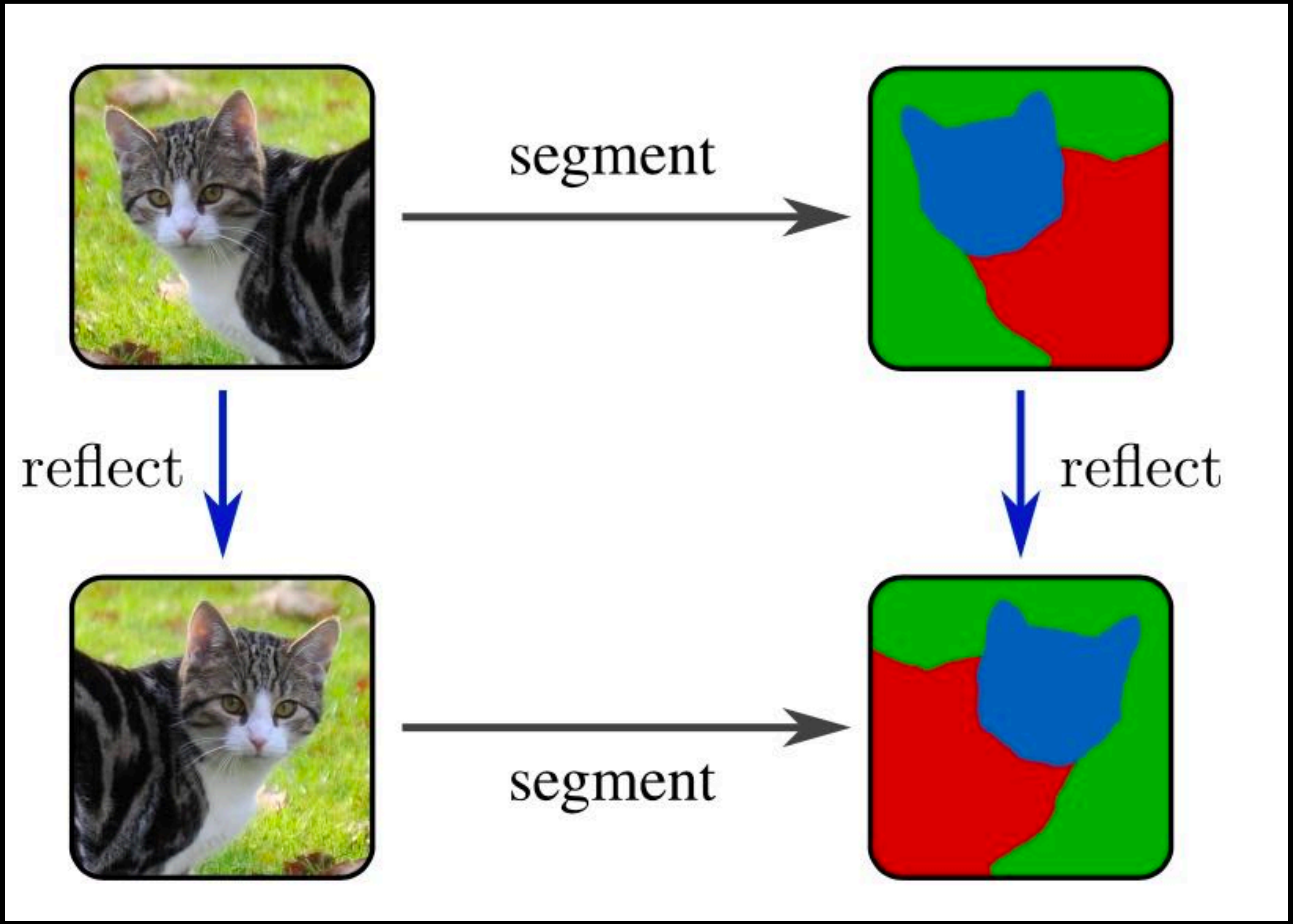
classify

“boxing glove”

Invariance:

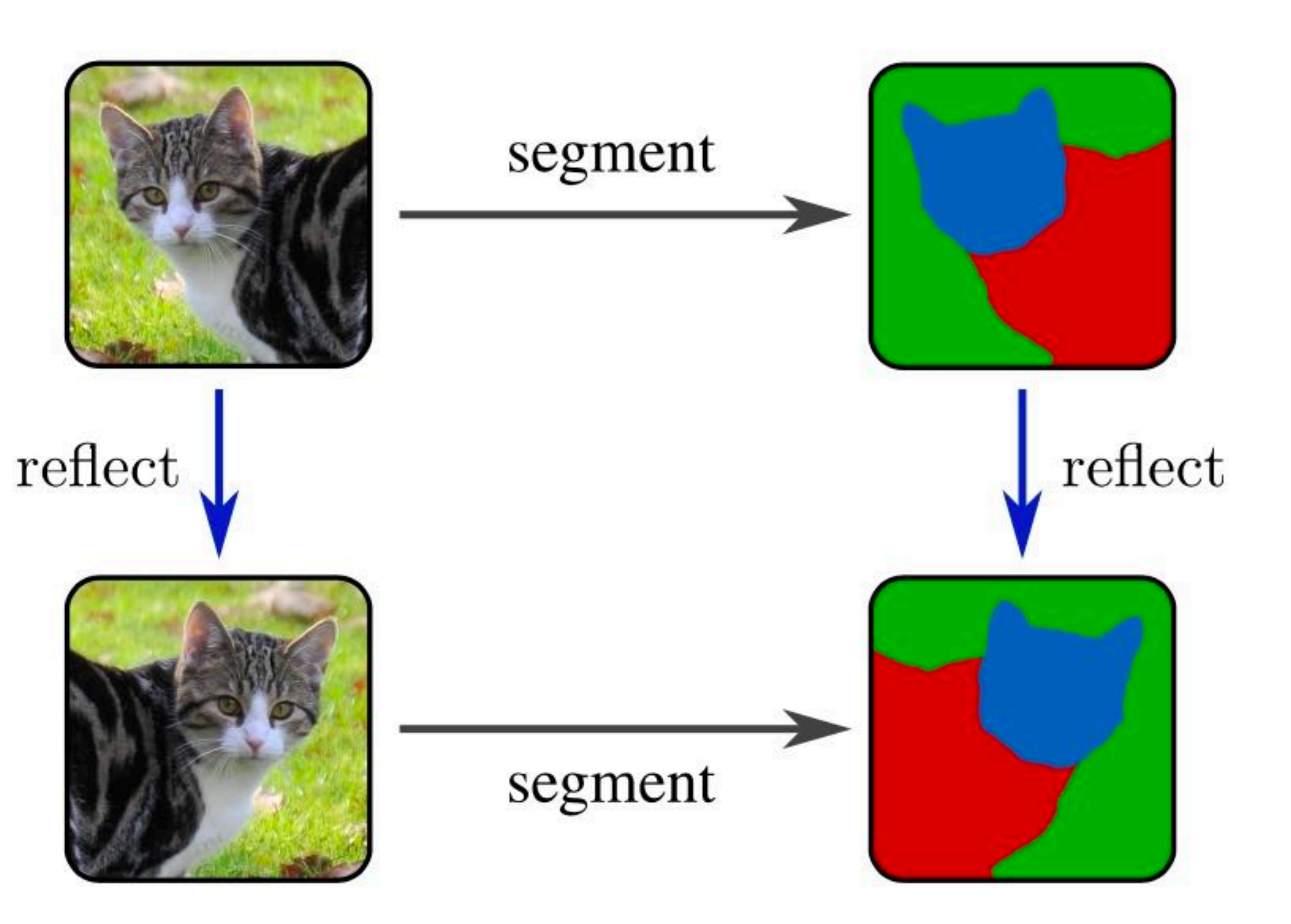
The output is ***unchanged*** when we change the input

Segmentation



[Image from: Weiler, Forré, Verlinde, Welling (2023)]

Segmentation



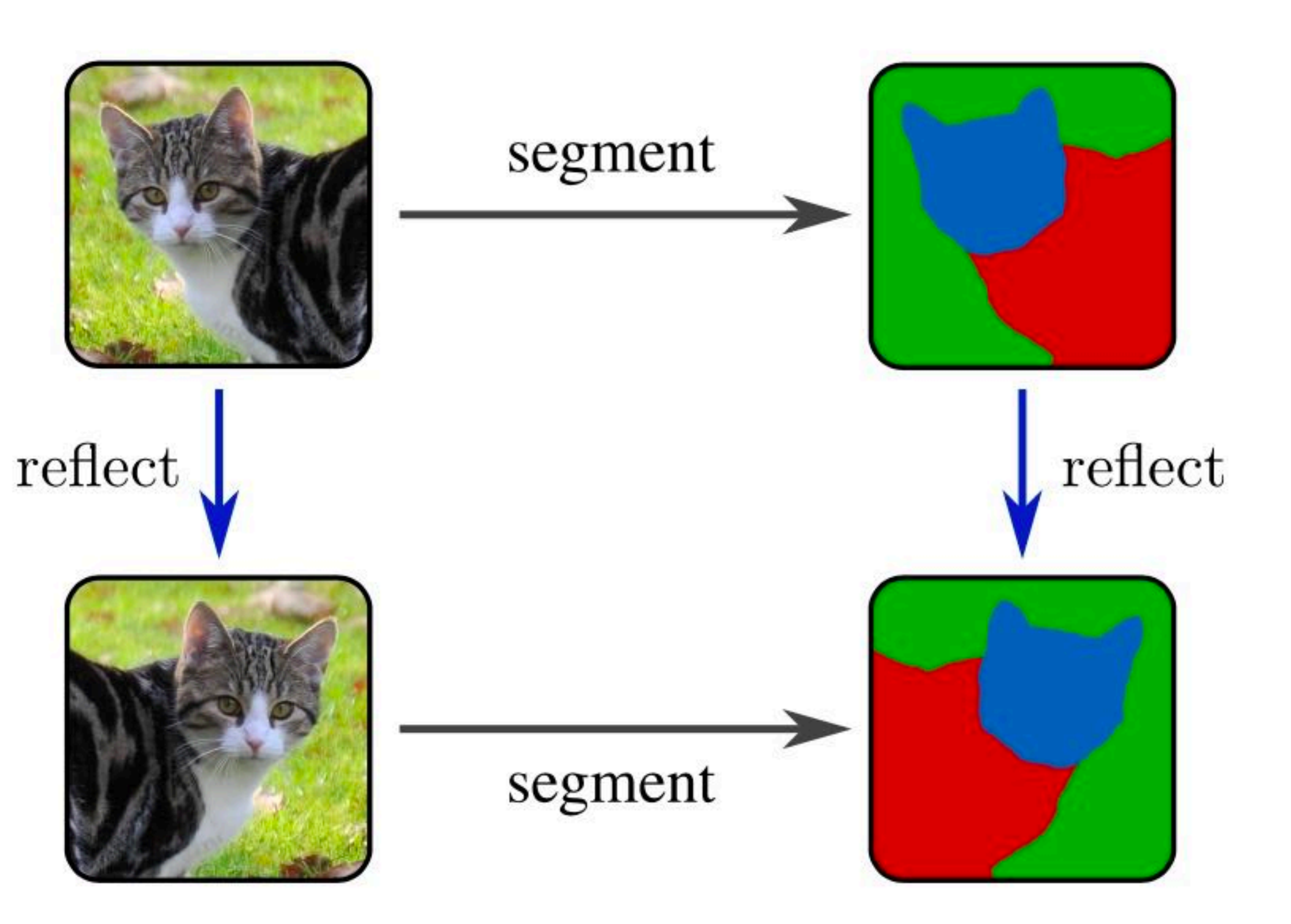
Equivariance:

The output *transforms*

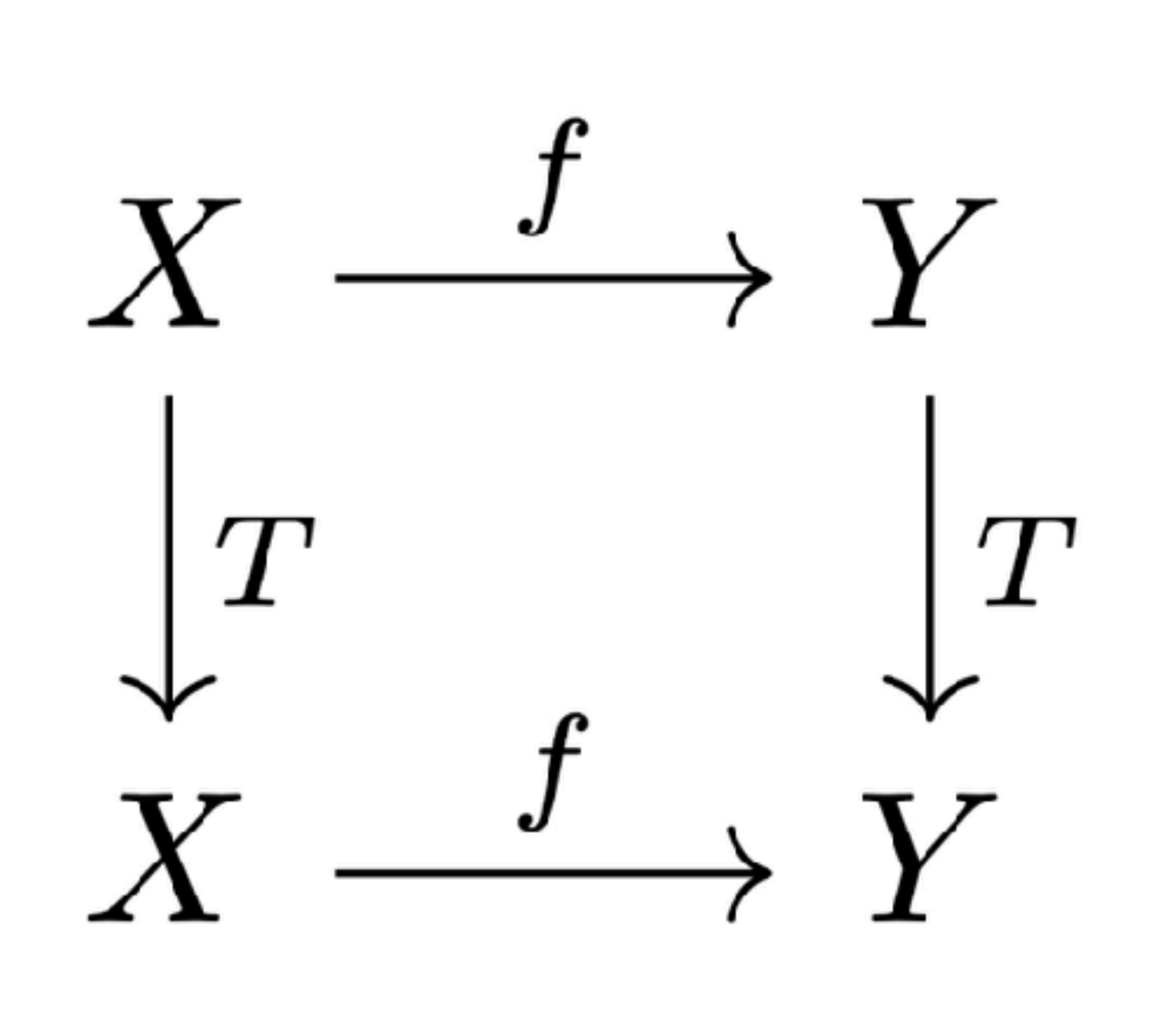
according to the

transformation of the input

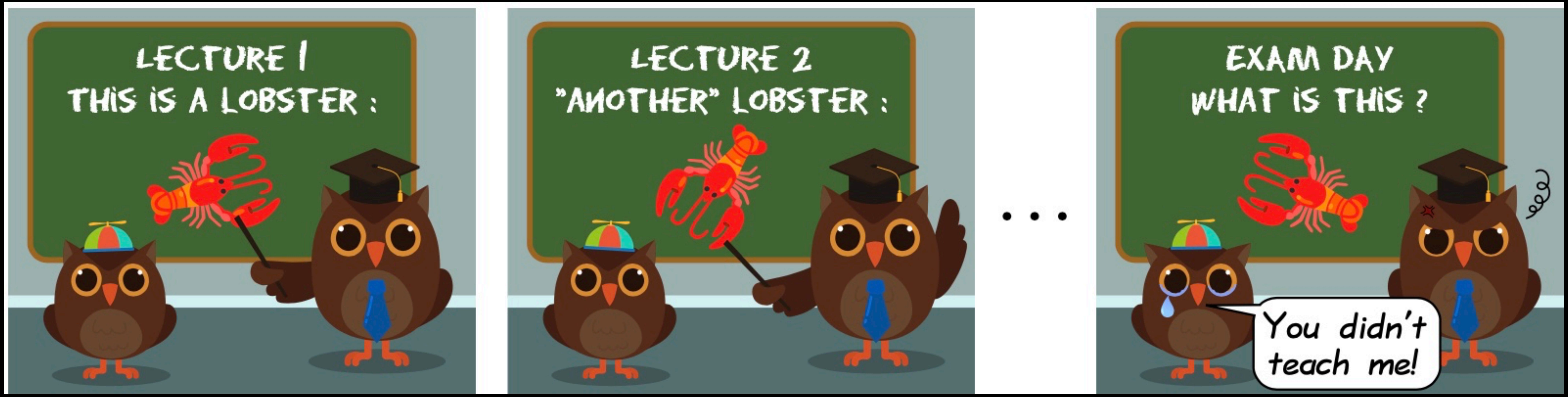
Segmentation



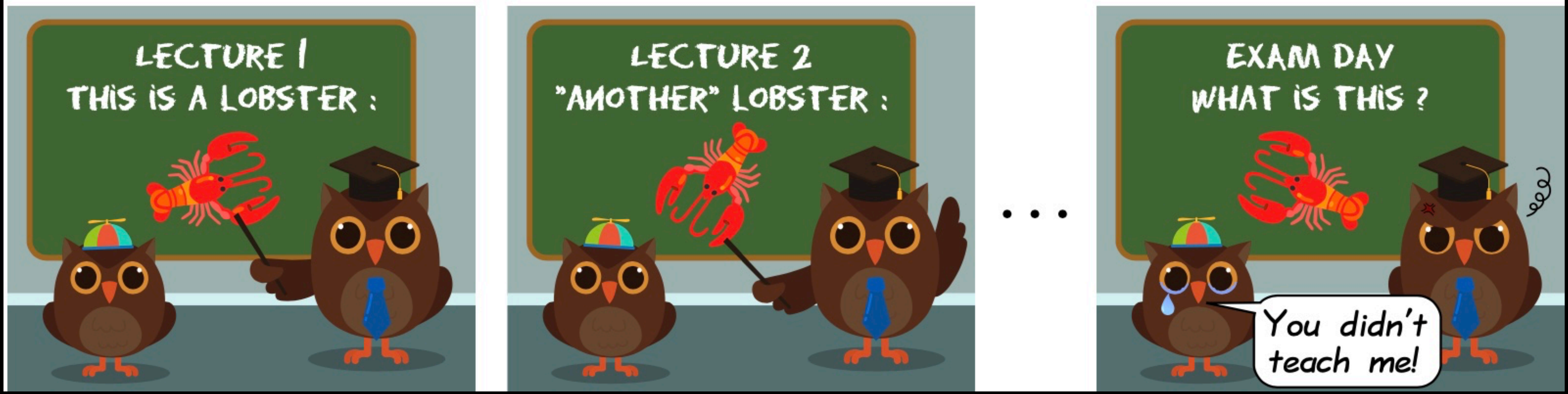
commutative
diagram

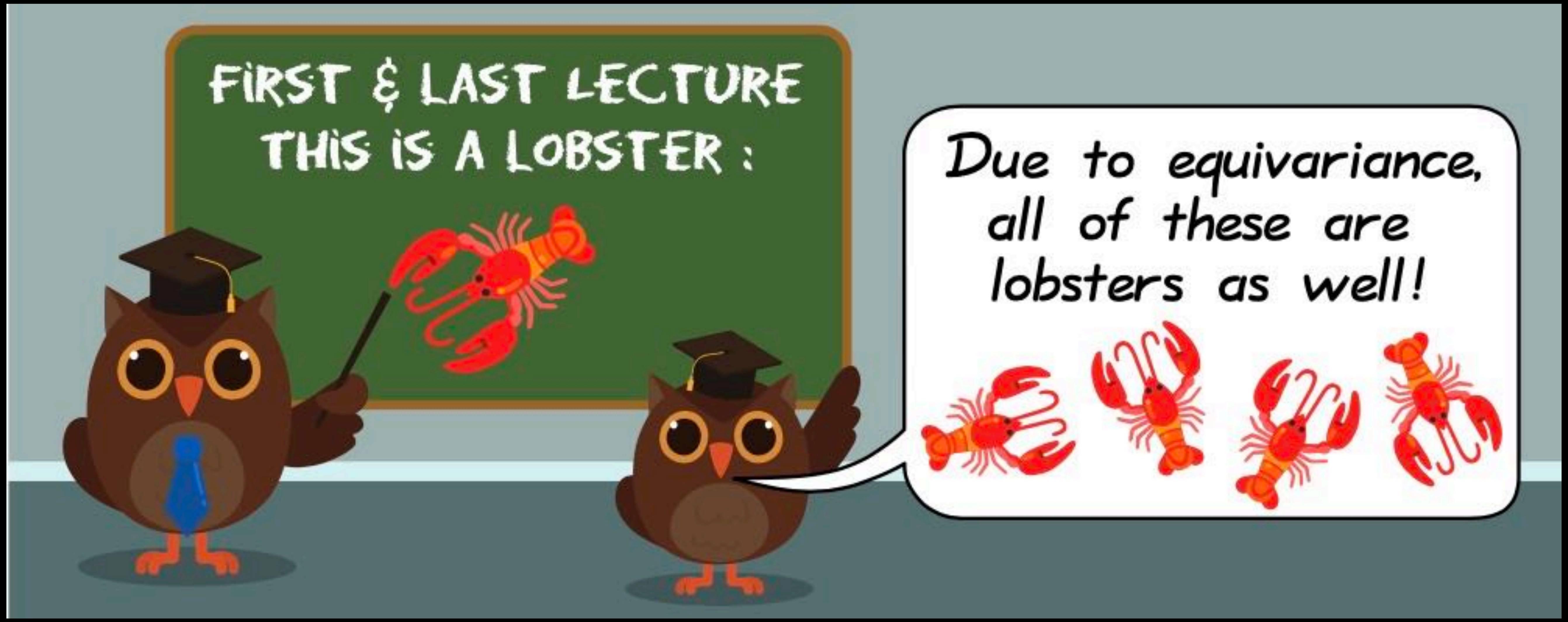


$$f(Tx) = Tf(x)$$

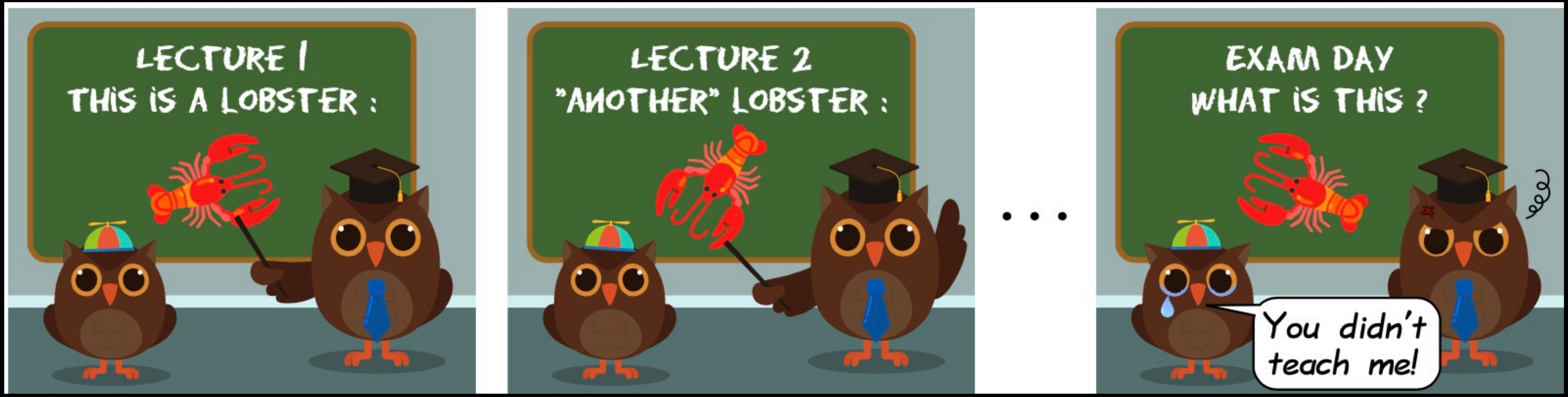


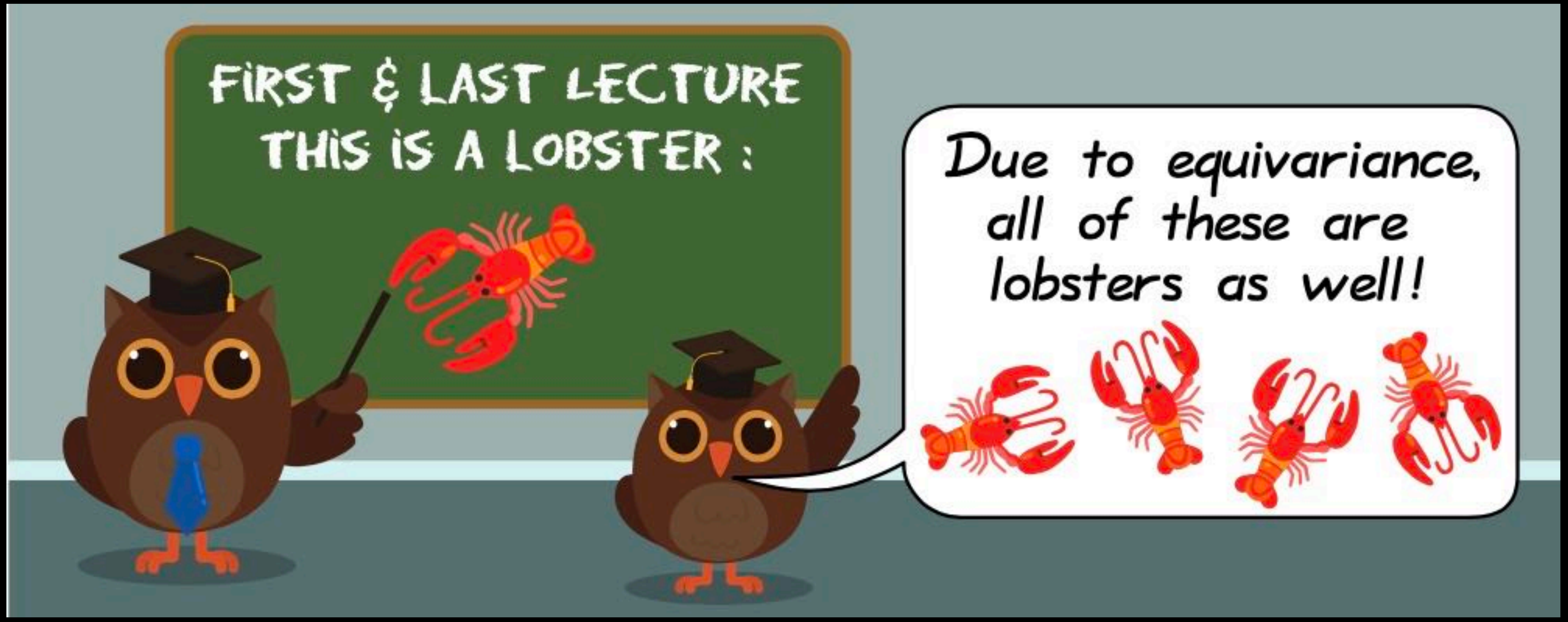
[Images from: Weiler, Forré, Verlinde, Welling (2023)]





[Images from: Weiler, Forré, Verlinde, Welling (2023)]



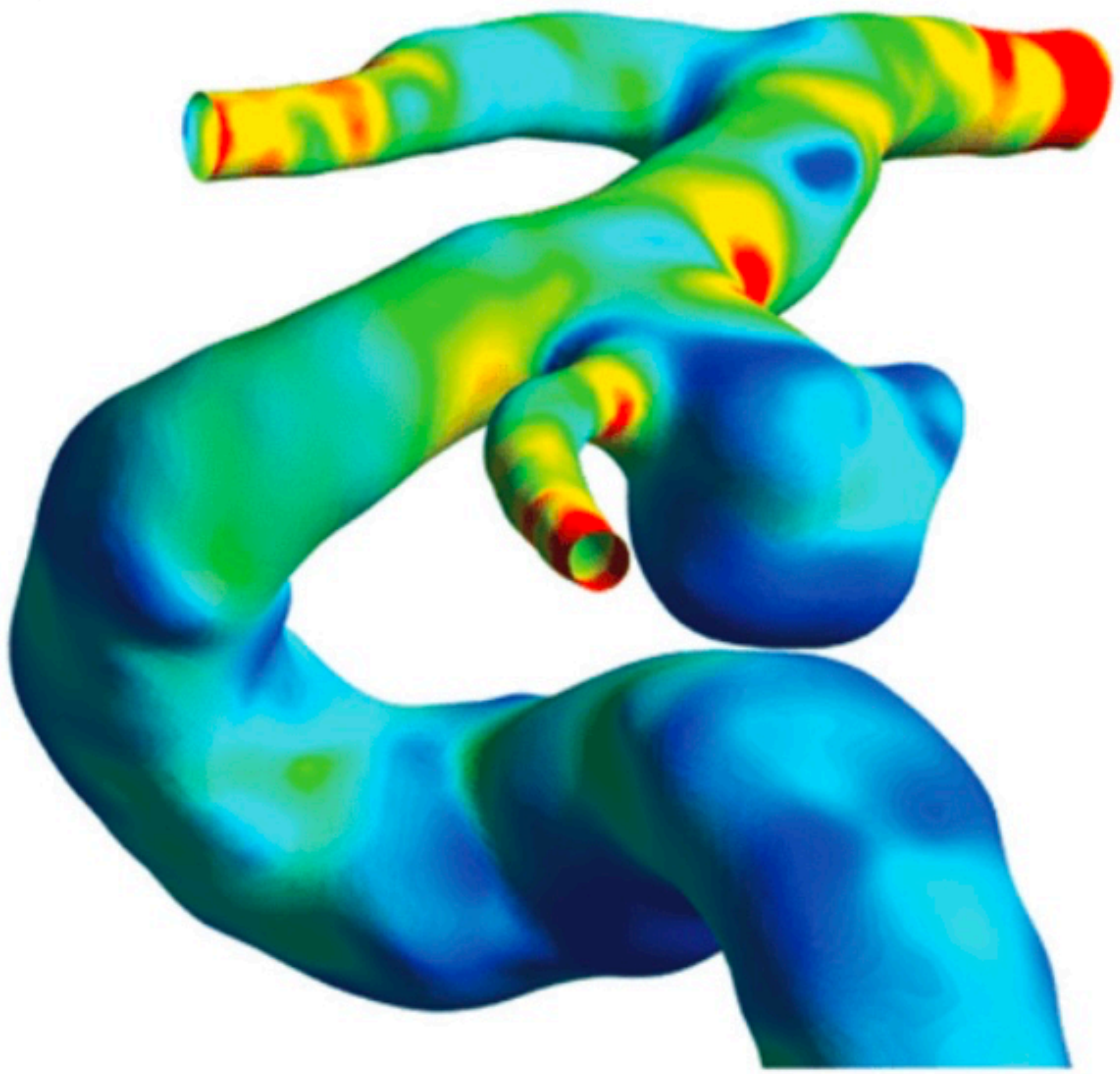


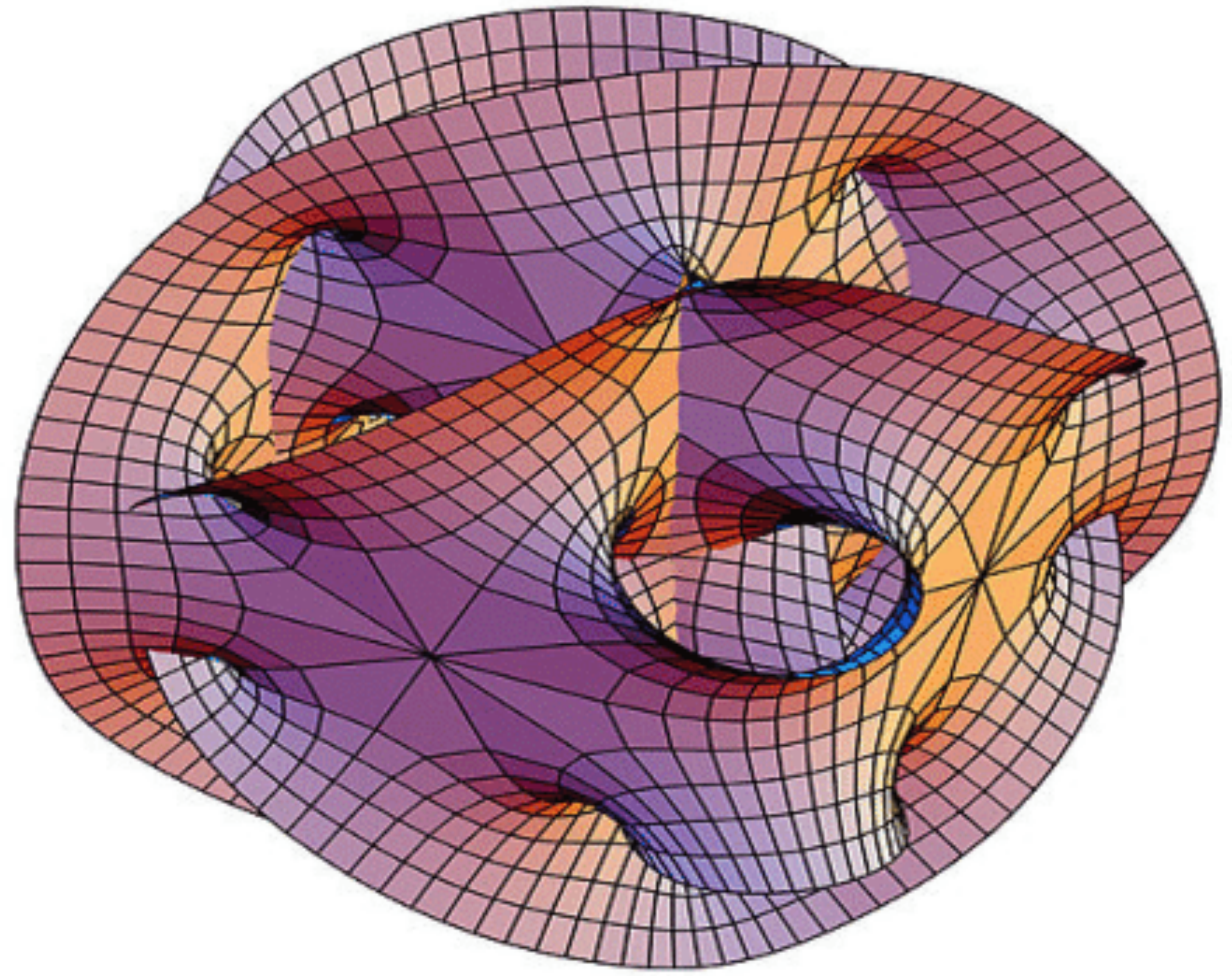
Design neural networks

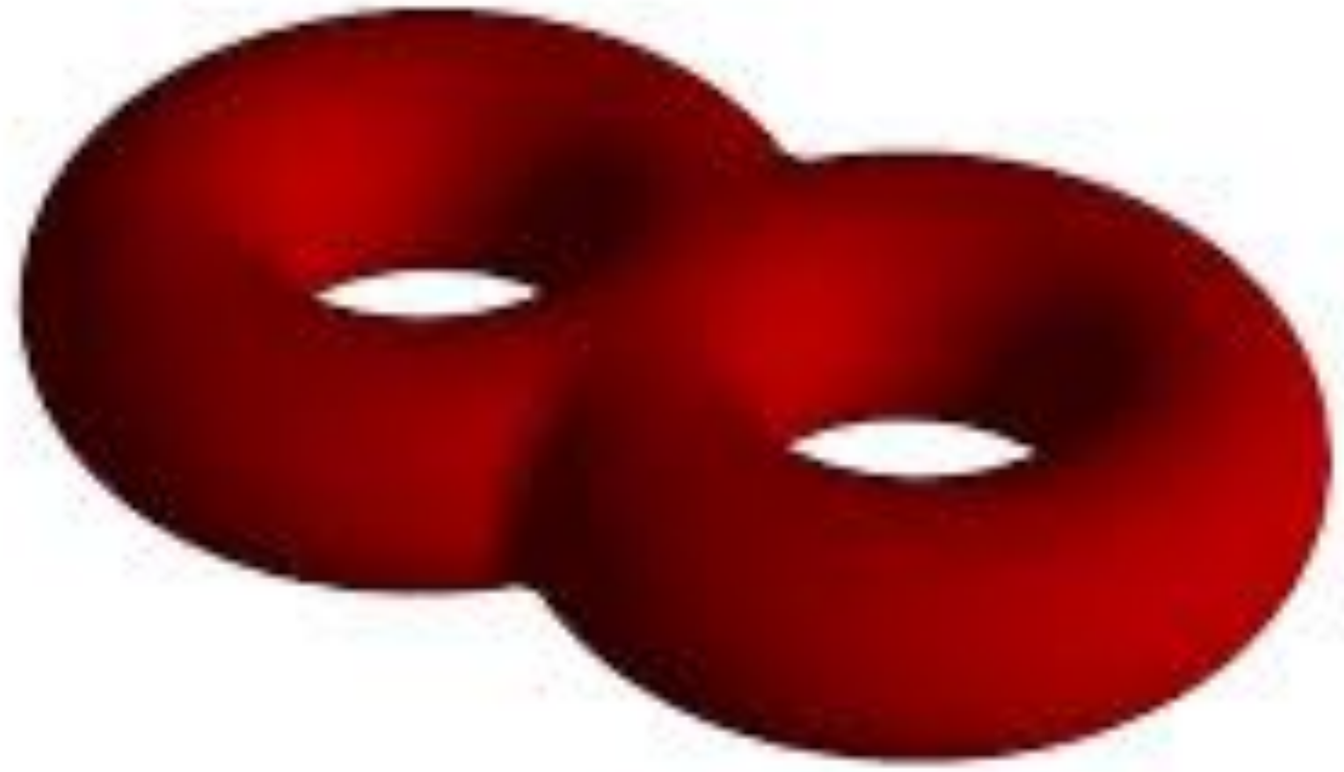
that have **intrinsic** symmetries

(equivariance)

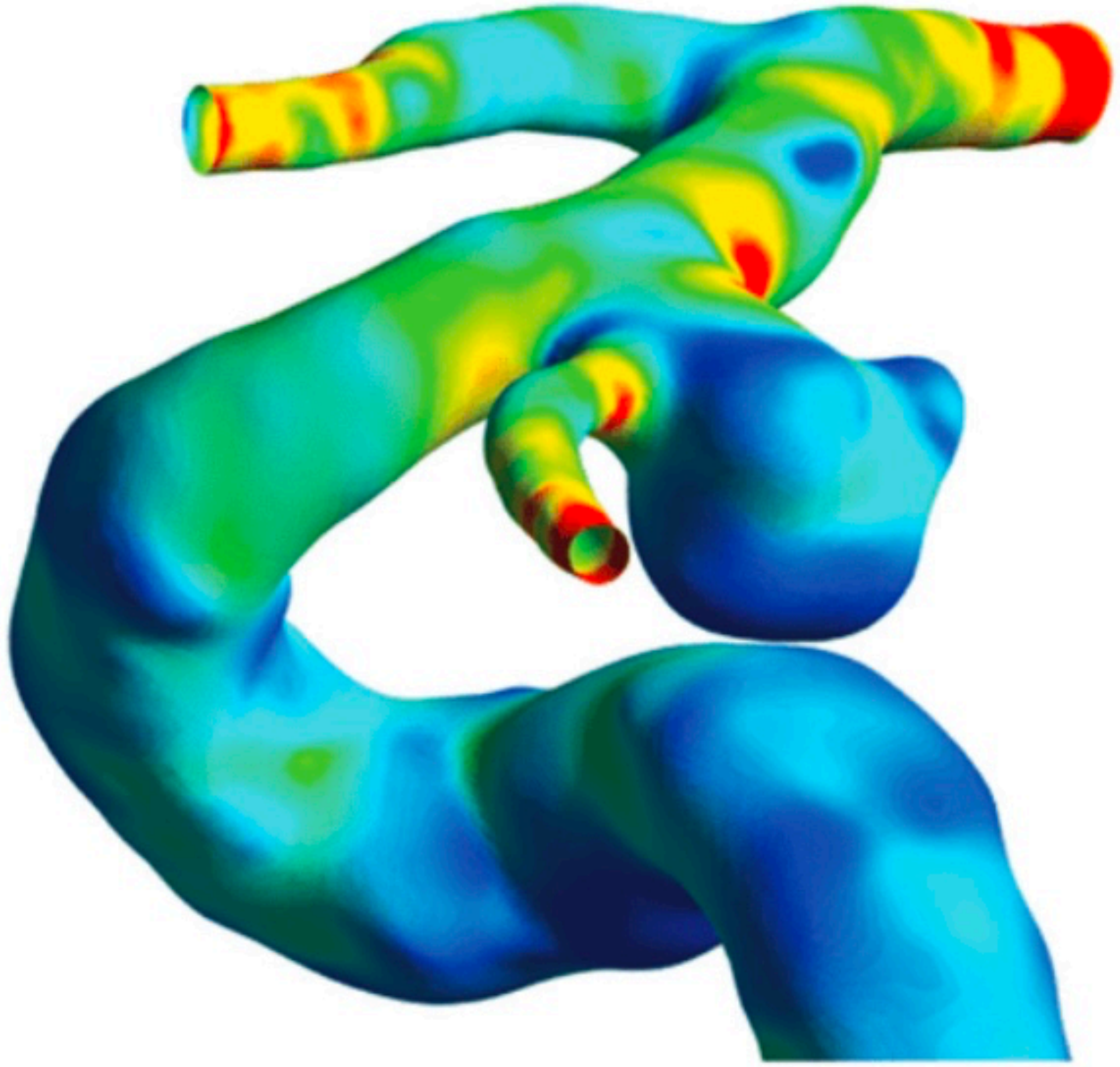
Geometric deep learning: Deep learning on manifolds







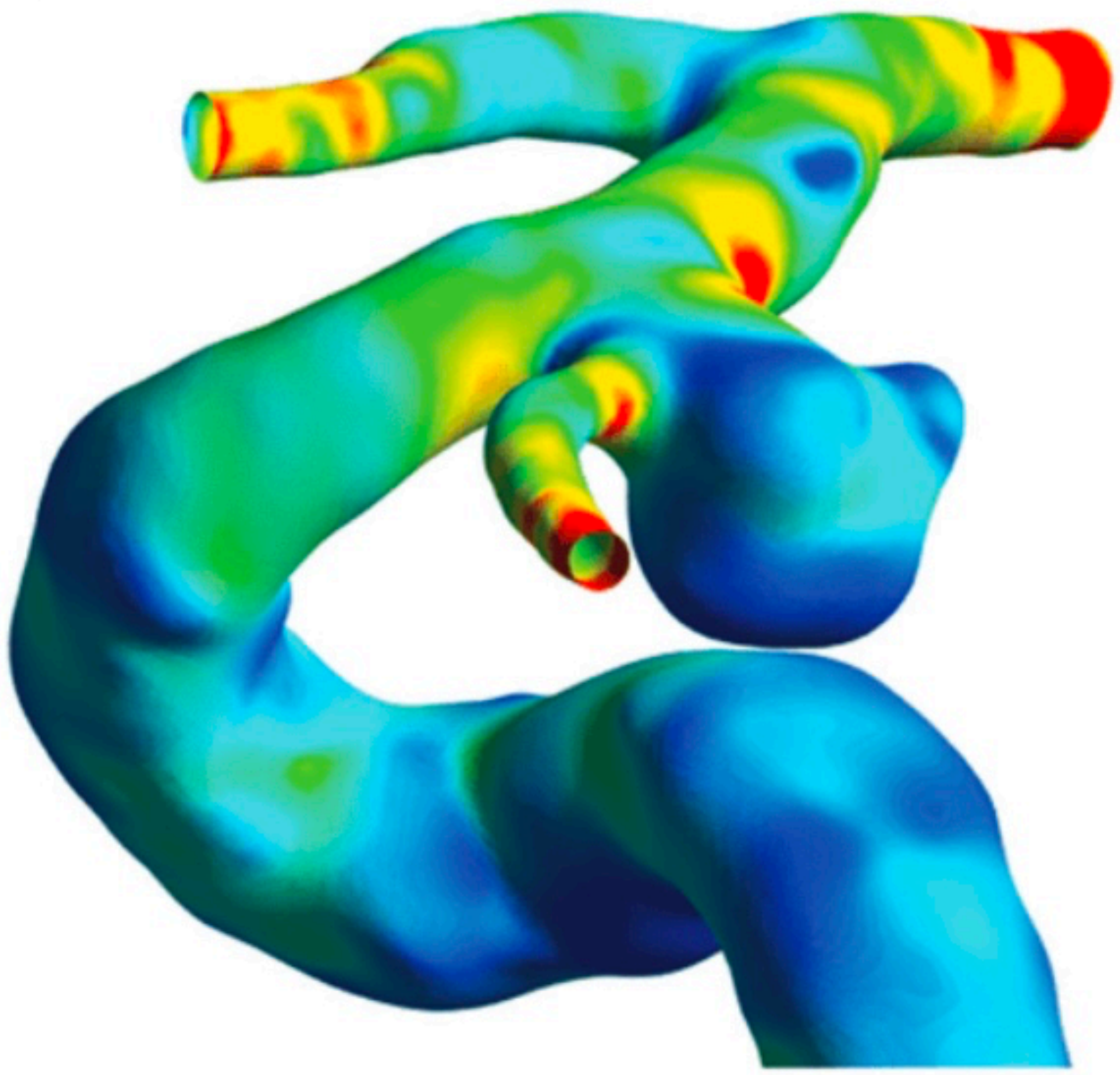
Geometric deep learning - A unified framework for deep learning



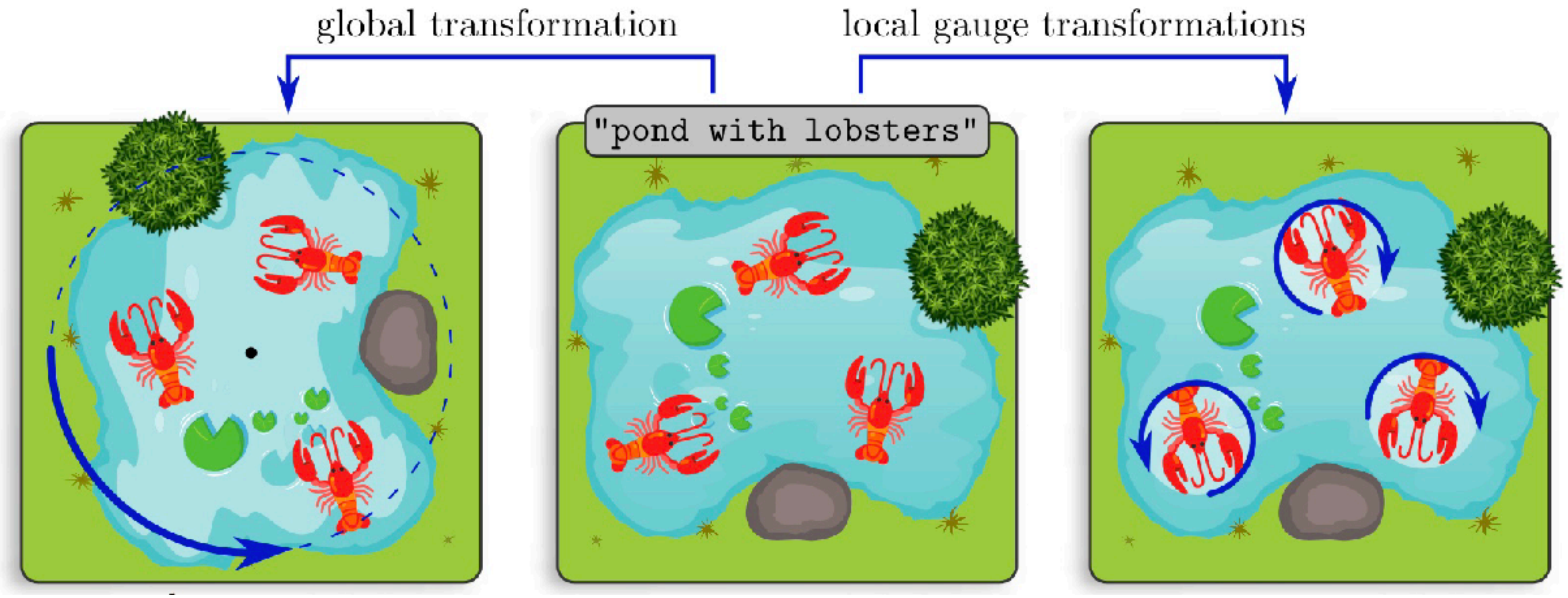
Principle of geometric deep learning:

The equations governing neural networks should be equivariant with respect to all **local** and **global** symmetries of the input data

Geometric deep learning - A unified framework for deep learning

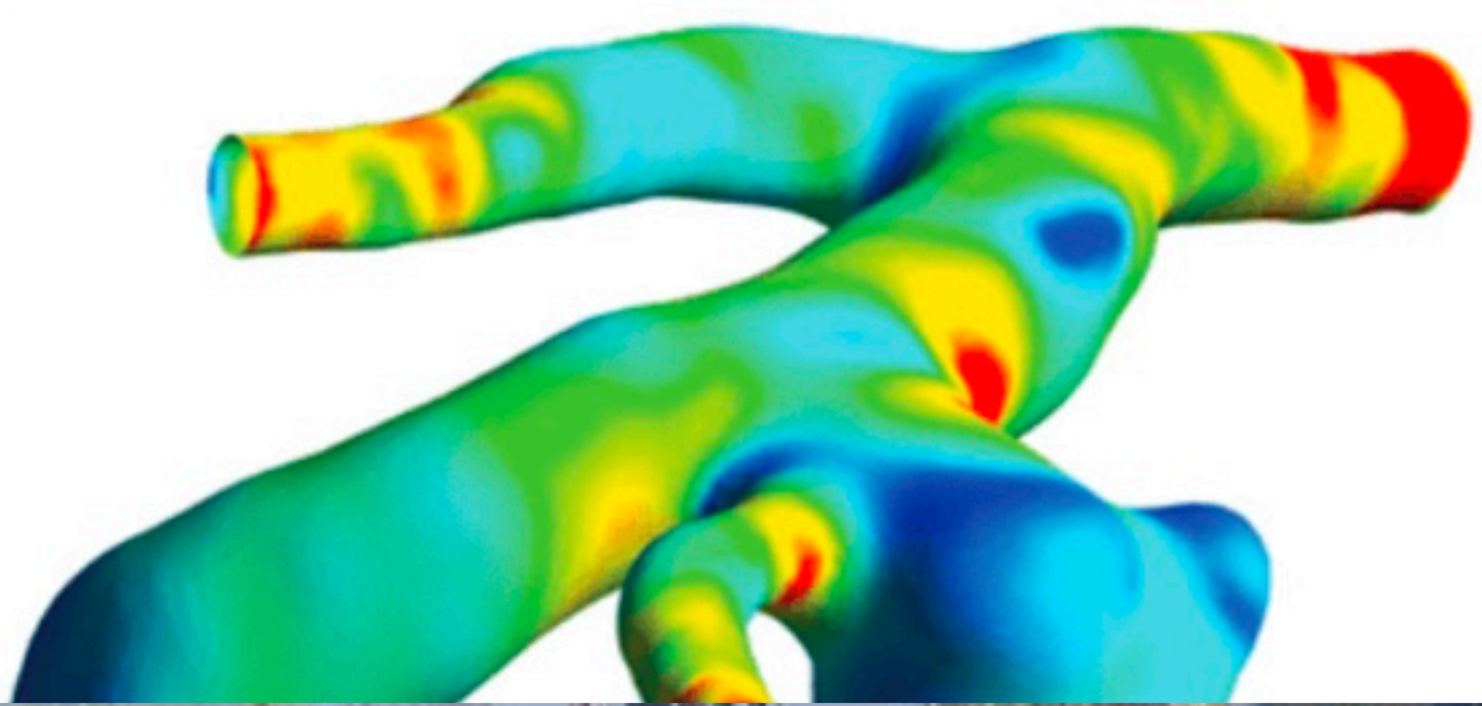


Principle of geometric deep learning:
The equations governing neural networks should be equivariant with respect to all **local** and **global** symmetries of the input data

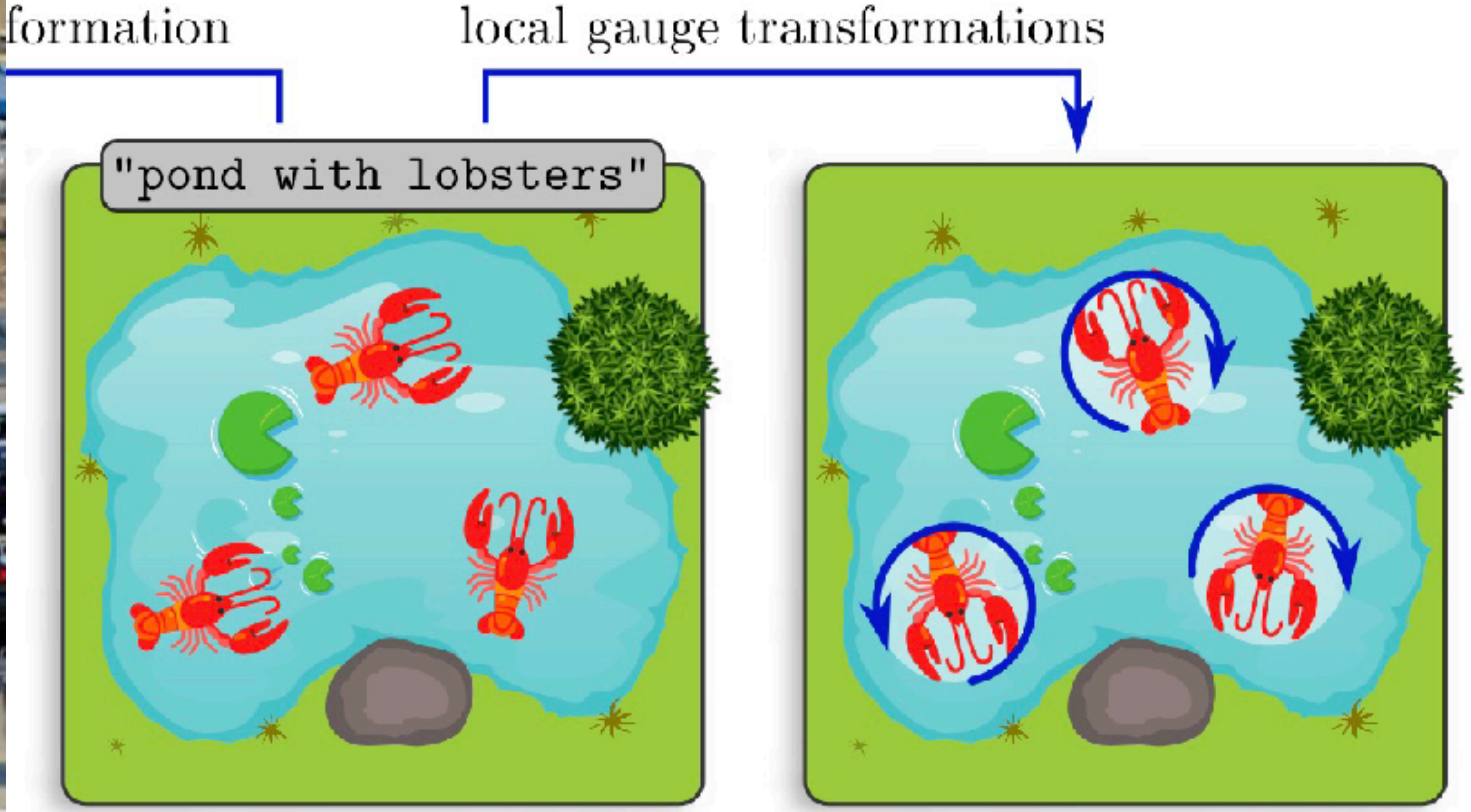


[Images from: Weiler, Forré, Verlinde, Welling (2023)]

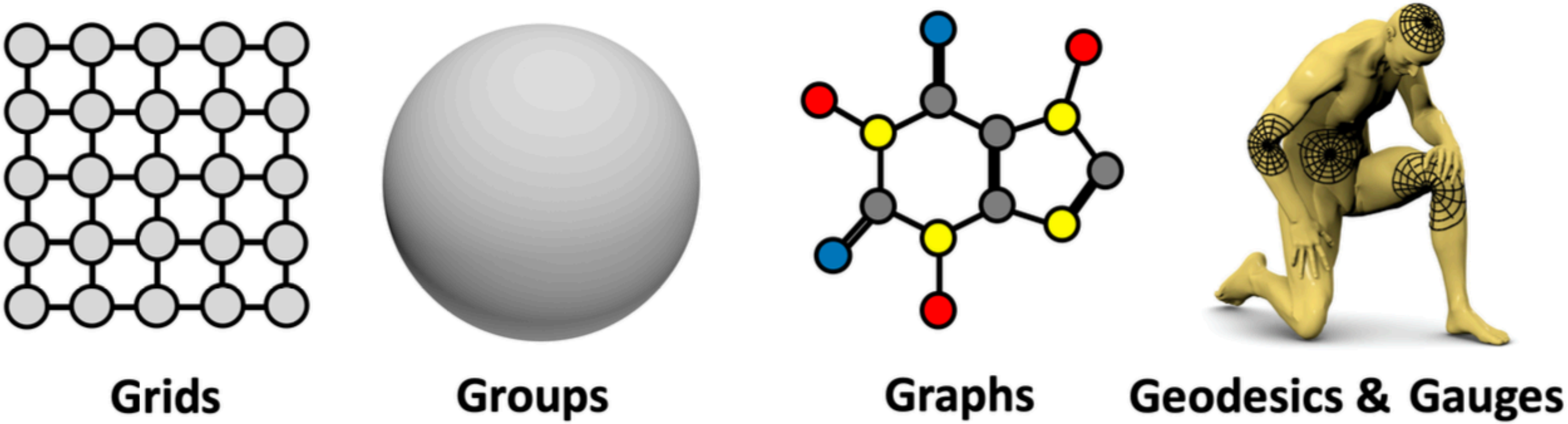
Geometric deep learning - A unified framework for deep learning



Principle of geometric deep learning:
The equations governing neural networks should be equivariant with respect to all **local** and **global** symmetries of the input data

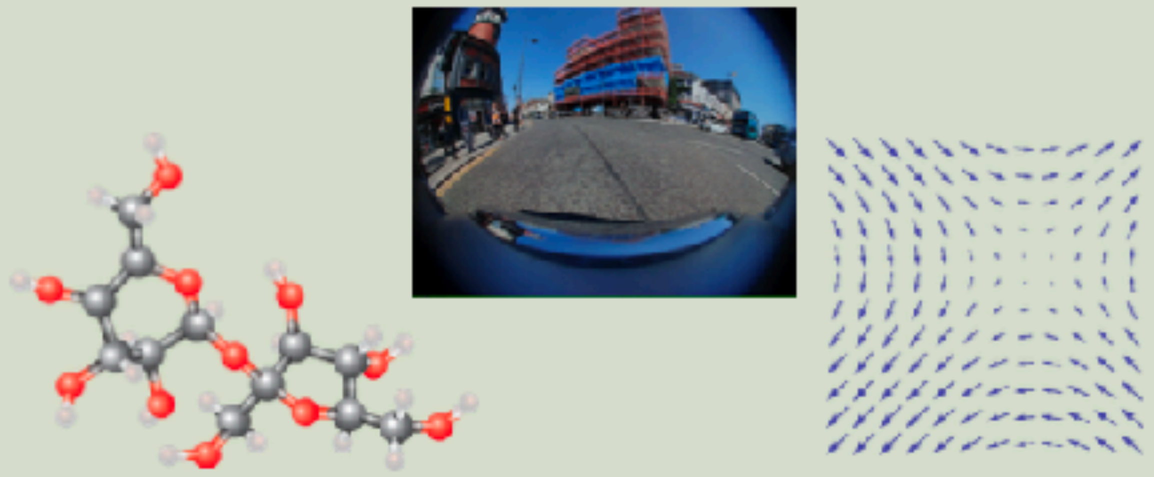


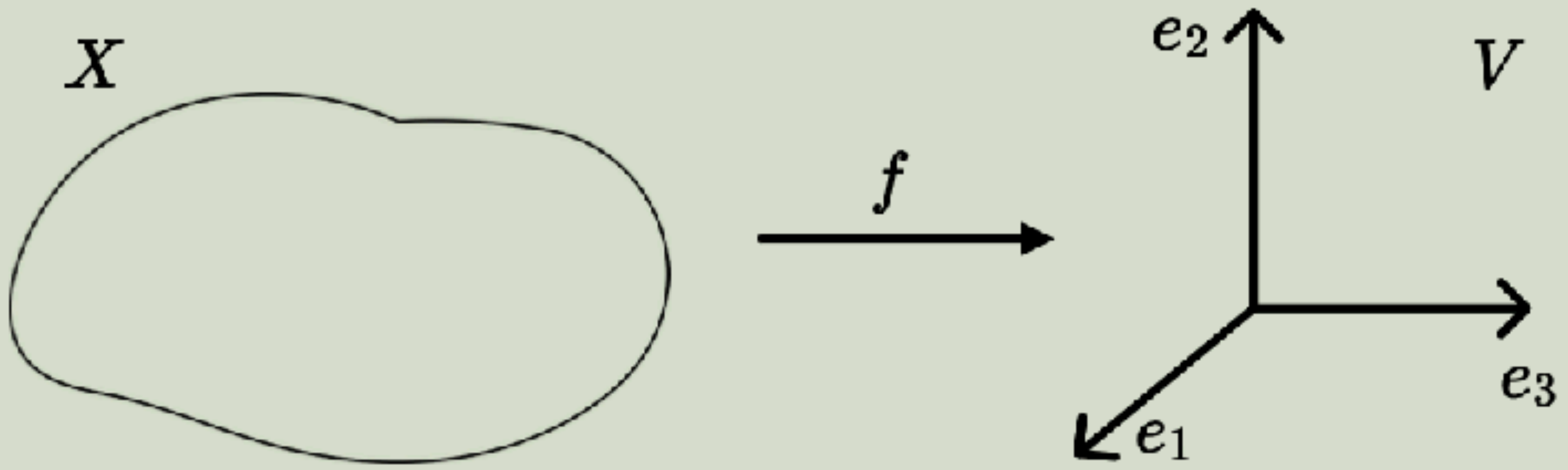
om: Weiler, Forré, Verlinde, Welling (2023)]



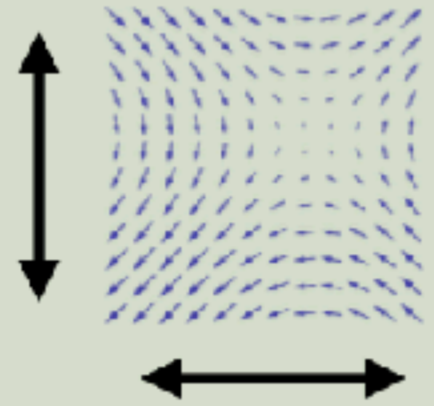
[Image from: Bronstein, Bruna, Cohen, Velickovic (2021)]

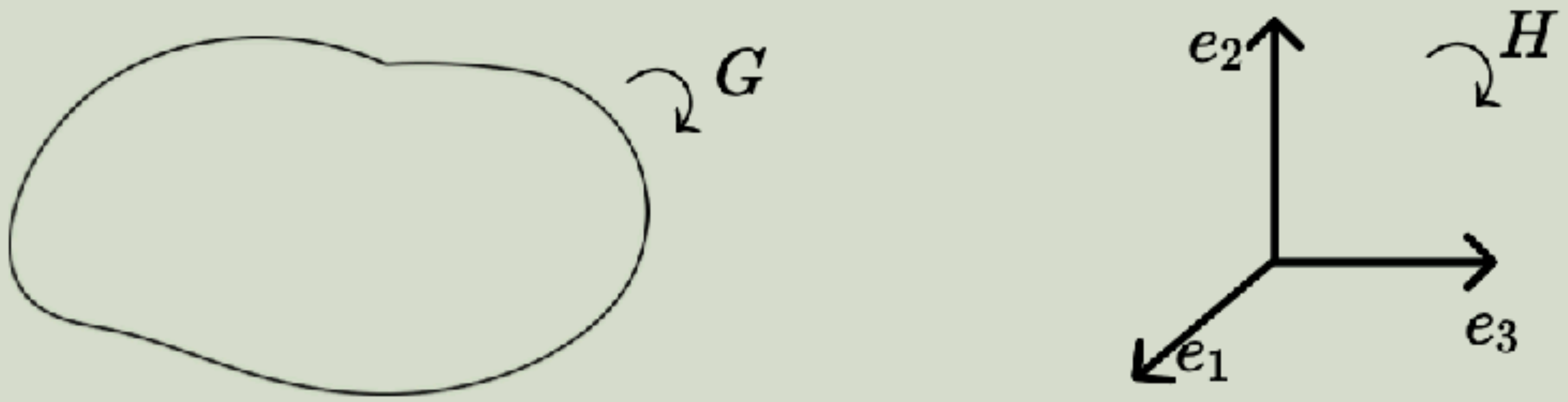
Vector data ⇝ **Vector-valued function $f : X \to V$**



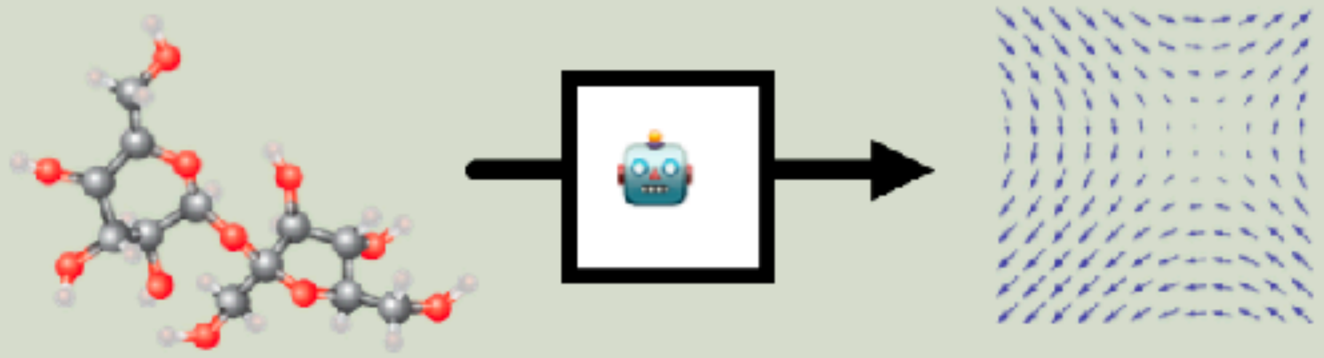


Symmetries in data ⇝ **Group action of G on X + representation of H on V**





Machine learning model ⇝ **Operator $\Phi : \mathrm{Map}(X, V) \to \mathrm{Map}(X, V)$**



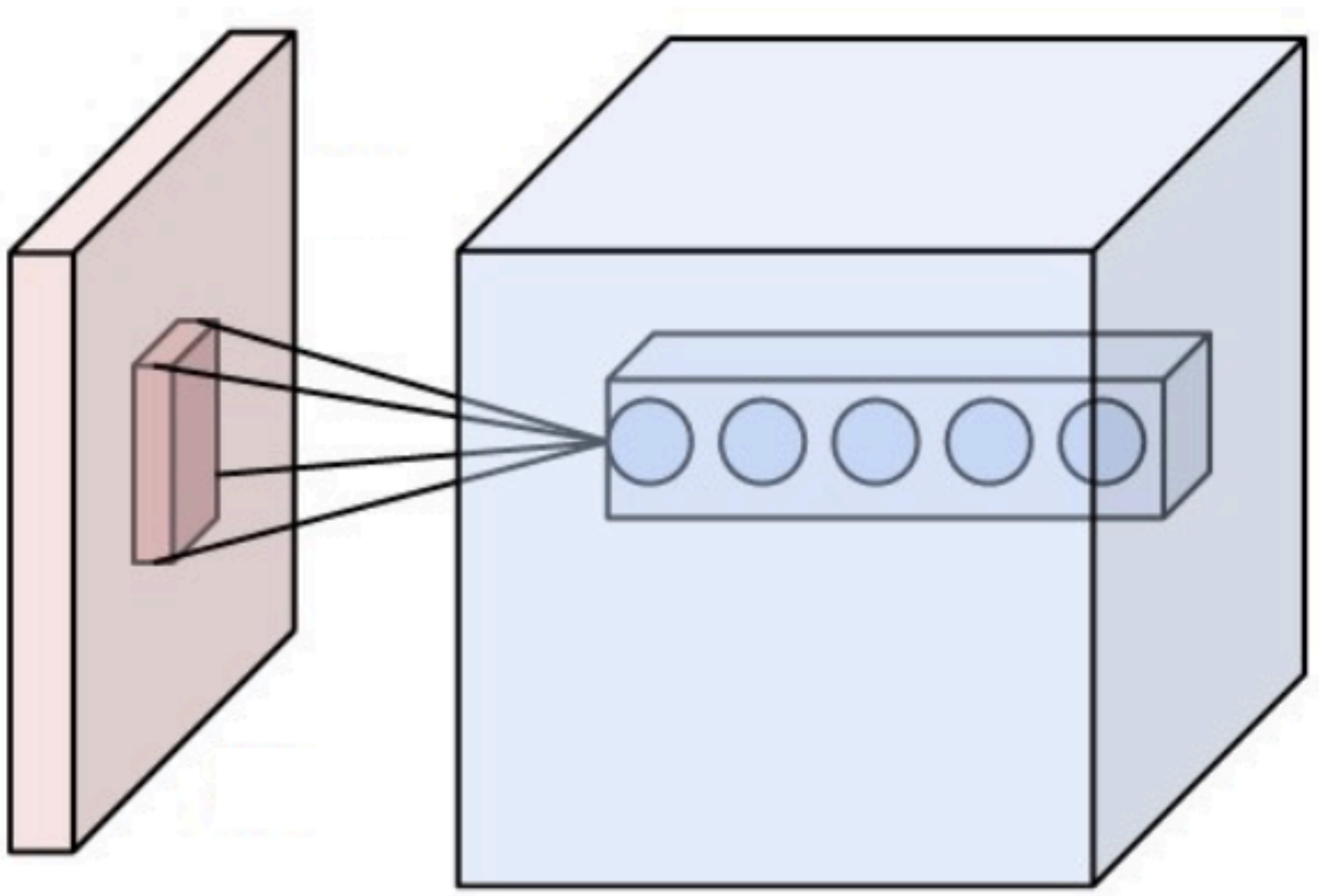
$$f \xrightarrow{\Phi} \Phi f$$

[Slide borrowed from my PhD student Elias!]

Convolutional Neural Networks

"Convolutional networks are simply neural networks that use convolution in place of general matrix multiplication in at least one of their layers."

[Goodfellow, Bengio, Courville]



Mathematical structure

For each layer we have a **feature map**:

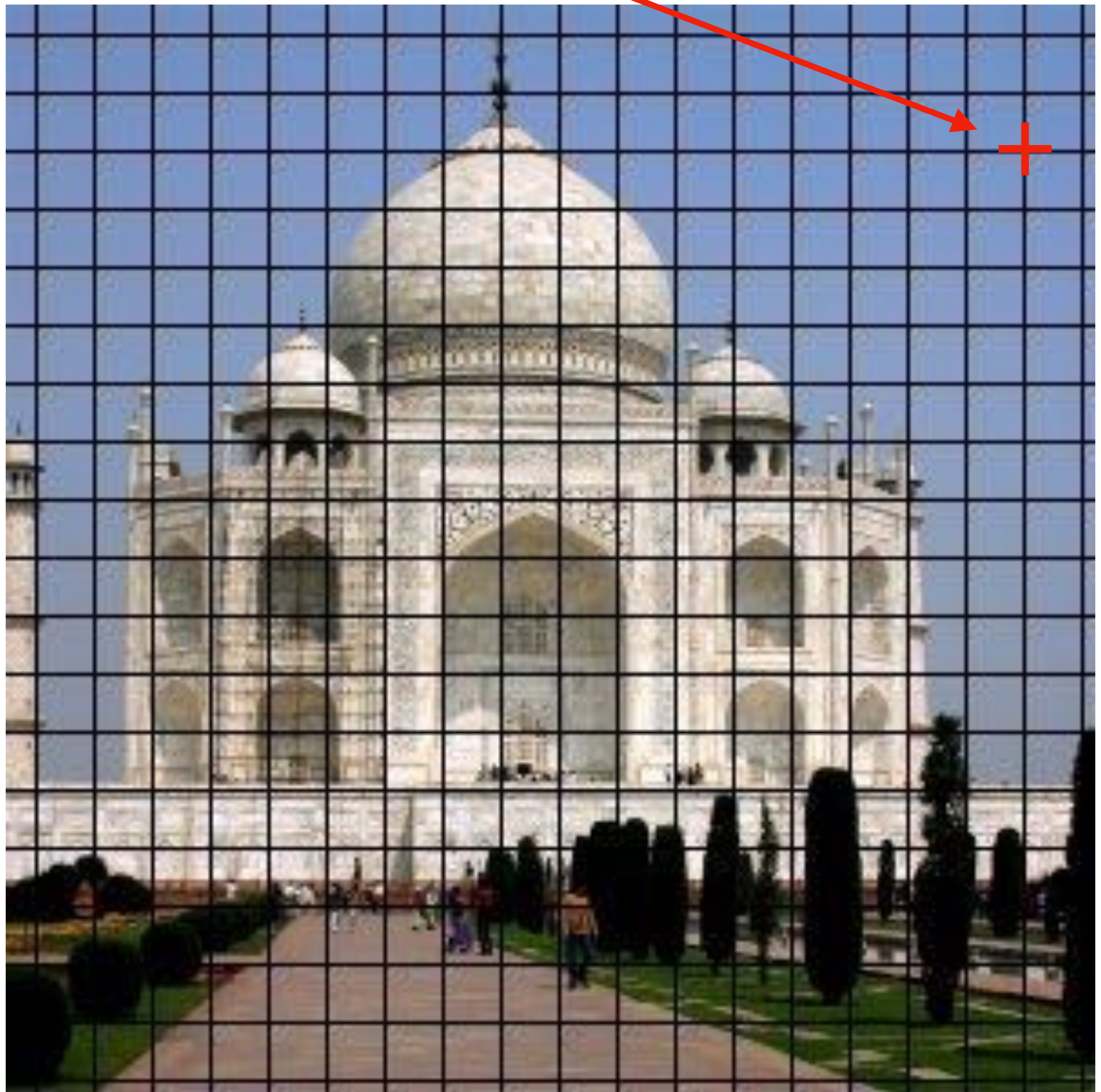
$$f : \mathbb{Z}^2 \to \mathbb{R}^K$$

Mathematical structure

For each layer we have a **feature map**:

no. of channels

$$f : \mathbb{Z}^2 \to \mathbb{R}^K$$



(p, q)

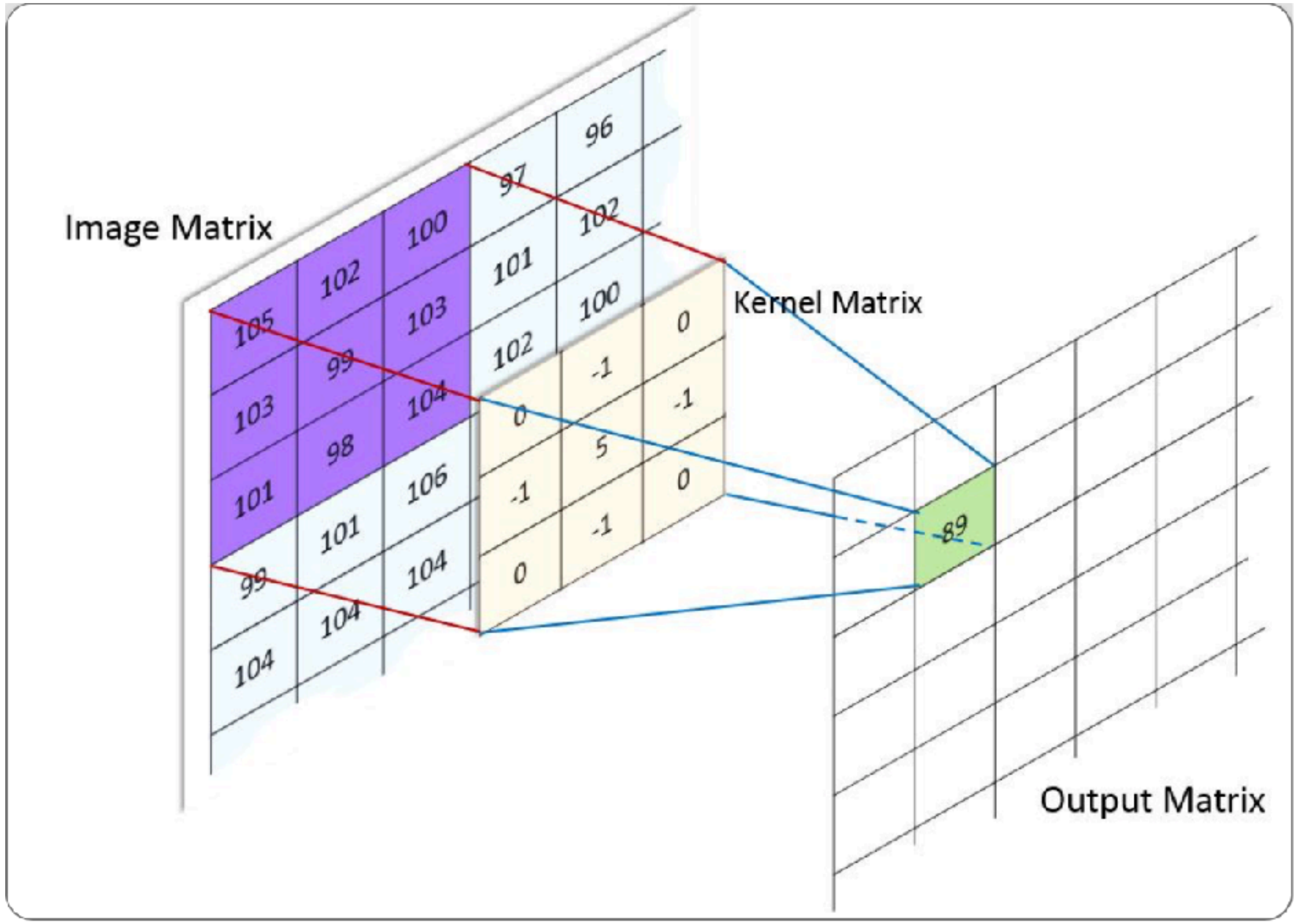
pixel coordinate

Kernel (filter): $\psi : \mathbb{Z}^2 \to \mathbb{R}^K$

Convolution: $$[f * \psi](x) = \sum_{y \in \mathbb{Z}^2} \sum_{k=1}^{K} f_k(y)\psi_k(x - y)$$

Kernel (filter): $\psi : \mathbb{Z}^2 \to \mathbb{R}^K$

Convolution: $$[f * \psi](x) = \sum_{y \in \mathbb{Z}^2} \sum_{k=1}^{K} f_k(y) \psi_k(x - y)$$



[Figure from machinelearninguru.com]

Kernel (filter): $\psi : \mathbb{Z}^2 \rightarrow \mathbb{R}^K$

Convolution: $$[f * \psi](x) = \sum_{y \in \mathbb{Z}^2} \sum_{k=1}^{K} f_k(y) \psi_k(x - y)$$

Translation map: $$[T(t)f](x) = f(x + t)$$

Kernel (filter): $\psi : \mathbb{Z}^2 \to \mathbb{R}^K$

Convolution: $$[f * \psi](x) = \sum_{y \in \mathbb{Z}^2} \sum_{k=1}^{K} f_k(y) \psi_k(x - y)$$

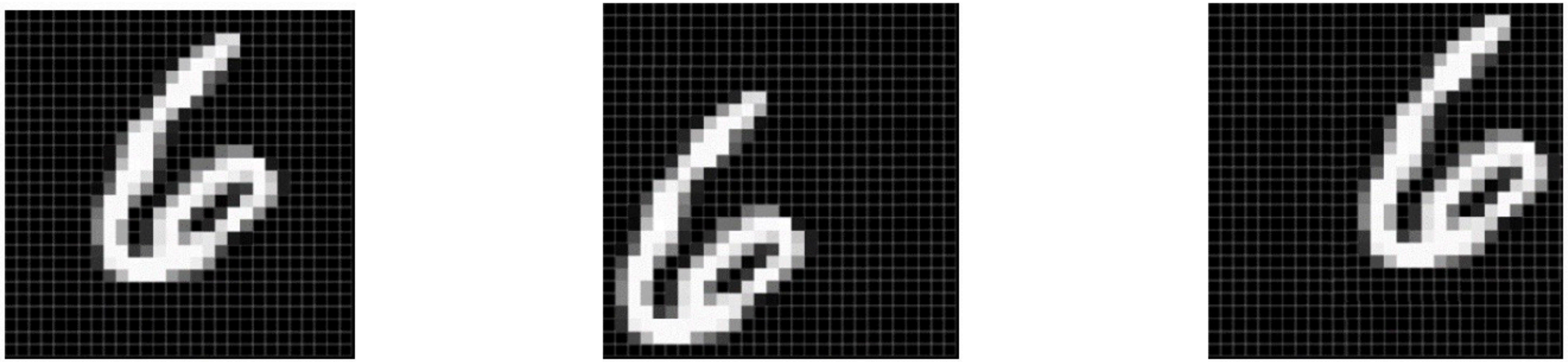
Translation map: $$[T(t)f](x) = f(x + t)$$

Convolution is equivariant w.r.t. translations

$$[T(t)f] * \psi = T(t)[f * \psi]$$

Convolution is equivariant w.r.t. translations

$$[T(t)f] * \psi = T(t)[f * \psi]$$



But what about more general symmetries?

Locally, we can think of feature maps as functions $f : G/H \to V$

Example: $G = \mathbb{Z}^2$ $H = \{1\}$ $V = \mathbb{R}^K$

Feature maps are sections! $f : \mathbb{Z}^2 \to \mathbb{R}^K$

Locally, we can think of feature maps as functions $f : G/H \to V$

Example: $G = \mathbb{Z}^2$ $H = \{1\}$ $V = \mathbb{R}^K$

Feature maps are sections! $f : \mathbb{Z}^2 \to \mathbb{R}^K$

This is a vector bundle:

$$\begin{array}{ccc} V & \longrightarrow & P \\ & & \Big\downarrow p \\ & & G/H \end{array}$$

Locally, we can think of feature maps as functions $f : G/H \to V$

Example: $G = \mathbb{Z}^2$ $H = \{1\}$ $V = \mathbb{R}^K$

Feature maps are sections! $f : \mathbb{Z}^2 \to \mathbb{R}^K$

General structure of group equivariant CNNs:

$$\left\{ \begin{array}{c} \text{Feature space of a} \\ G-\text{equivariant CNN} \end{array} \right\} \cong \left\{ \begin{array}{c} \text{Sections of vector} \\ \text{bundles } \; P \to G/H \end{array} \right\}$$

Layers defined with group-equivariant convolutions:

$$[f * \psi](g) = \int_G \sum_{k=1}^{K} f_k(h) \psi_k(gh) dh$$

[Kondor, Trivedi][Cohen, Geiger, Weiler]

$$\left\{ \begin{array}{c} \text{Feature space of a} \\ G-\text{equivariant CNN} \end{array} \right\} \cong \left\{ \begin{array}{c} \text{Sections of vector} \\ \text{bundles } P \to G/H \end{array} \right\}$$

Sections of $P \to G/H$ belong to the **induced representation**:

$$\mathcal{F} = \mathrm{Ind}_H^G \rho = \{ f : G \to V \mid f(gh) = \rho(h^{-1}) f(g) \}$$

$$\left\{ \begin{array}{c} \text{Feature space of a} \\ G-\text{equivariant CNN} \end{array} \right\} \cong \left\{ \begin{array}{c} \text{Sections of vector} \\ \text{bundles } P \to G/H \end{array} \right\}$$

Sections of $P \to G/H$ belong to the **induced representation**:

$$\mathcal{F} = \text{Ind}_H^G \rho = \{f : G \to V \mid f(gh) = \rho(h^{-1}) f(g)\}$$

$$\left\{ \begin{array}{c} \text{Maps between layers} \\ \text{in the CNN} \end{array} \right\} \cong \left\{ \begin{array}{c} G-\text{equivariant linear maps} \\ \text{between feature spaces } \mathcal{F} \to \mathcal{F}' \end{array} \right\}$$

$$\left\{\begin{array}{c}\text{Feature space of a}\\ G-\text{equivariant CNN}\end{array}\right\} \cong \left\{\begin{array}{c}\text{Sections of vector}\\ \text{bundles } P \to G/H\end{array}\right\}$$

Sections of $P \to G/H$ belong to the **induced representation**:

$$\mathcal{F} = \operatorname{Ind}_H^G \rho = \{f : G \to V \mid f(gh) = \rho(h^{-1}) f(g)\}$$

$$\left\{\begin{array}{c}\text{Maps between layers}\\ \text{in the CNN}\end{array}\right\} \cong \left\{\begin{array}{c}G-\text{equivariant linear maps}\\ \text{between feature spaces } \mathcal{F} \to \mathcal{F}'\end{array}\right\}$$

$$\operatorname{Hom}_G(\mathcal{F}, \mathcal{F}')$$

(intertwining operators)

Equivariant non-linear maps for neural networks on homogeneous spaces

Elias Nyholm, Oscar Carlsson, Maurice Weiler, **D.P.** [arXiv:2504.20974]

A mathematical formalism for non-linear neural networks!

Equivariant non-linear maps for neural networks on homogeneous spaces

Elias Nyholm, Oscar Carlsson, Maurice Weiler, **D.P.** [arXiv:2504.20974]

A mathematical formalism for non-linear neural networks!

Linear case:

$$\Phi \overset{\text{(Homogeneity)}}{\Longrightarrow} [\Phi f](g) \overset{\text{(Linearity)}}{\Longrightarrow} [\Phi f](g) = \int_G \kappa(g, g') f(g') dg' \overset{\text{(Equivariance)}}{\Longrightarrow} [\Phi f](g) = \int_G \tilde{\kappa}(g^{-1} g') f(g') dg'$$

[Slide borrowed from my PhD student Elias!]

Equivariant non-linear maps for neural networks on homogeneous spaces

Elias Nyholm, Oscar Carlsson, Maurice Weiler, **D.P.** [arXiv:2504.20974]

A mathematical formalism for non-linear neural networks!

Linear case:

$$\Phi \overset{\text{(Homogeneity)}}{\Longrightarrow} [\Phi f](g) \overset{\text{(Linearity)}}{\Longrightarrow} [\Phi f](g) = \int_G \kappa(g, g') f(g') dg' \overset{\text{(Equivariance)}}{\Longrightarrow} [\Phi f](g) = \int_G \tilde{\kappa}(g^{-1} g') f(g') dg'$$

Non-linear generalization:

$$\Phi \overset{\text{(Homogeneity)}}{\Longrightarrow} [\Phi f](g) \overset{\text{(Equivariance)}}{\Longrightarrow} [\Phi f](g) = \tilde{\Phi}(g^{-1} \triangleright f)$$

[Slide borrowed from my PhD student Elias!]

Non-Linear Equivariant Layers on Homogeneous Spaces

Nyholm, Carlsson, Weiler, Persson — arXiv:2504.20974

The linear story (G-CNNs)

- Features on G/H: sections of associated vector bundles $\cong$ induced reps $\mathrm{Ind}_H^G \rho$.
- Linear equivariant layers = convolutions:
  $$[\Phi f](g) = \int_G \hat{\kappa}(g^{-1}g')\, f(g')\, dg'$$
- Kernel $\hat{\kappa}$ satisfies **steerability**: $\hat{\kappa}(hgh') = \sigma(h)\,\hat{\kappa}(g)\,\rho(h')$.
- Well understood (Cohen et al. 2018), but *only describes linear layers*.

Our generalisation: non-linear layers

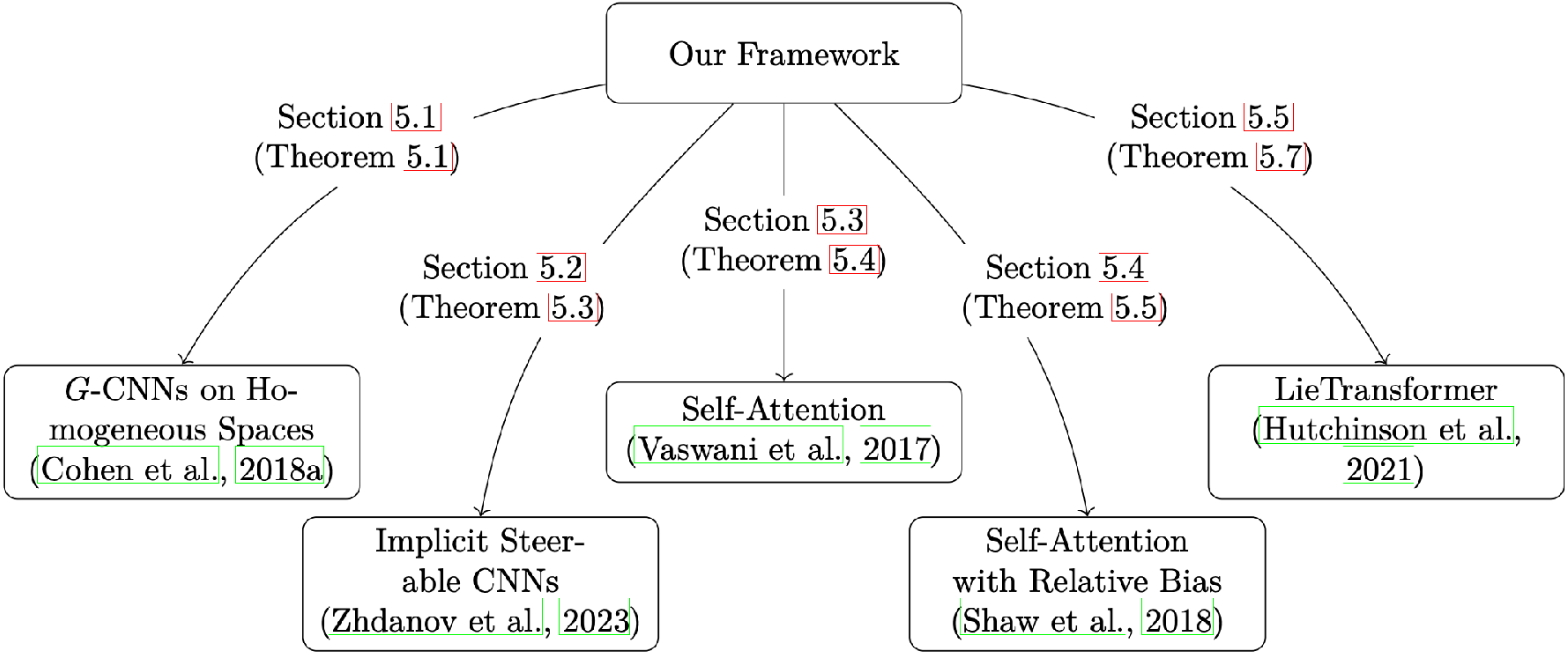
Replace the *input-independent* kernel with an **input-dependent** integrand $\hat{\omega}$:

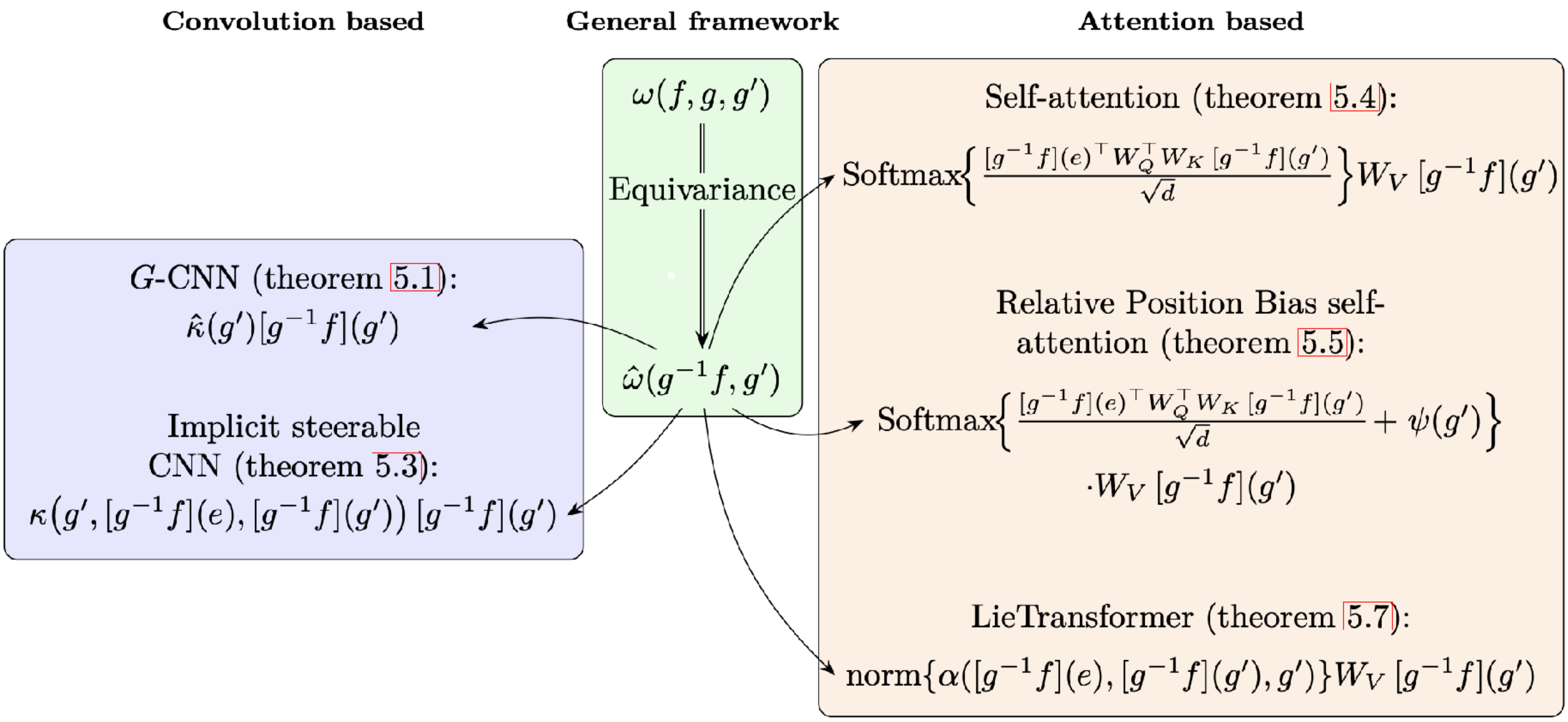
$$[\Phi f](g) = \int_G \hat{\omega}(g^{-1}f,\ g')\, dg'$$

- Generalised steerability constraint:

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

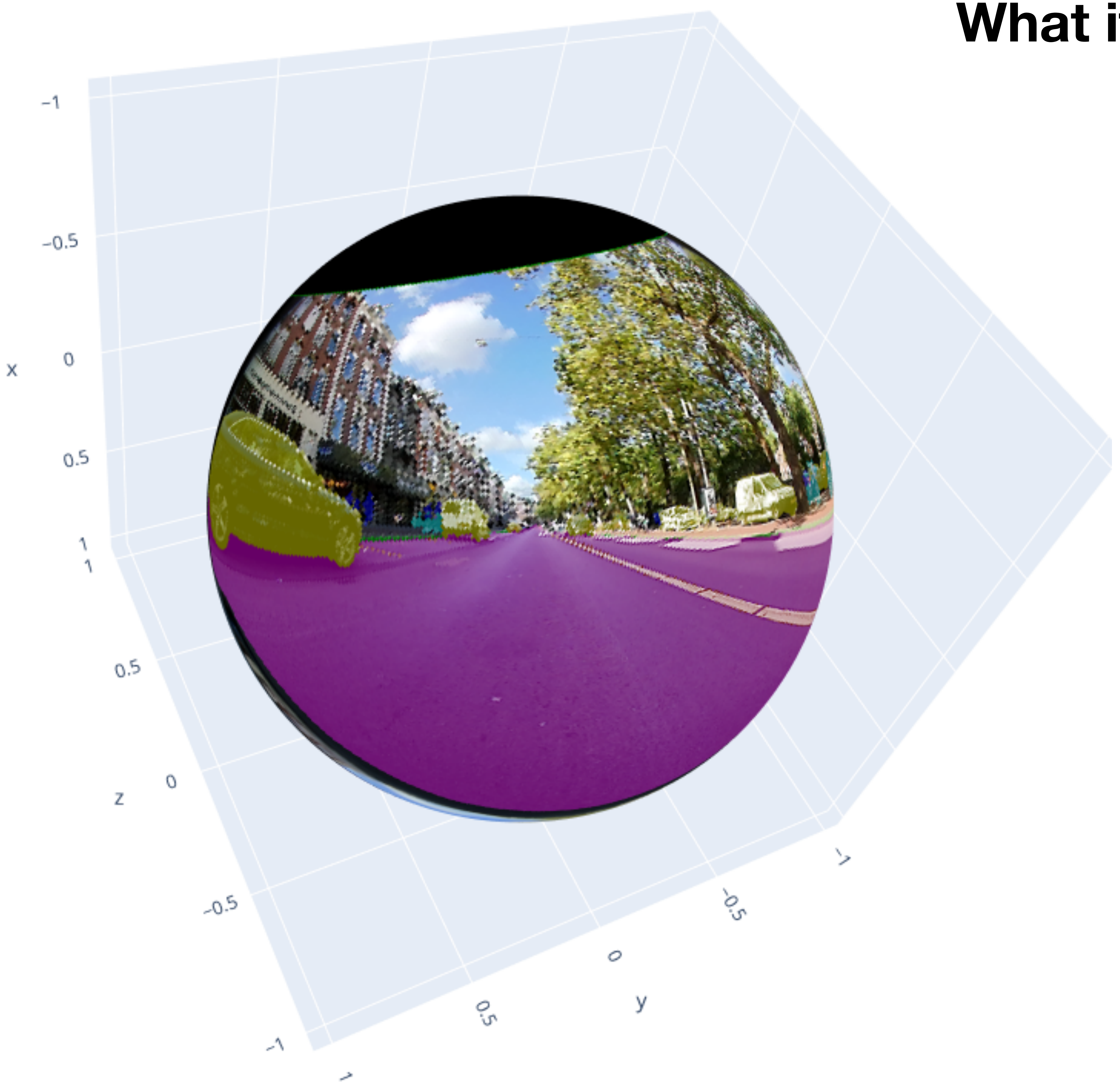
- $\hat{\omega} : \mathrm{Ind}_H^G \rho \times G \to V_\sigma$,
- **Universality**: every equivariant map $\mathrm{Ind}_H^G \rho \to \mathrm{Ind}_H^G \sigma$ can be written in this form.


Our Framework
Section 5.1 (Theorem 5.1)
Section 5.2 (Theorem 5.3)
Section 5.3 (Theorem 5.4)
Section 5.4 (Theorem 5.5)
Section 5.5 (Theorem 5.7)
G-CNNs on Homogeneous Spaces (Cohen et al., 2018a)
Implicit Steerable CNNs (Zhdanov et al., 2023)
Self-Attention (Vaswani et al., 2017)
Self-Attention with Relative Bias (Shaw et al., 2018)
LieTransformer (Hutchinson et al., 2021)

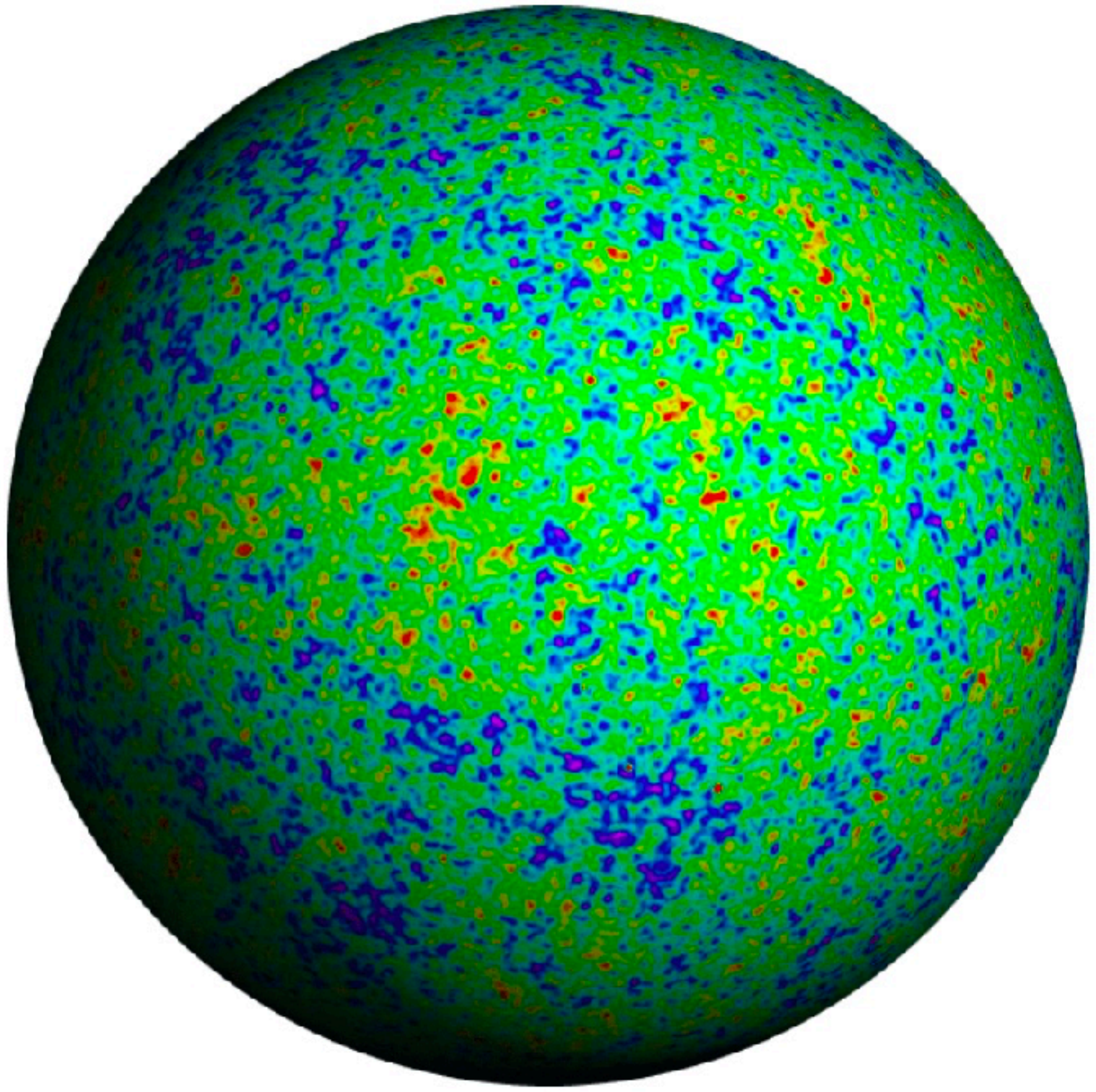

Convolution based
General framework
Attention based
$\omega(f, g, g')$
Equivariance
$\hat{\omega}(g^{-1}f, g')$
G-CNN (theorem 5.1):
$\hat{\kappa}(g')[g^{-1}f](g')$
Implicit steerable CNN (theorem 5.3):
$\kappa\big(g', [g^{-1}f](e), [g^{-1}f](g')\big)\ [g^{-1}f](g')$
Self-attention (theorem 5.4):
$\text{Softmax}\Big\{\frac{[g^{-1}f](e)^\top W_Q^\top W_K\,[g^{-1}f](g')}{\sqrt{d}}\Big\} W_V\,[g^{-1}f](g')$
Relative Position Bias self-attention (theorem 5.5):
$\text{Softmax}\Big\{\frac{[g^{-1}f](e)^\top W_Q^\top W_K\,[g^{-1}f](g')}{\sqrt{d}} + \psi(g')\Big\}$
$\cdot W_V\,[g^{-1}f](g')$
LieTransformer (theorem 5.7):
$\text{norm}\{\alpha([g^{-1}f](e), [g^{-1}f](g'), g')\} W_V\,[g^{-1}f](g')$

Applications

What if the input data is curved?



[Image from the Woodscape dataset, projected onto a sphere]



Cosmic microwave background radiation

Equivariance versus augmentation for spherical signals

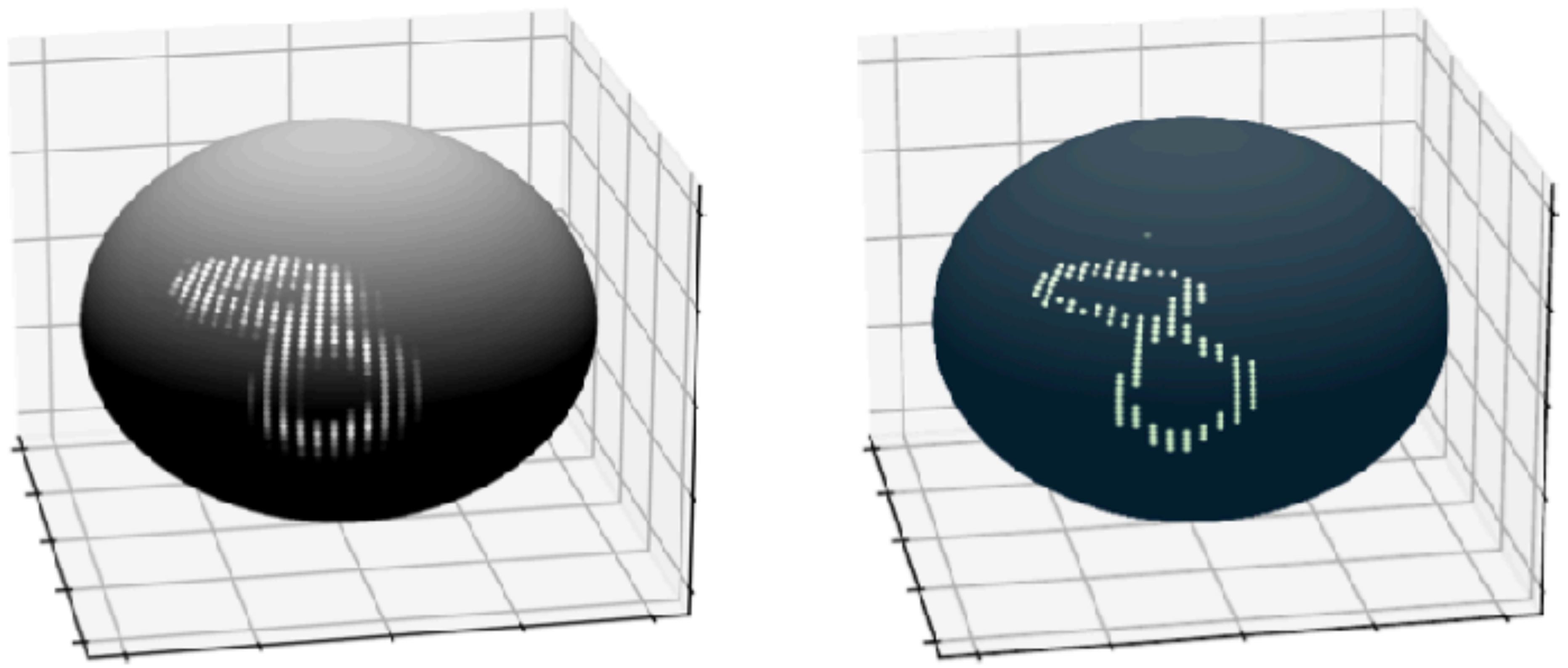
By Gerken, Carlsson, Linander, Ohlsson, Petersson, **D.P.**

[ICML 2022, arXiv: 2202.03990]

$$G = SO(3)$$

$$H = SO(2)$$

$$G/H \cong S^2$$



Feature maps $f : S^2 \to \mathbb{R}^K$

Relevant for :

- Omnidirectional vision
- Weather and climate data
- Cosmology & astrophysics

$$(\kappa \star f)(R) = \int_{S^2} \kappa(R^{-1}x) f(x)\, \mathrm{d}x$$

$$(\kappa \star f)(R) = \int_{\mathrm{SO}(3)} \kappa(S^{-1}R) f(S)\, \mathrm{d}S$$

Equivariance versus augmentation for spherical signals

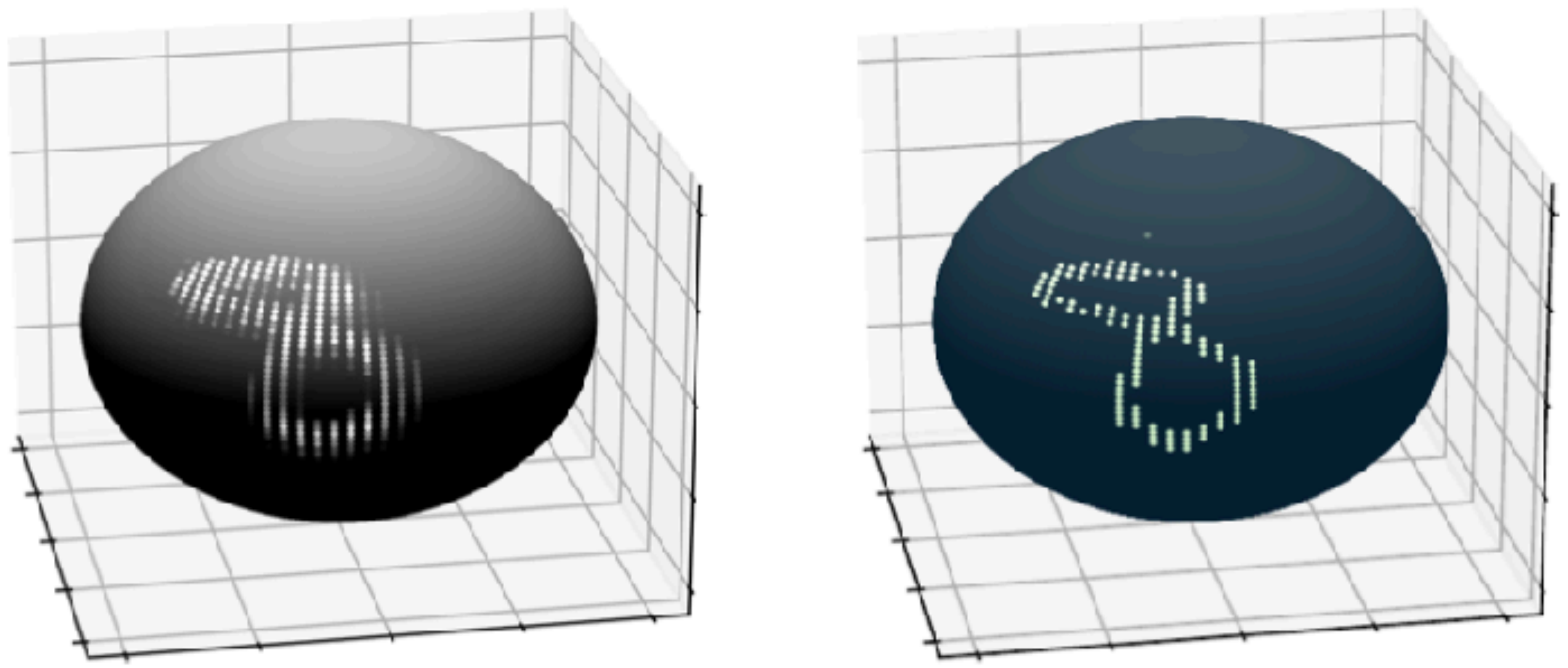
By Gerken, Carlsson, Linander, Ohlsson, Petersson, **D.P.**

[ICML 2022, arXiv: 2202.03990]

$$G = SO(3)$$

$$H = SO(2)$$

$$G/H \cong S^2$$



Feature maps $f : S^2 \to \mathbb{R}^K$

- We demonstrate that non-equivariant classification models require considerable data augmentation to reach the performance of smaller equivariant networks
- We show that the performance of non-equivariant semantic segmentation models saturates well below that of equivariant models as the amount of data augmentation is increased

$$(\kappa \star f)(R) = \int_{S^2} \kappa(R^{-1}x) f(x)\ \mathrm{d}x$$

$$(\kappa \star f)(R) = \int_{\mathrm{SO}(3)} \kappa(S^{-1}R) f(S)\, \mathrm{d}S$$

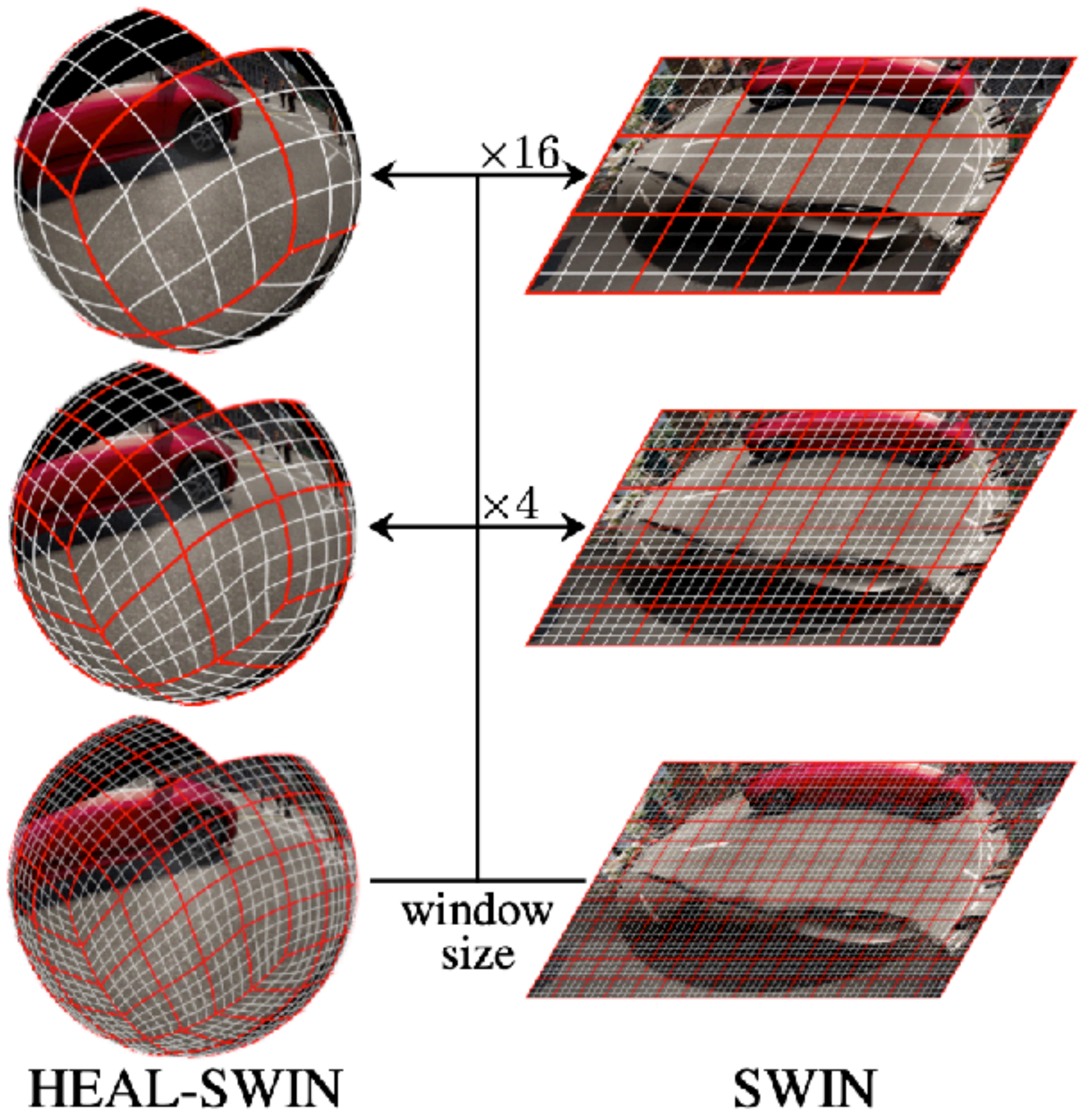
HEAL-SWIN: A Vision Transformer On The Sphere

[CVPR 2024]

Oscar Carlsson*[ab] Jan E. Gerken*[a] Hampus Linander[a] Heiner Spieß[c] Fredrik Ohlsson[d]

Christoffer Petersson[ea] Daniel Persson[a]

- We combine an adapted vision transformer with the **Hierarchical Equal Area iso-Latitude Pixelisation** (HEALPix) grid
- HEALPix was developed for capturing the high-resolution measurements of the CMB performed by the WMAP/PLANCK
- Demonstrate superior performance for semantic segmentation of traffic fisheye images, both for synthetic and real automotive datasets



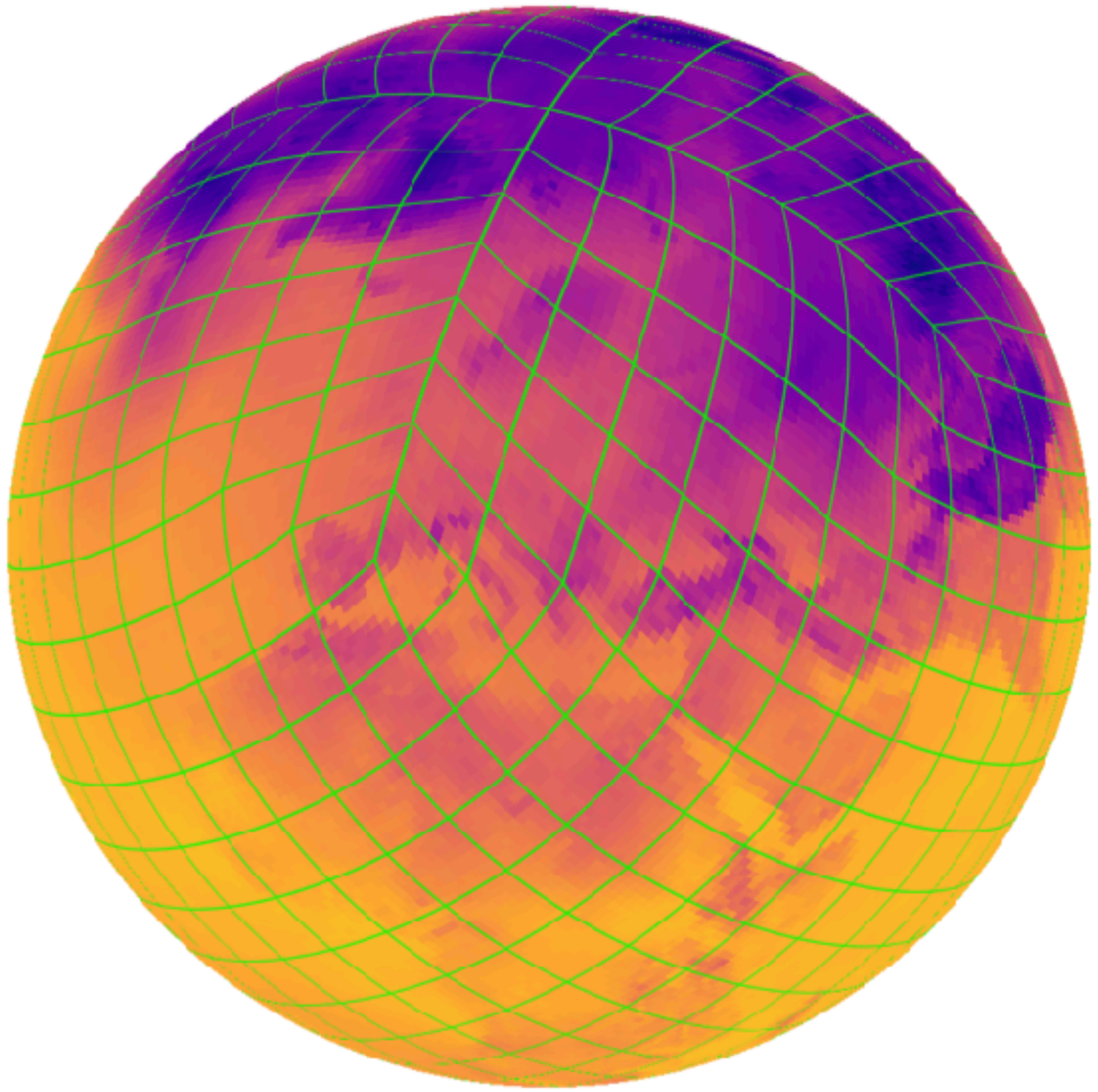
PEAR: Equal Area Weather Forecasting on the Sphere

Hampus Linander [1,2] **Christoffer Petersson** [2,3] **Daniel Persson** [2] **Jan E. Gerken** [2]

[1] VERSES AI
Los Angeles, USA

[2] Department of Mathematical Sciences
Chalmers University of Technology
University of Gothenburg, Sweden

[3] Zenseact
Gothenburg, Sweden



Predicted surface level temperature

Current AI weather models still work on an **equiangular spherical grid** (Driscoll–Healy), which over-resolves the poles relative to the equator.
Why not forecast directly on an equal-area discretization of the sphere?

PEAR ("Pangu Equal ARea"), is a **transformer-based global weather forecasting model** that operates **natively on HEALPix features**

PEAR also **outperforms Pangu-Large** for forecast horizons of **5 days and beyond**

Learning Chern Numbers of Multiband Topological Insulators with Gauge Equivariant Neural Networks

NEURAL INFORMATION PROCESSING SYSTEMS

Longde Huang, Oleksandr Balabanov, Hampus Linander, Mats Granath
Daniel Persson, Jan E. Gerken

- **Topological Insulators:** Materials that behave as insulators in their bulk but allow current to flow along their boundaries, due to underlying topological invariants (Chern numbers)
- **Describe using lattice gauge theory:** Underlying U(N) local symmetry at each site
- We extend the **gauge equivariant networks** of Favoni et al and apply them to the problem of predicting Chern numbers of multi-band systems

Topological Insulators & the Chern Number

Topological insulators

- **Bulk insulating**, robust conducting edge states.
- Topologically protected against deformations/disorder.
- Applications: spintronics, topological photonics, quantum computing.

The Chern number

- Integer invariant $C \in \mathbb{Z}$ classifying band topology.
- Quantised Hall conductance: $\sigma_{xy} = C\, e^2/h$.

Why is prediction hard?

- C computed from **Berry curvature** — requires a **gauge choice** for Bloch states.
- For N filled bands: *local* $\mathrm{U}(N)$ rotation at every k-point.
- Total symmetry group: $\mathrm{U}(N)^{N_{\mathrm{site}}}$ — *exponentially* large.

The challenge

Standard NNs cannot handle this local symmetry.
Previous ML limited to $N{=}1$ (abelian $\mathrm{U}(1)$).

Learning Chern Numbers of Topological Insulators

The problem

Multiband topological insulators with N filled bands.

Chern number (topological invariant):

$$C = \frac{1}{2\pi i}\int_{\mathrm{BZ}} \mathrm{Tr}\big[\mathcal{F}(k)\big]\, d^2k \in \mathbb{Z}$$

Discretised on a lattice via **Wilson loops** $W_k \in \mathrm{U}(N)$ built from link variables (overlaps of Bloch states).

Gauge symmetry (local, at every site):

$$W_k \longrightarrow \Omega_k^\dagger W_k \Omega_k, \qquad \Omega_k \in \mathrm{U}(N)$$

Why gauge equivariance?

Total symmetry group: $\mathrm{U}(N)^{N_{\mathrm{site}}}$

$\Rightarrow$ *exponentially* larger than any global symmetry.

Standard CNNs fail for $N > 1$ band.

Key idea

Adapt **Lattice Gauge Equivariant CNNs** (Favoni et al., from lattice QCD) to condensed matter:

$$f(\Omega^\dagger W \Omega) = f(W) \quad \text{by construction.}$$

Architecture & Key Results

GEBLNet — purely local layers

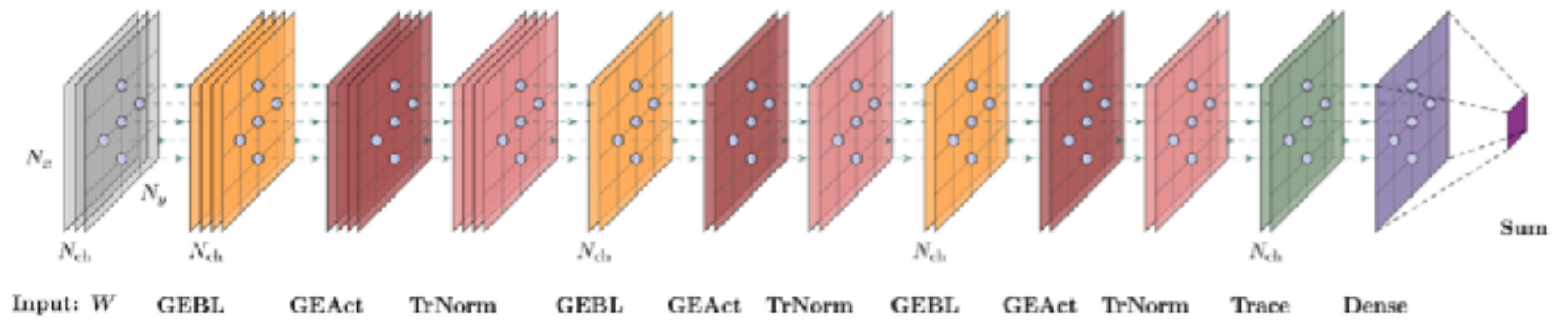
- **GEBL** (bilinear): $W'^{\gamma} = \alpha^{\gamma}_{\mu\nu} W^{\mu} W^{\nu}$
- **GEAct** (nonlinearity): $\sigma(\mathrm{Tr}\, W) \cdot W$
- **TrNorm** (new): normalise by $|\langle \mathrm{Tr}\, W \rangle|$
- **Trace** $\rightarrow$ gauge-invariant scalars $\rightarrow \sum_k$

Universal approximation theorem

Class functions on $\mathrm{U}(N)$ are spanned by symmetric polynomials in eigenvalues (Peter–Weyl). GEBLNet generates $\mathrm{Tr}\, g, \mathrm{Tr}\, g^2, \ldots, \mathrm{Tr}\, g^M$
$\Rightarrow$ can approximate *any* gauge invariant.

Key results

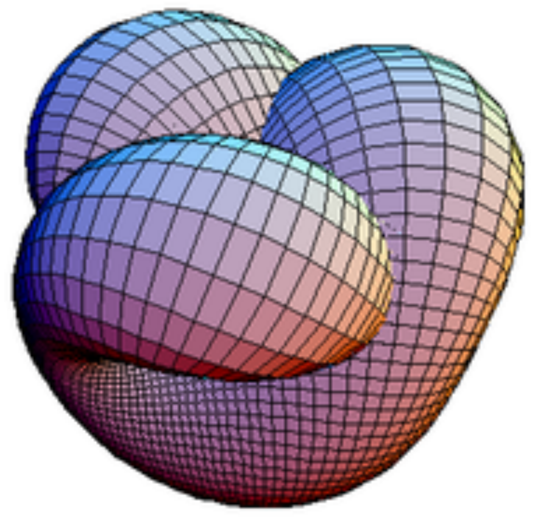
- **Up to $N = 7$ bands** — previous methods: $N = 1$ only. ~96% accuracy (N=4, 5×5 grid).
- **Generalises from $C = 0$** — trained on trivial topology, predicts non-trivial (~94%).
- **Scales to large lattices** — local architecture; works on unseen grid sizes.
- **4D Chern numbers** — learns C_2 on 4D Brillouin zones ($\mathrm{MAE} \approx 0.25$).



Conclusions

	Physics	Mathematics	Neural Networks
Space	Spacetime $\mathcal{M}$	Homog. space G/H	Input domain
Data	Fields (scalars, spinors, . . .)	Sections of $G \times_\rho V_\rho$	Feature maps
Symmetry	Gauge group	Structure group H	Stabiliser subgroup
Redundancy	Gauge choice	Choice of section $s\colon G/H \to G$	Local coordinates
Coord.-free	Covariant formulation	$\mathrm{Ind}_H^G \rho$	Mackey functions
Dynamics	Yang–Mills, Einstein eqs.	Equivariant maps	Network layers
Linear	Free field propagator	Convolution $\hat{\kappa}(g^{-1}g')$	G-CNN
Non-linear	Interactions	$\hat{\omega}(g^{-1}f, g')$	Attention, message passing

Gauge equivariant neural networks



[Cheng, Anagiannis, Weiler, de Haan, Cohen, Welling]

[Gerken, Carlsson, Aronsson, Linander, Ohlsson, Petersson, **D.P.**]

CNNs on arbitrary manifolds
require local equivariance

covariance w. r. t.
gauge transformations
(general coordinate transformations)

gauge equivariant
feature maps

Fields
Sections of vector bundles
(frame bundles)

“elementary feature types”
?

irreducible representations of G
elementary particles
(scalars, vectors, spinors…)

Are these the seeds of a deeper relation between neural networks and gauge theory?

Future work

➡ **Explore connections with neuroalgebraic geometry**

Joint with Magdalena & Yacoub (arXiv:2603.29566)

➡ **Applications to climate data**

Joint with Gerken, Linander & Petersson

➡ **Equivariance in weather & climate data?**

➡ **Geometric deep learning for medical images**

Collaboration with Sahlgrenska University Hospital


TACK!