Equivariant PDE-based Neural Networks:

Strong gometric priors for deep learning with images

Bart Smets
TU Eindhoven

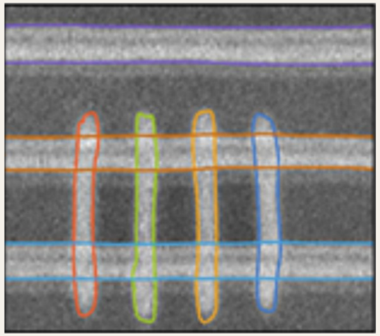
Outline

- PDE-based Group equivariant Neural Networks (PDE-G-CNN)
  - Motivation
  - Construction
  - Experiments
  - LieTorch software package

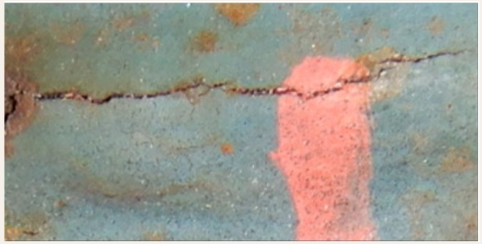
with Remco Duits, Jim Portegies, Erik Bekkers (UvA), Gijs Bellaard, Gautam Pai (UTwente), Daan Bon

Motivating applications

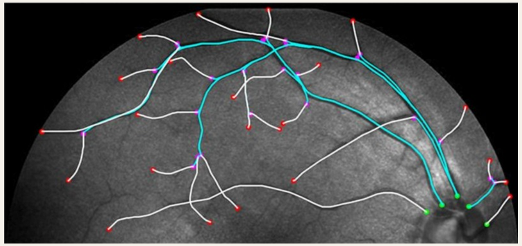
SEM image analysis for semiconductor manufacturing



Crack detection for automated bridge inspection

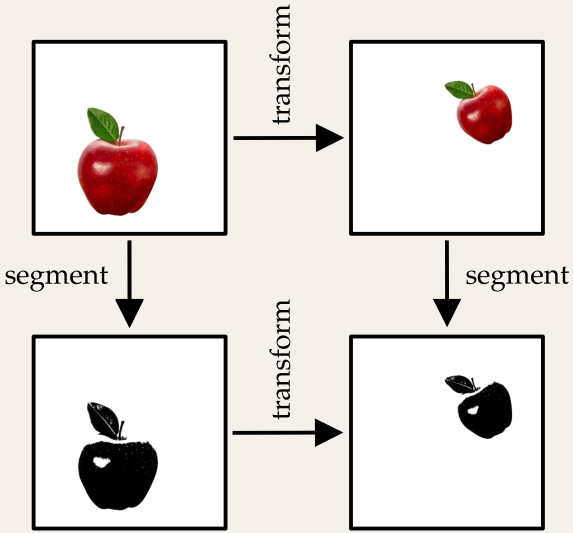


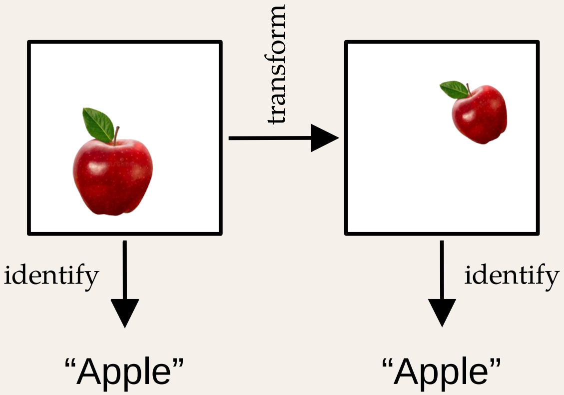
Retinal vessel segmentation & tracking



Why geometry?

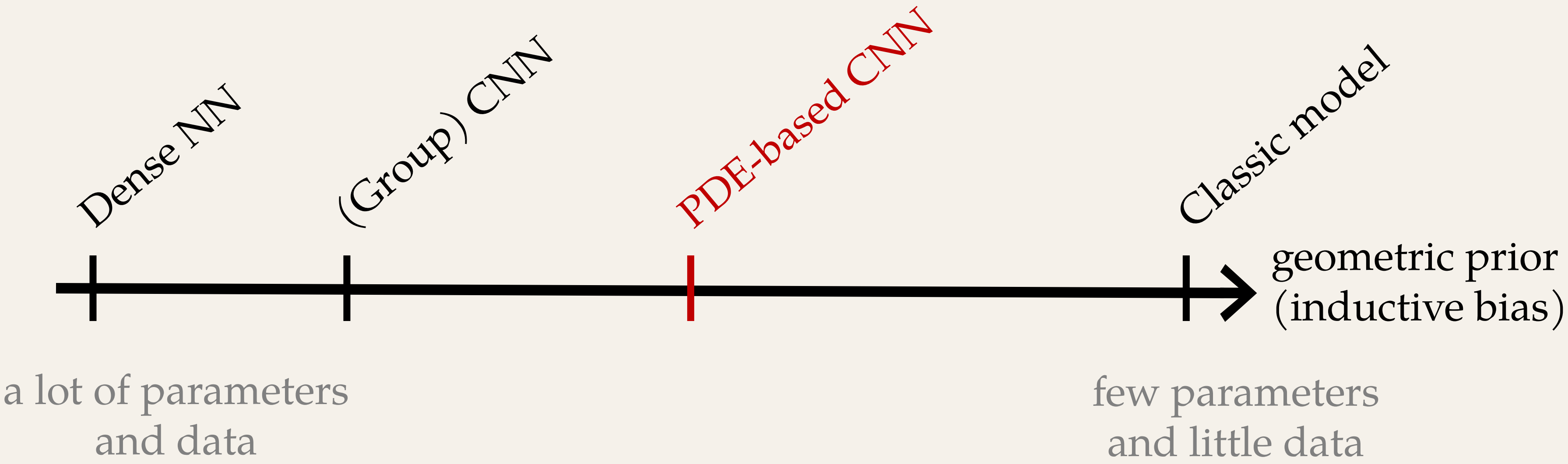
1. Equivariance & invariance





2. Little available data
3. Interpretability

Geometric prior



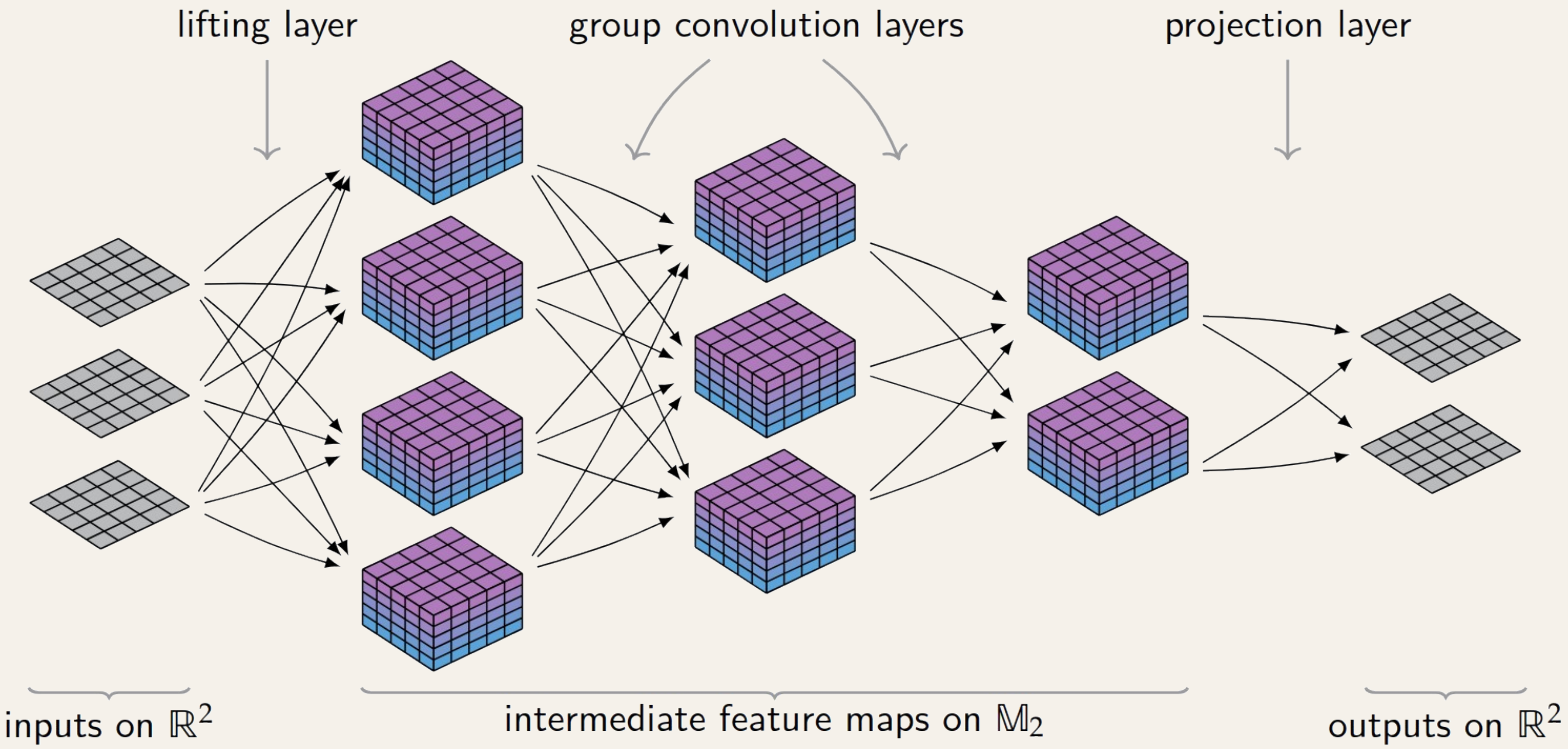
Starting point: Group CNN (Cohen, Welling 2016)

- neural network + translation equivariance = CNN
  G = T(2) (translations in 2 dimensions)

- neural network + roto-translation equivariance = SE(2)-CNN
  G = SE(2) (rotations and translation in 2 dimensions)

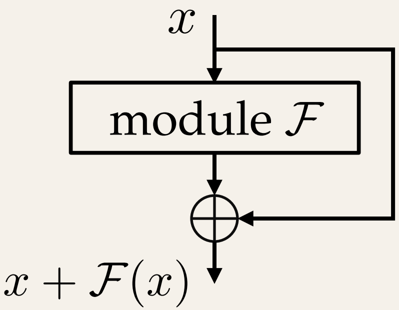
G can in *principle* be any group of transformations

Lift-Convolve-Project



CNNs share similarities with PDE solvers

Heat equation	Analytic solution	CNN
$\begin{cases} \frac{\partial f}{\partial t} = \nabla^2 f \\ f(0) = f_0 \end{cases}$	$f(t) = G_t * f_0$	$f_{out} = \sigma\left(K * f_{in}\right)$

Additive residual connection	Forward Euler discretization
	$\dot{x}(t) = \mathcal{F}(x(t))$ is solved as $x(t + \Delta t) = x(t) + \Delta t\, \mathcal{F}(x(t))$

Idea

Take a PDE from the image processing world and put it in a neural network

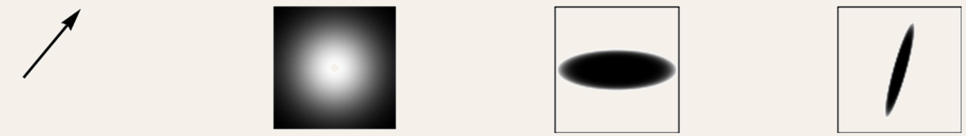
$$\begin{cases} \frac{\partial u}{\partial t} = -\boldsymbol{c}u - \left(-\Delta_{\mathcal{G}_1}\right)^{\nu} u + \frac{1}{\alpha} \left\| du \right\|^{\alpha}_{\mathcal{G}_2^+} - \frac{1}{\alpha} \left\| du \right\|^{\alpha}_{\mathcal{G}_2^-} \\ u(0) = u_0 \end{cases}$$

trainable: convection vector field $\boldsymbol{c}$ & metrics $\mathcal{G}_1, \mathcal{G}_2^+, \mathcal{G}_2^-$

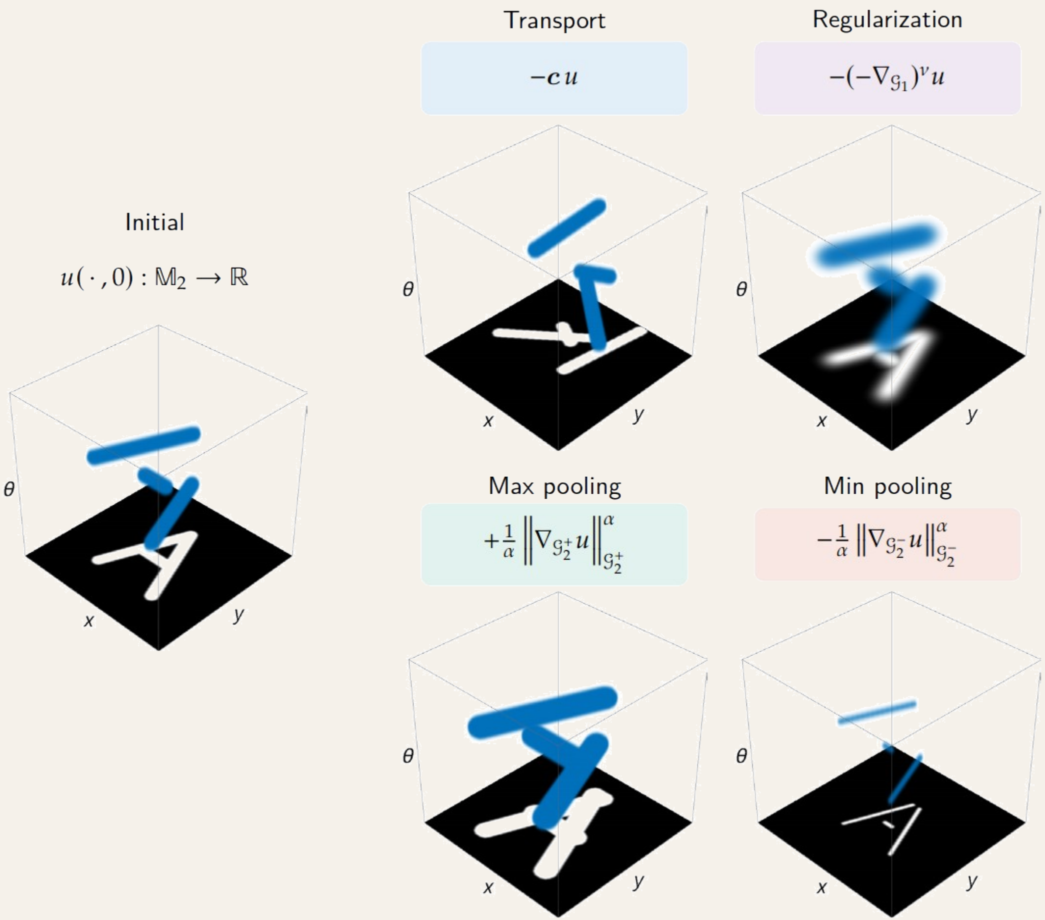
hyper parameter $0 < \nu \leq 1, \alpha > 1$

Interpretation

$\mathbb{R}^2$	Transport	Regularization	Max pooling	Min pooling
PDE Term:	$-\boldsymbol{c}u$	$-\left(-\Delta_{\mathcal{G}_1}\right)^{\nu} u$	$+\frac{1}{\alpha}\left\|du\right\|^{\alpha}_{\mathcal{G}_2^+}$	$-\frac{1}{\alpha}\left\|du\right\|^{\alpha}_{\mathcal{G}_2^-}$
Parameters:	transport vector $\begin{pmatrix}100\\120\end{pmatrix}$	Riemannian metric tensor $\begin{pmatrix}1&0\\0&1\end{pmatrix}$	Riemannian metric tensor $\begin{pmatrix}9&0\\0&1\end{pmatrix}$	Riemannian metric tensor $\begin{pmatrix}5&11\\11&40\end{pmatrix}$
Operation:	resample with offset	convolution with kernel	dilation with kernel	erosion with kernel

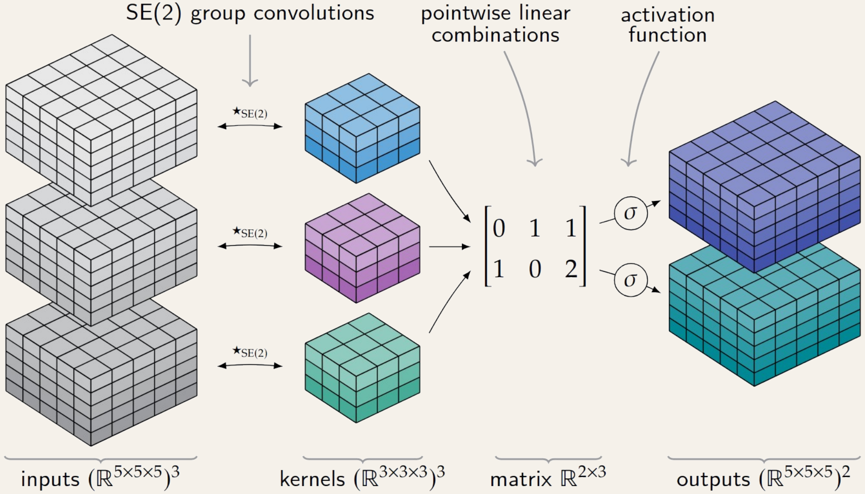


The PDE generalizes to any homogeneous space

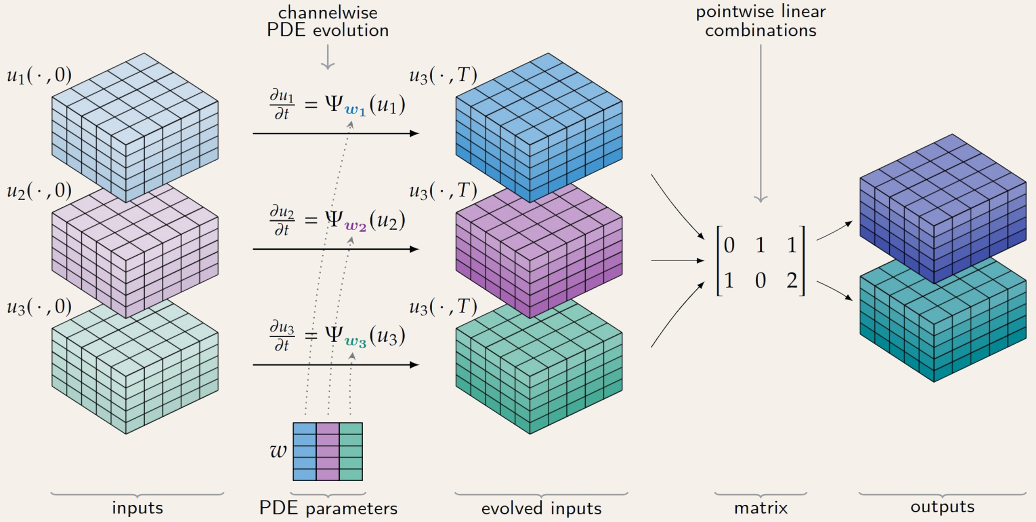


PDE-based layer

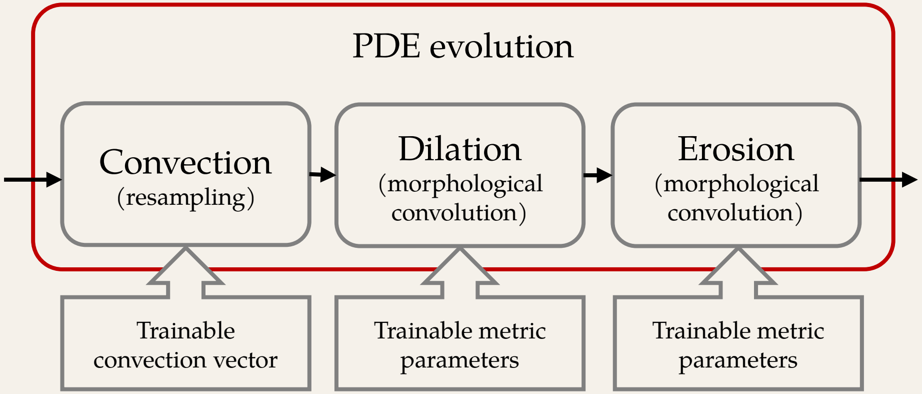
Depthwise CNN layer:



PDE-based layer:



PDE-based layer implementation



Morphological convolution

On $\mathbb{R}^2$ for translation equivariance:

$$(k \square f)(x) = \sup_{y \in \mathbb{R}^2} k(x-y) + f(y)$$

- dilation: $k \square f$
- erosion: $-(-k \square - f)$
- generalizes to the group setting

Morphological Group Convolution

DEFINITION 6.6 (Morphological convolution) Let G/H be a homogeneous space of a Lie group G, $f \in L^\infty(G/H)$ and let $k : G/H \to \mathbb{R} \cup \{\infty\}$ be proper (i.e. not everywhere ∞) then we define the *morphological convolution* of f with kernel k as:

$$(k \,\square_{G/H}\, f)(p) := \inf_{q \in G/H} \left\{ (g_p \cdot k)(q) + f(q) \right\} \tag{6.29}$$

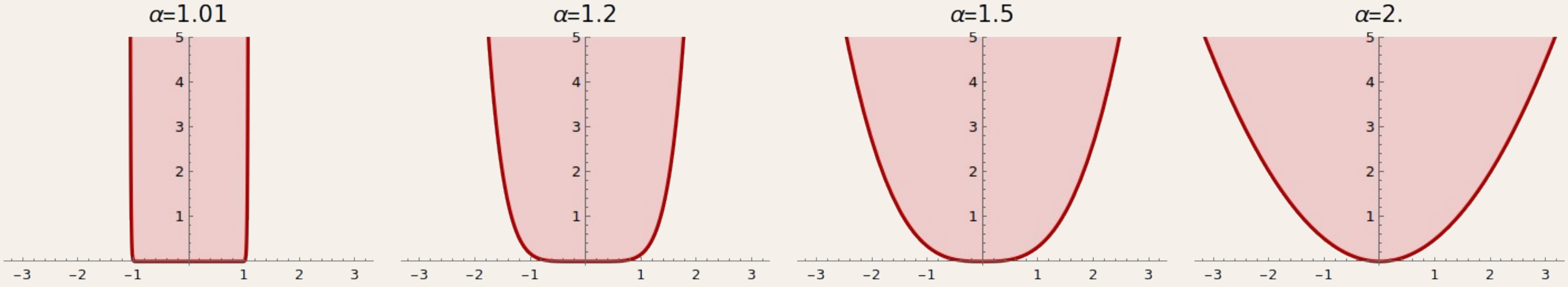
where $g_p \in p$ is freely chosen for all $p \in G/H$. For this expression to be well-defined we require that the kernel k satisfies the compatibility condition:

$$\forall h \in H : h \cdot k = k. \tag{6.30}$$

Morphological kernel

$$k_t^\alpha(p) = \frac{t}{\beta}\left(\frac{d_{\mathcal{G}}(p_0, p)}{t}\right)^\beta$$

- α,β controls sharpness $\frac{1}{\alpha} + \frac{1}{\beta} = 1$
- the metric contains the trainable parameters



Problem

Recomputing $p \mapsto d_{\mathcal{G}}(p_0, p)$ the whole time is expensive, instead:

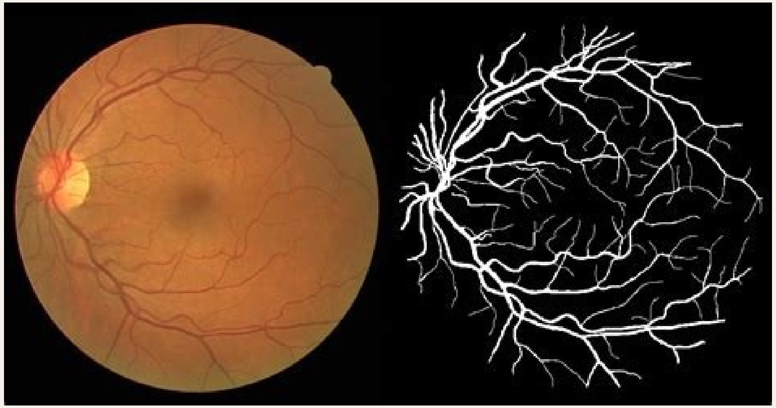
$$\rho_{\mathcal{G}}(p) := \inf_{g \in p} \left\| \log_G g \right\|_{\mathcal{G}} \approx d_{\mathcal{G}}(p_0, p)$$

where we interpret $p \in G/H$ and so $p \subset G$.

In the principal case this is just $\rho_{\mathcal{G}}(g) := \left\| \log_G g \right\|_{\mathcal{G}}$.

Essentially: estimate distance by the shortest exponential curve.

Retinal vessel segmentation

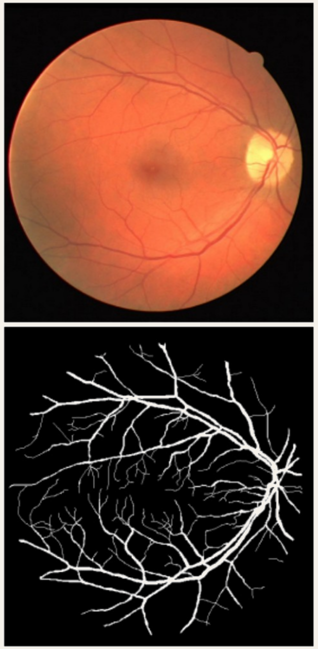


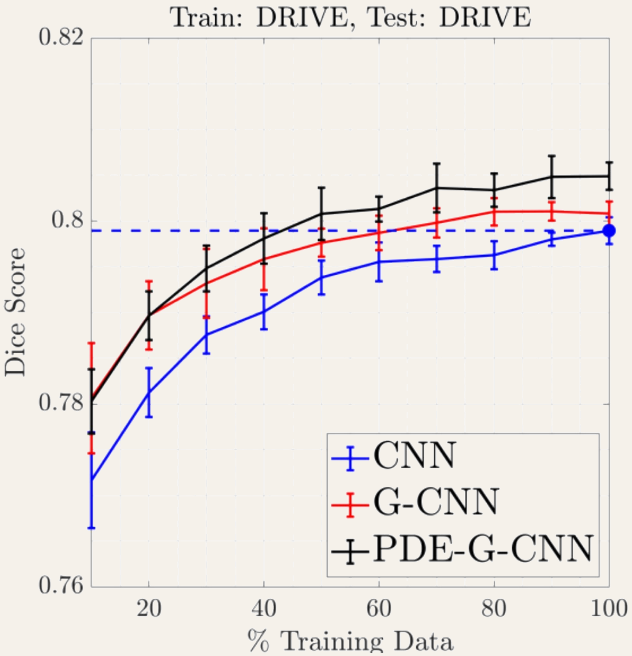
Model (transformation group)	Parameters	DICE score %
CNN 6 T(2)	47,352	80.58 ±0.17
G-CNN 6 SE(2)	39,258	80.85 ±0.22
CDE-PDE-CNN 6 SE(2)	4,128	81.15 ±0.18
CNN 12 T(2)	129,432	81.89 ±0.05
G-CNN 12 SE(2)	114,378	81.92 ±0.12
CDE-PDE-CNN 12 SE(2)	3,768	82.20 ±0.07

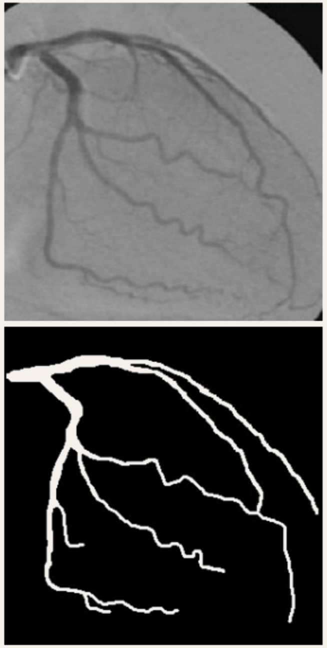
RotNIST digit classification

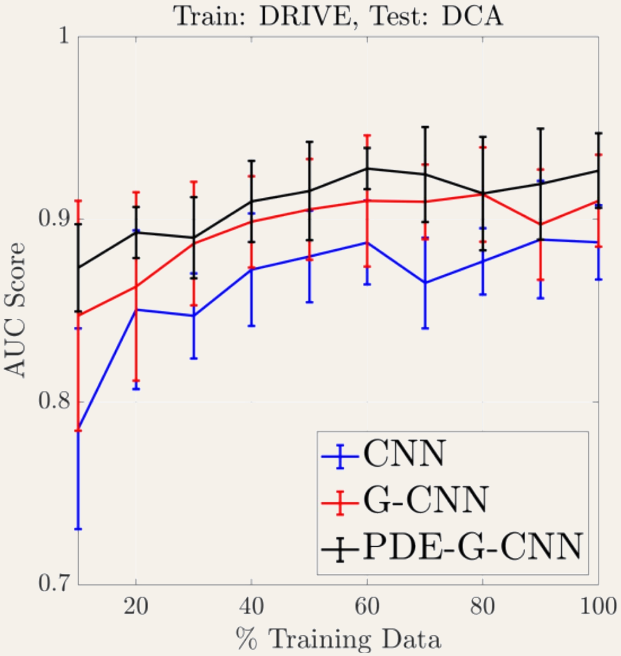
Model (transformation group)	Parameters	Accuracy %
LeNet 5 $T(2)$	44,426	97.42 $\pm$0.66
G-CNN Classifier 4 $SE(2)$	12,700	98.86 $\pm$0.21
CDE-PDE-CNN Classifier 4 $SE(2)$	2,542	98.90 $\pm$0.18
CDE-PDE-CNN Classifier 4 $T(2)$	2,518	98.96 $\pm$0.07

Limited training data & generalization









Strengths & weaknesses

+ fewer parameters
+ geometric interpretability
+ need less data
+ equivariance

- computational complexity / memory capacity in higher dimensions
- strong geometric prior can be counter productive

PyTorch extension

LieTorch

Geometric Machine Learning and Lie Analysis

```
> conda install lietorch::lietorch
> pip install lietorch
> docker pull ghcr.io/bmnsmets/lietorch-ssh
```

gitlab.com/bsmetsjr/lietorch

Thank you

[1] Smets B.M.N., Portegies J., Bekkers E.J., Duits R. **PDE-Based Group Equivariant Convolutional Neural Networks**. J Math Imaging Vis 65, 209–239 (2022). https://doi.org/10.1007/s10851-022-01114-x

[2] Bellaard G., Bon D.L.J., Pai G., Smets B.M.N., Duits R. **Analysis of (sub-)Riemannian PDE-G-CNNs**. J Math Imaging Vis (2023). https://doi.org/10.1007/s10851-023-01147-w

[3] Pai, G. Bellaard G., Smets B.M.N., Duits R. (2023). **Functional Properties of PDE-based Group Equivariant Convolutional Neural Networks**. In: Nielsen, F., Barbaresco, F. (eds) Geometric Science of Information. GSI 2023. Lecture Notes in Computer Science, vol 14071. Springer, Cham. https://doi.org/10.1007/978-3-031-38271-0_7