4TU.AMI Event
Mathematics of Deep Learning
Delft, 2019-11-05

Group Equivariant CNNs beyond Roto-Translations:
B-Spline CNNs on Lie groups

Erik J Bekkers

Department of Mathematics Computer Science,

Centre for Analysis, Scientific computing and Applications (CASA)

Eindhoven University of Technology

Starting next month at

UvA

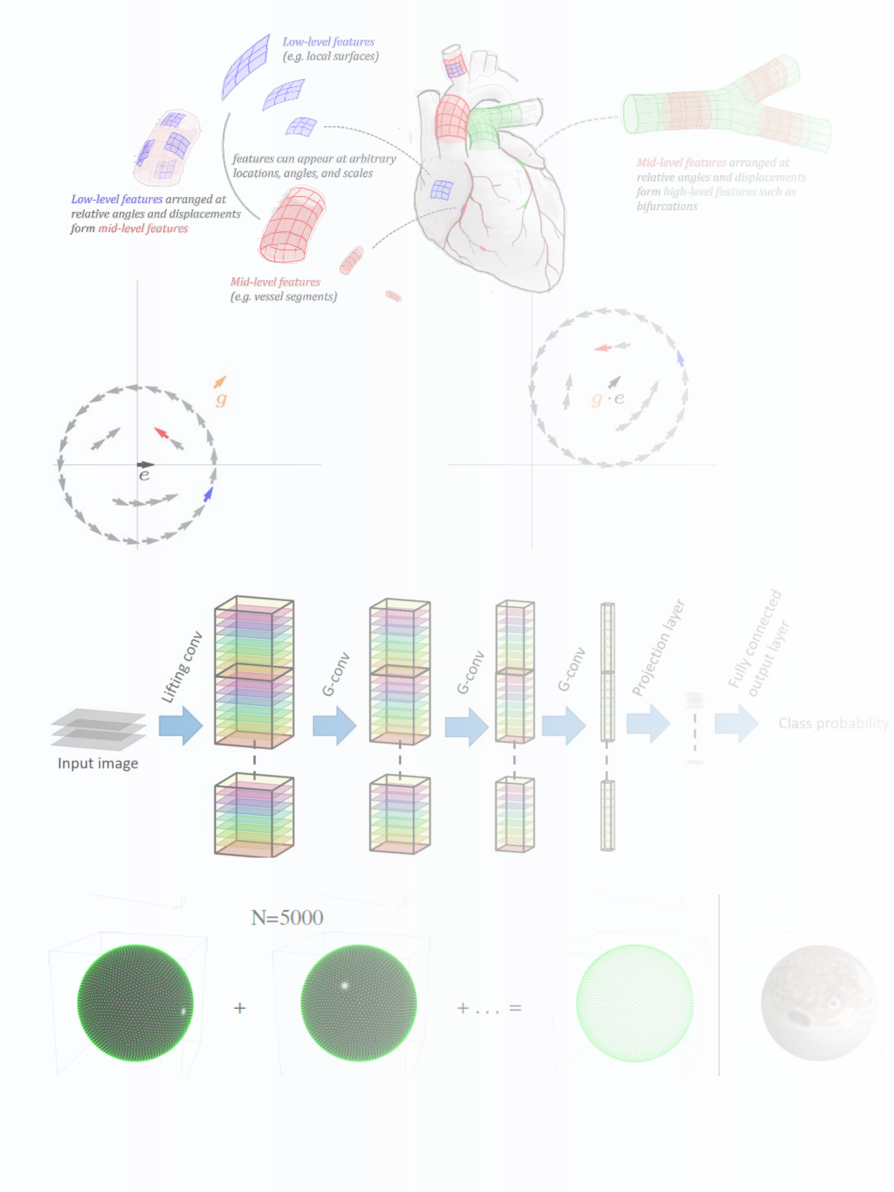
AMLAB
Amsterdam
Machine Learning Lab

Amsterdam Machine Learning Lab
Informatics Institute
University of Amsterdam

TU/e EINDHOVEN UNIVERSITY OF TECHNOLOGY

Presentation outline

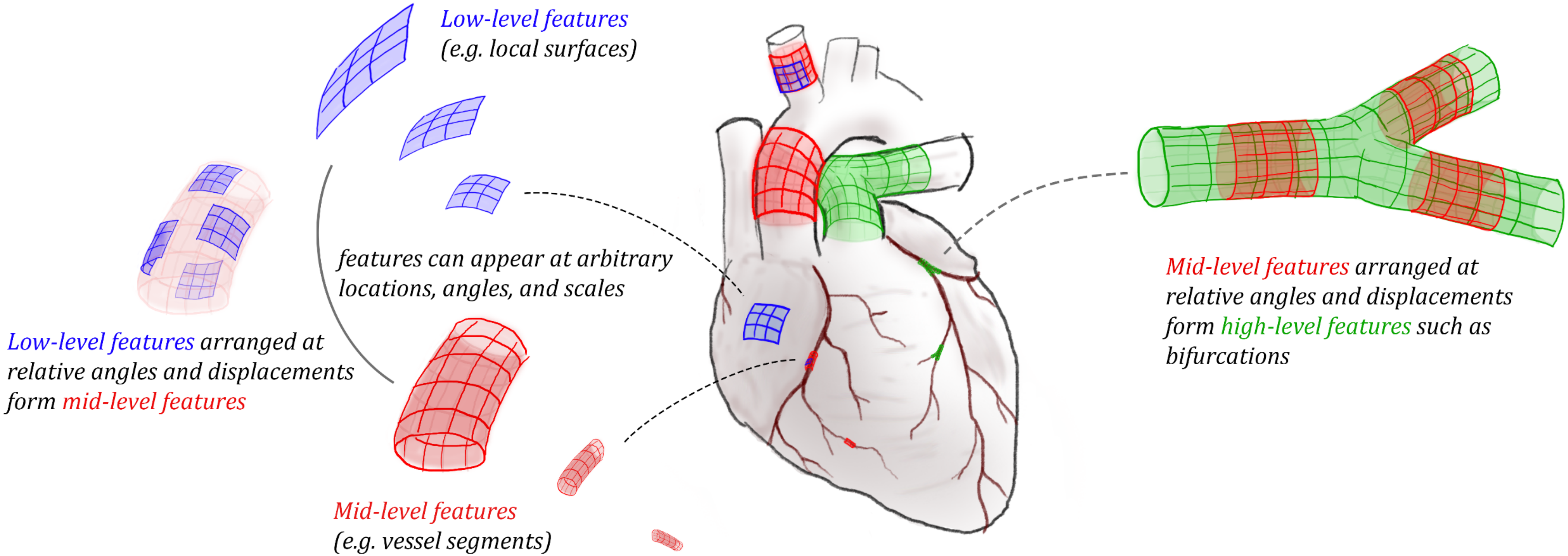
- Motivation
- Group theory (preliminaries)
- G-CNNs
  - Construction and intuition
  - Theorem: NN-layers with equivariance constraints => G-CNNs
- B-Spline based G-CNNs: G-CNNs on arbitrary Lie groups
- Conclusion



Motivation

Recognition by components

Biederman, 1987



Low-level features
(e.g. local surfaces)

features can appear at arbitrary locations, angles, and scales

Low-level features arranged at relative angles and displacements form mid-level features

Mid-level features
(e.g. vessel segments)

Mid-level features arranged at relative angles and displacements form high-level features such as bifurcations

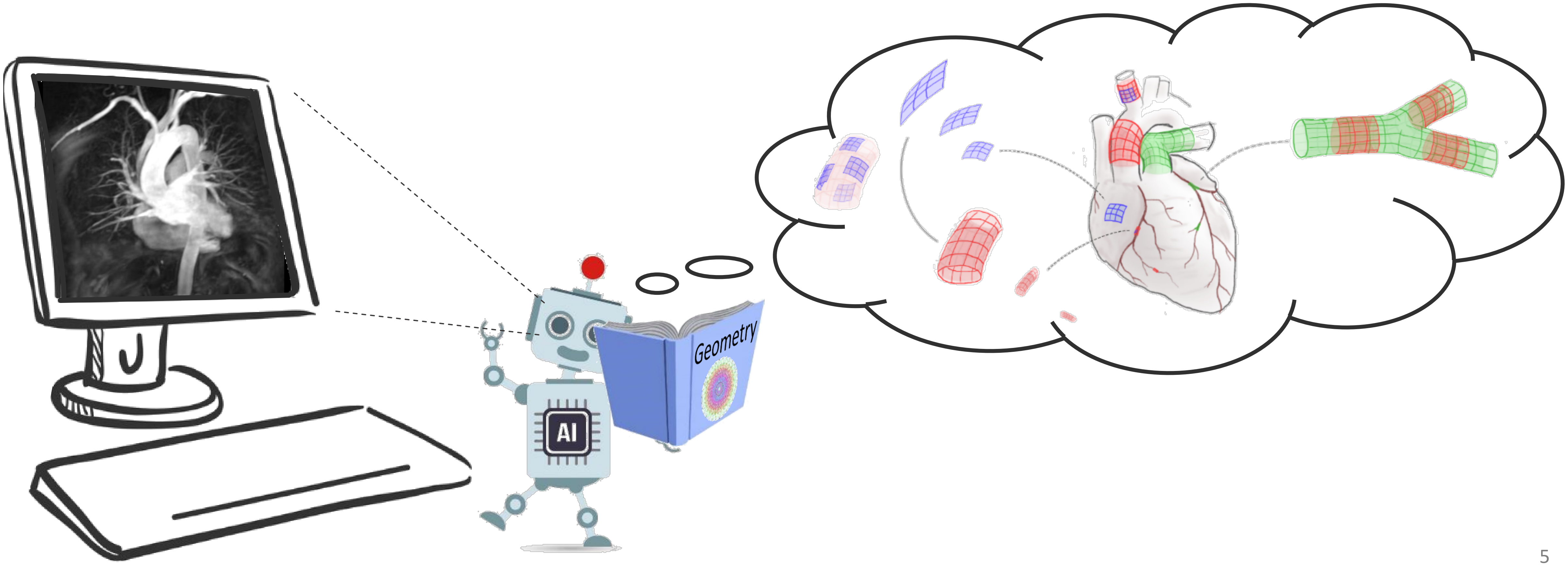
Group theory: Symmetries and relative information processing

Such reasoning motivates
related work on capsule nets

Hinton, Krizhevsky & Wang, 2011
Lenssen, Fey & Libuschewski, 2018

Aim: Build AI systems that are equipped with geometric understanding

- Do not have to learn geometric structure and relations (equivariance)
- Are data-efficient by exploiting symmetries (no need for geometric data augmentation)
- High representation power by recognition by components (capsule net view point)



Group theory (preliminaries)

(Symmetry) Groups

A group $(G, \cdot)$ is a set of elements G equipped with a group product $\cdot$, a binary operator, that satisfies the following four axioms:

- *Closure*: Given two elements g and h of G, the product $(g \cdot h)$ is also in G.
- *Associativity*: For $g, h, i \in G$ the product $\cdot$ is associative, i.e., $g \cdot (h \cdot i) = (g \cdot h) \cdot i$.
- *Identity element*: There exists an identity element $e \in G$ such that $e \cdot g = g \cdot e = g$ for any $g \in G$.
- *Inverse element*: For each $g \in G$ there exists an inverse element $g^{-1} \in G$ such that $g^{-1} \cdot g = g \cdot g^{-1} = e$.

The translation group $(\mathbb{R}^2, +)$

The translation group consists of all possible translations in $\mathbb{R}^2$ and is equipped with the group product and group inverse:

$$\begin{aligned} g \cdot g' &= \mathbf{x} + \mathbf{x}' \\ g^{-1} &= -\mathbf{x}. \end{aligned}$$

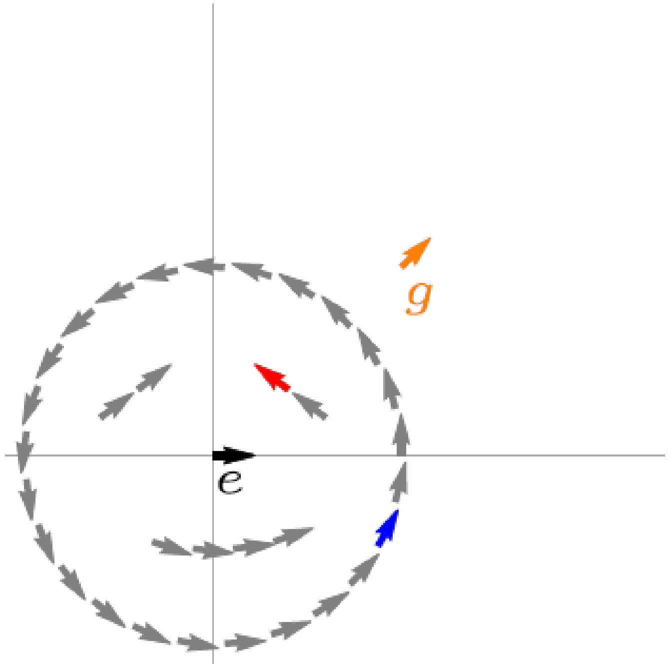
with $g = \mathbf{x}, g' = \mathbf{x}' \in \mathbb{R}^2$.

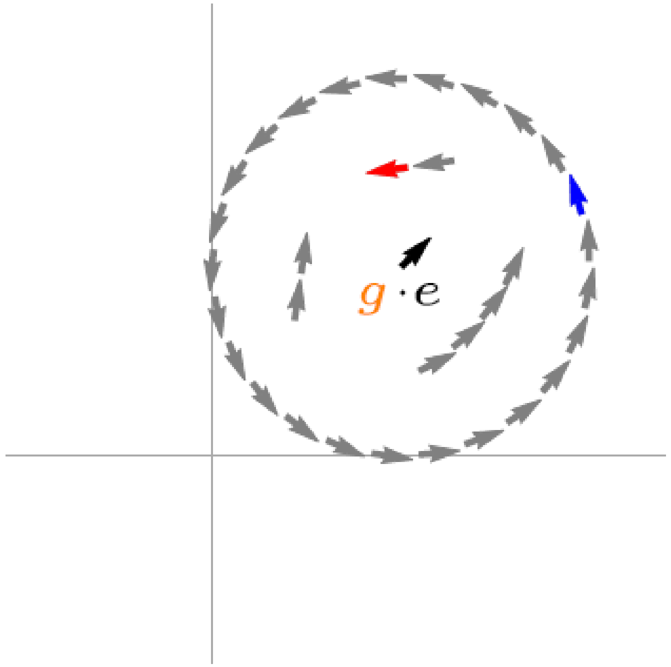
The roto-translation group $SE(2)$

***S**pecial **E**uclidean Motion group*

The group $SE(2)$ consists of the coupled space $\mathbb{R}^2 \rtimes S^1$ of positions (translations) in $\mathbb{R}^2$, and orientations (rotations) S^1, and is equipped with the group product and group inverse:

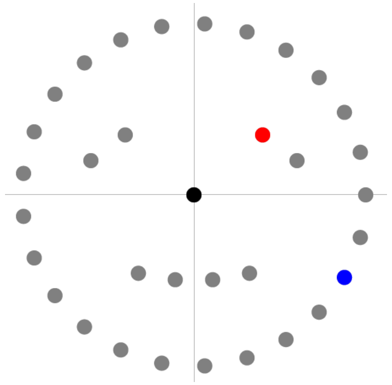
$$\begin{aligned} g \cdot g' &= (\mathbf{x}, \mathbf{R}_\theta) \cdot (\mathbf{x}', \mathbf{R}_{\theta'}) = (\mathbf{R}_\theta \mathbf{x}' + \mathbf{x}, \mathbf{R}_{\theta+\theta'}) \\ g^{-1} &= (-\mathbf{R}_\theta^{-1} \mathbf{x}, \mathbf{R}_\theta^{-1}). \end{aligned}$$





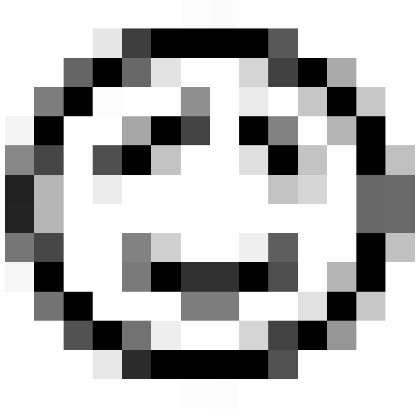
Representations transfer group structure to images

Set of points



$$\{g_1, g_2, \ldots\} \subset (\mathbb{R}^2, +)$$

Convolution kernel



$$f \in \mathbb{L}_2(\mathbb{R}^2)$$

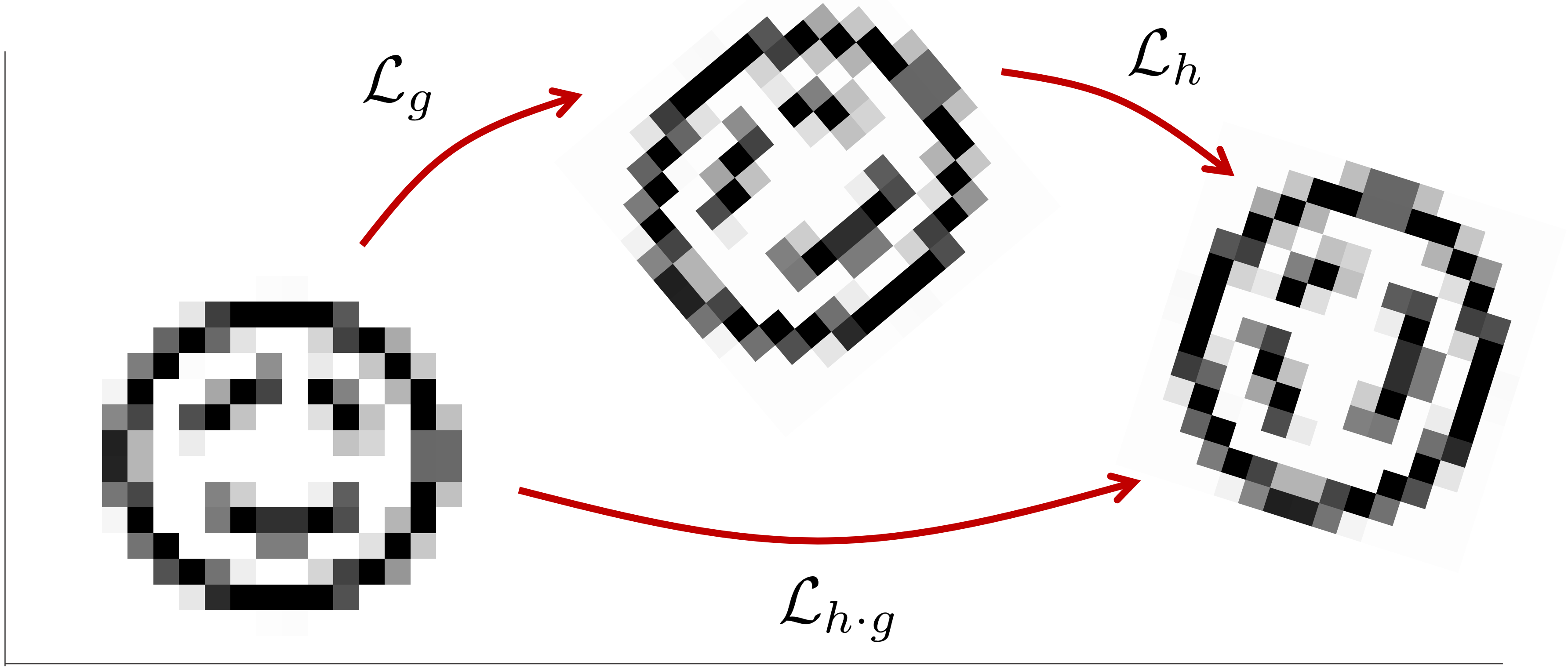
A linear operator $\mathcal{L}_g^{G \to \mathbb{L}_2(X)}$ that transforms functions on some space X and parameterized by group elements $g \in G$ is called a representation of the group if it caries the group structure in the following way

$$\mathcal{L}_h^{G \to \mathbb{L}_2(X)}(\mathcal{L}_g^{G \to \mathbb{L}_2(X)}(f)) = \mathcal{L}_{h \cdot g}^{G \to \mathbb{L}_2(X)}(f)$$

Representations transfer group structure to images

Example:

f 2D image
G the group SE(2)
$\mathcal{L}_g$ roto-translation

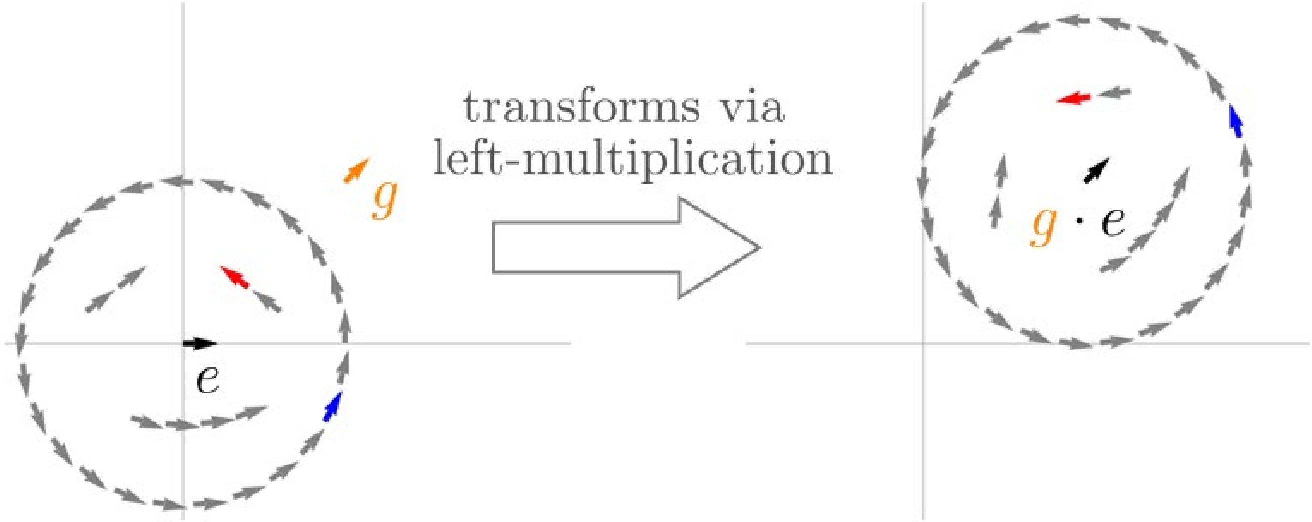


A linear operator $\mathcal{L}_g$ that transforms functions on some space X and parameterized by group elements $g \in G$ is called a representation of the group if it caries the group structure in the following way

$$\mathcal{L}_h(\mathcal{L}_g(f)) = \mathcal{L}_{h \cdot g}(f)$$

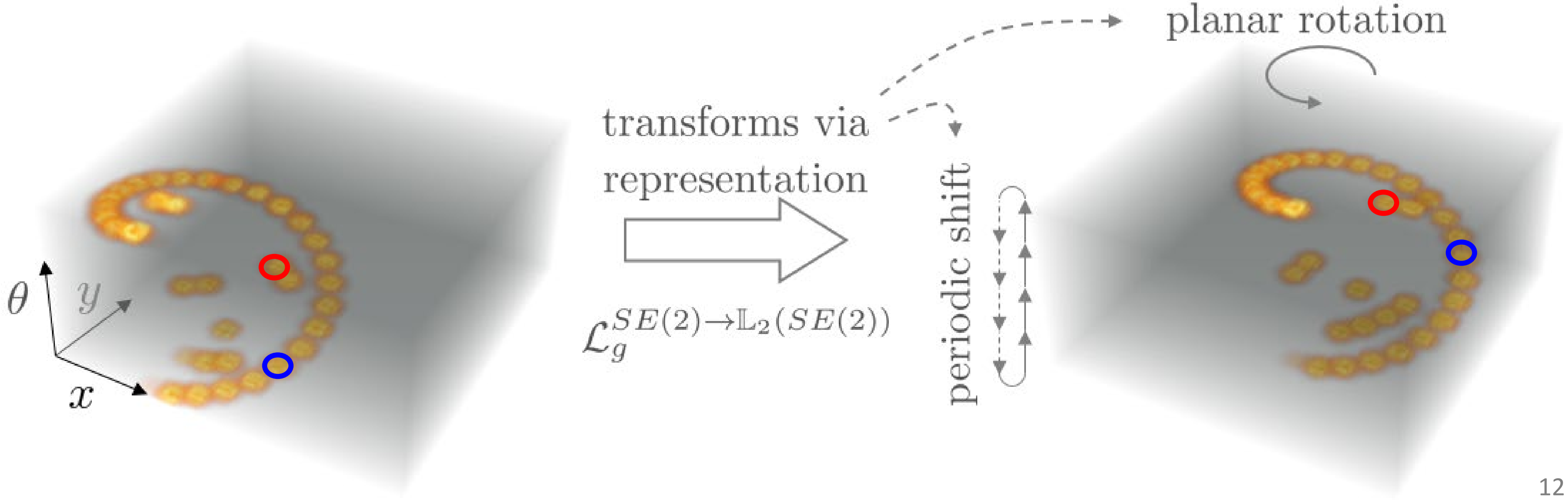
Transforming SE(2) descriptors

Pattern of local orientations:



transforms via left-multiplication

Density on position orientation space:



transforms via representation

$\mathcal{L}_g^{SE(2)\to\mathbb{L}_2(SE(2))}$

CNNs and G-CNNs

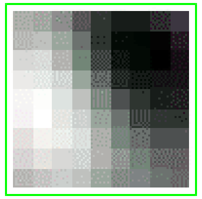
via group representations

Cross-correlations

Cross-correlation:

Representation of the translation group!

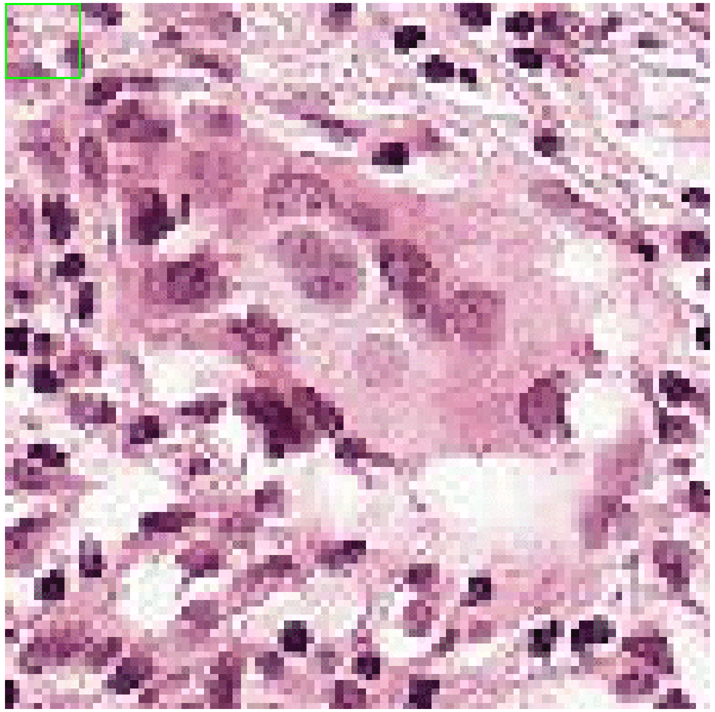
$$(k \star_{\mathbb{R}^2} f)(\mathbf{x}) = \int_{\mathbb{R}^2} k(\mathbf{x}' - \mathbf{x}) f(\mathbf{x}') \mathrm{d}\mathbf{x}'$$



k

2D kernel

$\star_{\mathbb{R}^2}$



f

2D feature map

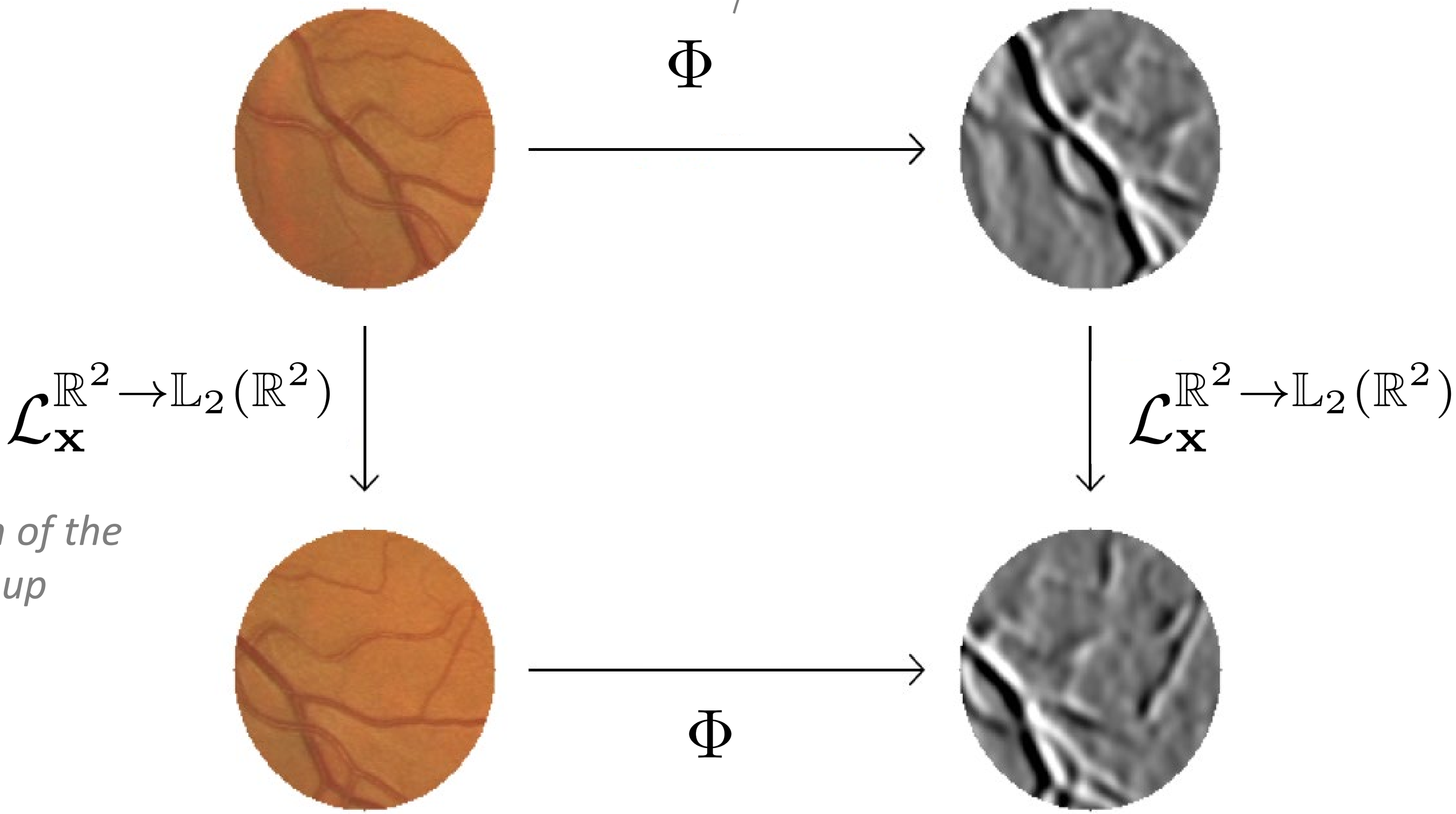
$=$

2D feature map (after ReLU)

Group equivariance

Example: Convolutions are equivariant w.r.t. the translation group

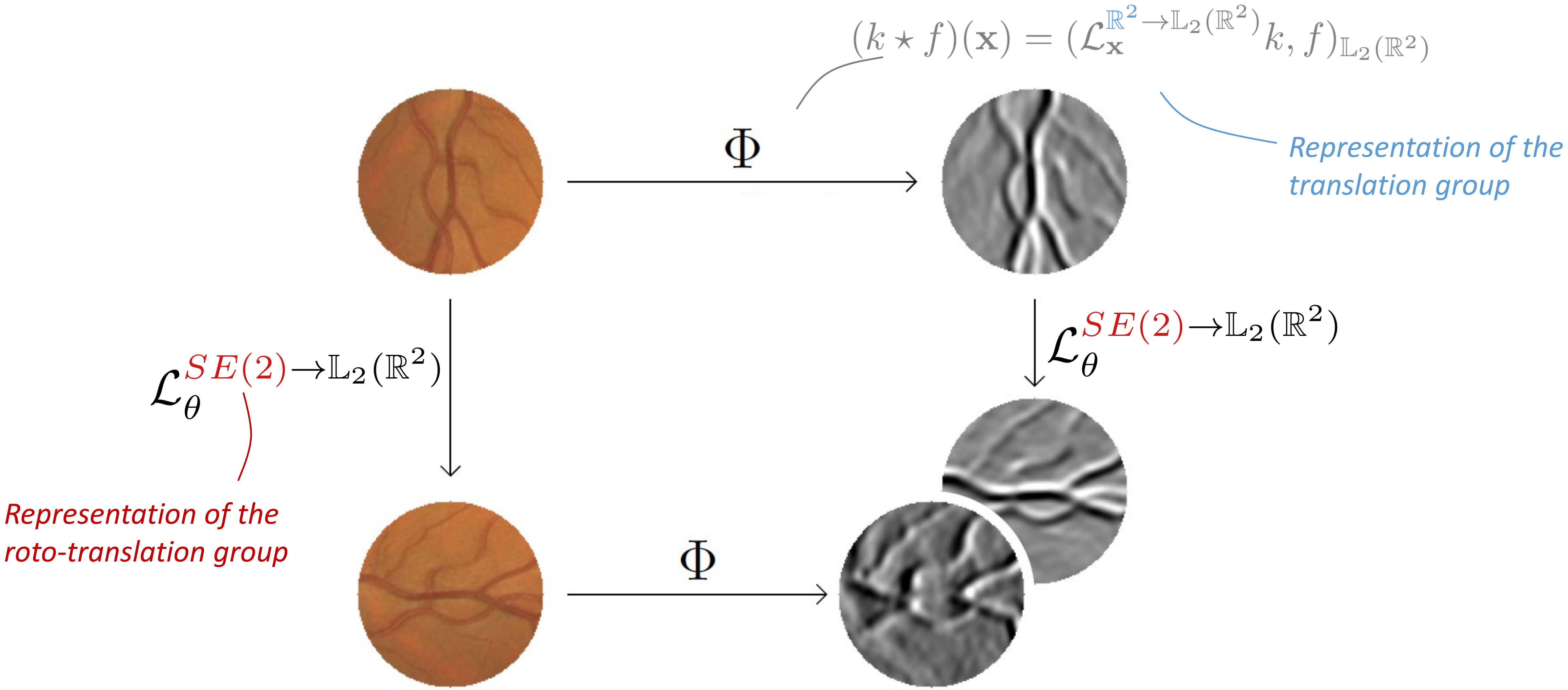
$$(k \star f)(\mathbf{x}) = (\mathcal{L}_{\mathbf{x}}^{\mathbb{R}^2 \to \mathbb{L}_2(\mathbb{R}^2)} k, f)_{\mathbb{L}_2(\mathbb{R}^2)}$$



Representation of the translation group

Group equivariance

Example: Convolutions are generally not equivariant w.r.t. roto-translations.



$$(k \star f)(\mathbf{x}) = (\mathcal{L}_{\mathbf{x}}^{\mathbb{R}^2 \to \mathbb{L}_2(\mathbb{R}^2)} k, f)_{\mathbb{L}_2(\mathbb{R}^2)}$$

Representation of the translation group

Φ

$\mathcal{L}_{\theta}^{SE(2) \to \mathbb{L}_2(\mathbb{R}^2)}$

Representation of the roto-translation group

$\mathcal{L}_{\theta}^{SE(2) \to \mathbb{L}_2(\mathbb{R}^2)}$

Φ

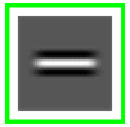
Roto-translation equivariant cross-correlations

Lifting correlations:

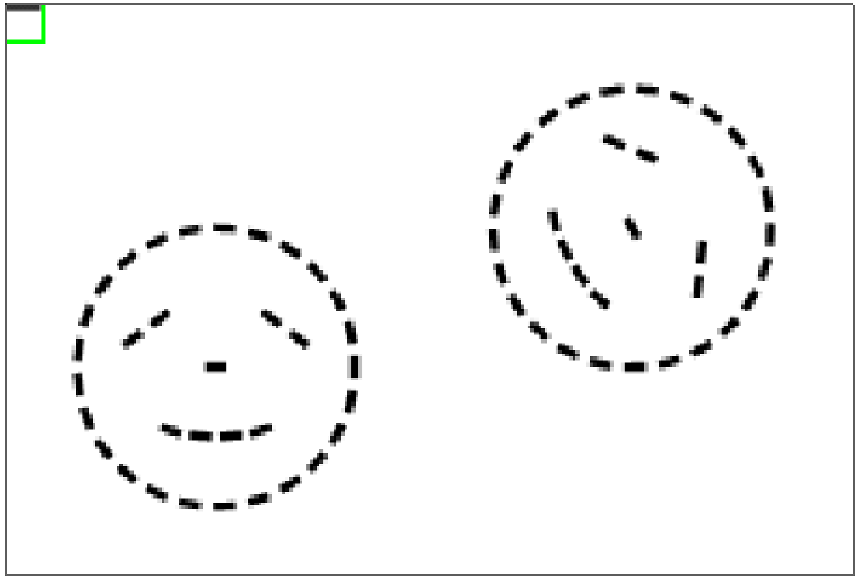
Representation of the roto-translation group!

$$(k \tilde{\star} f)(\mathbf{x}) = (\mathcal{L}_g^{SE(2)\rightarrow\mathbb{L}_2(\mathbb{R}^2)} k, f)_{\mathbb{L}_2(\mathbb{R}^2)} \quad = (\mathcal{L}_{\mathbf{x}}^{\mathbb{R}^2\rightarrow\mathbb{L}_2(\mathbb{R}^2)} \mathcal{L}_\theta^{SO(2)\rightarrow\mathbb{L}_2(\mathbb{R}^2)} k, f)_{\mathbb{L}_2(\mathbb{R}^2)}$$

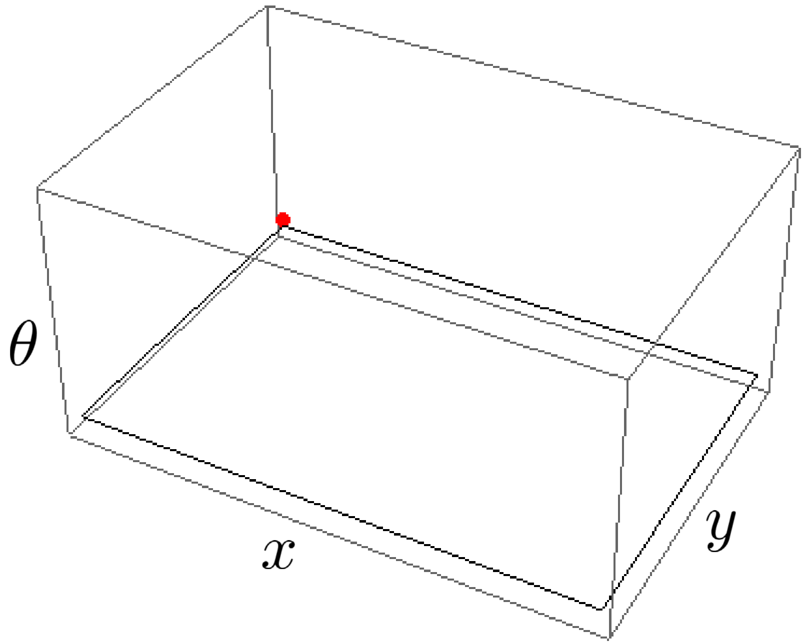
translation rotation



$\star_{\mathbb{R}^2}$



$=$



$\mathcal{L}_\theta^{SO(2)\rightarrow\mathbb{L}_2(\mathbb{R}^2)} k$

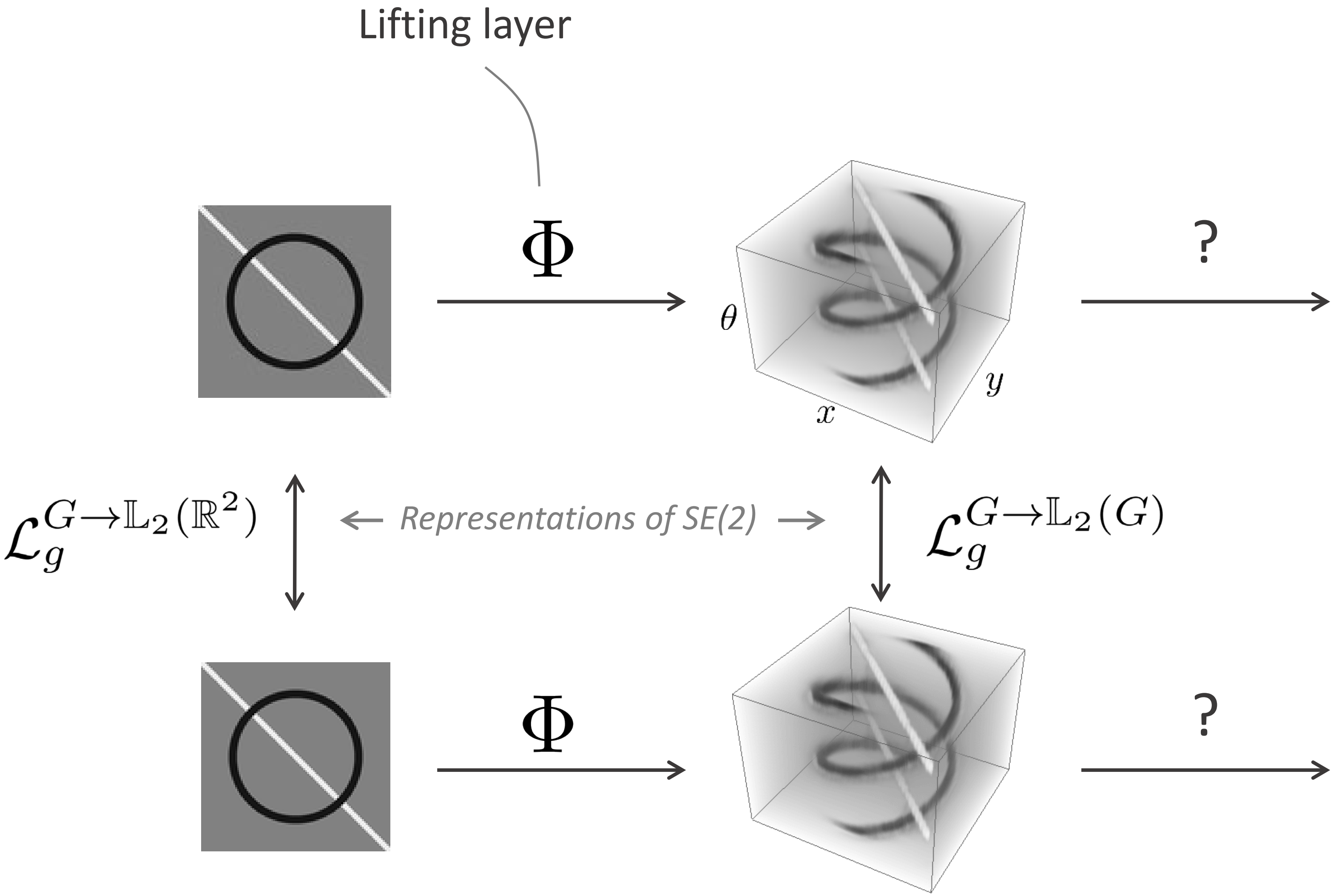
Rotated kernel

f

2D feature map

SE(2) feature map

Roto-translation equivariant cross-correlations



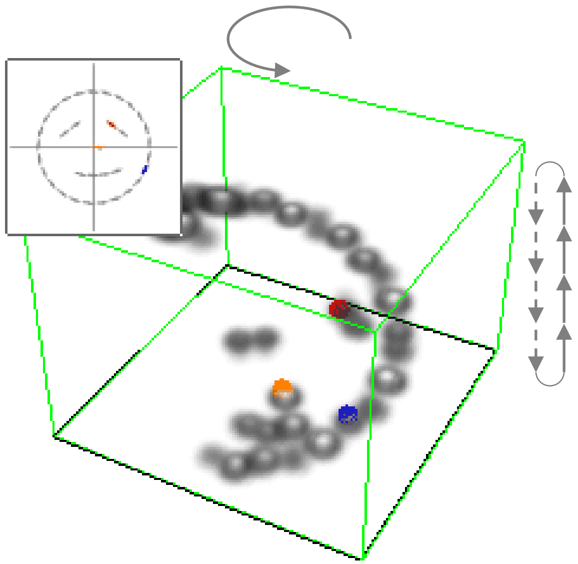
Roto-translation equivariant cross-correlations

Group correlations:

$$(k \star f)(\mathbf{x}) = (\mathcal{L}_g^{SE(2)\to\mathbb{L}_2(SE(2))} k, f)_{\mathbb{L}_2(SE(2))} = (\mathcal{L}_{\mathbf{x}}^{\mathbb{R}^2\to\mathbb{L}_2(SE(2))} \mathcal{L}_{\theta}^{SO(2)\to\mathbb{L}_2(SE(2))} k, f)_{\mathbb{L}_2(SE(2))}$$

translation rotation

planar rotation



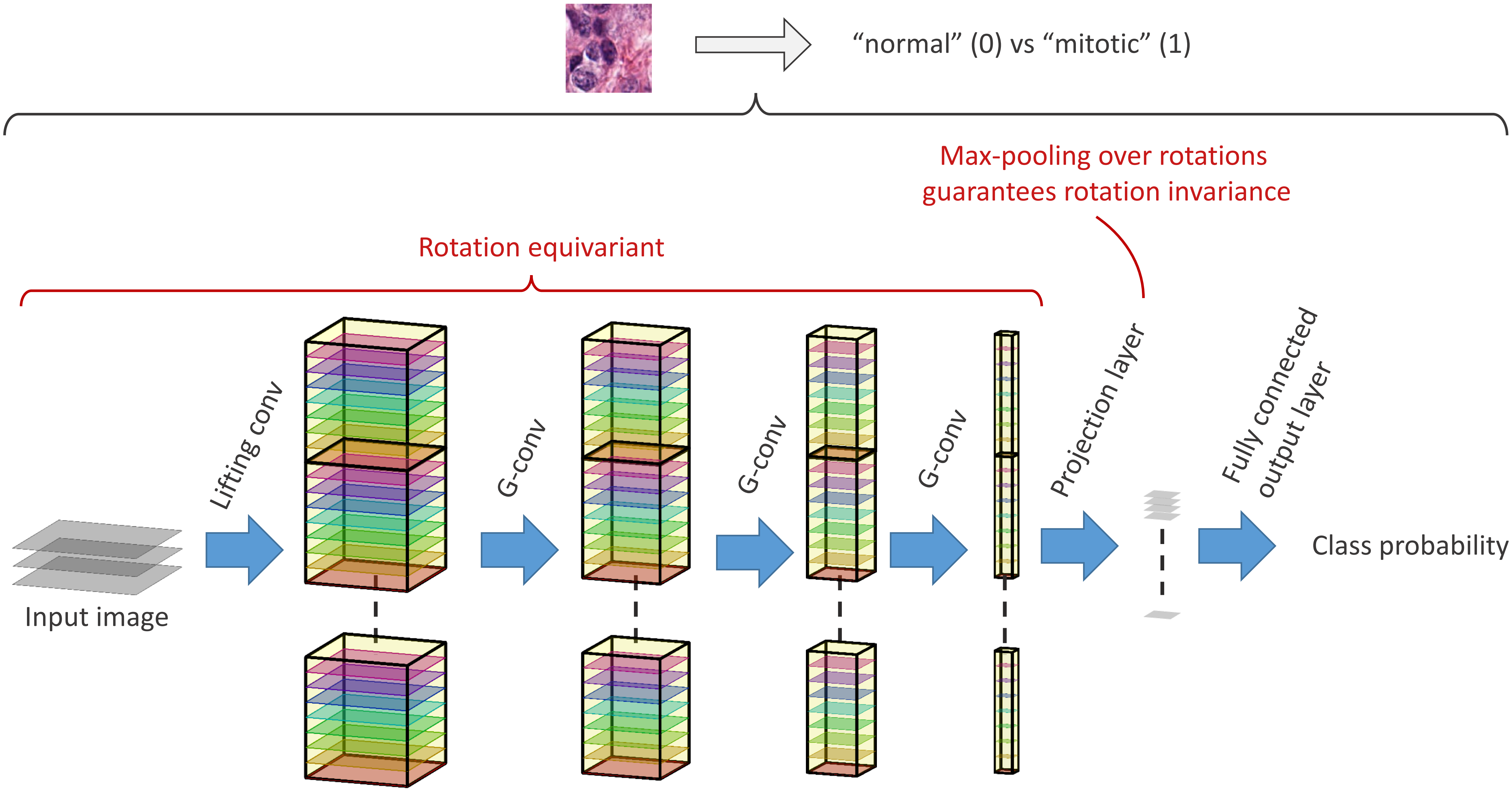
periodic shift

$\mathcal{L}_{\theta}^{SO(2)\to\mathbb{L}_2(SE(2))} k$

Rotated kernel

Architecture for rotation invariant patch classification

Bekkers, Lafarge et al. 2018



"normal" (0) vs "mitotic" (1)

Max-pooling over rotations guarantees rotation invariance

Rotation equivariant

Input image

Lifting conv

G-conv

G-conv

G-conv

Projection layer

Fully connected output layer

Class probability

Results

Bekkers, Lafarge et al. 2018

G-CNNs outperform CNNs (matched in network complexity):

- Even when training the classical CNNs **with** and G-CNNs **without** data-augmentation
- G-CNNs do not have to spend valuable network capacity on learning geometric structure -> focus entirely on learning effective representations

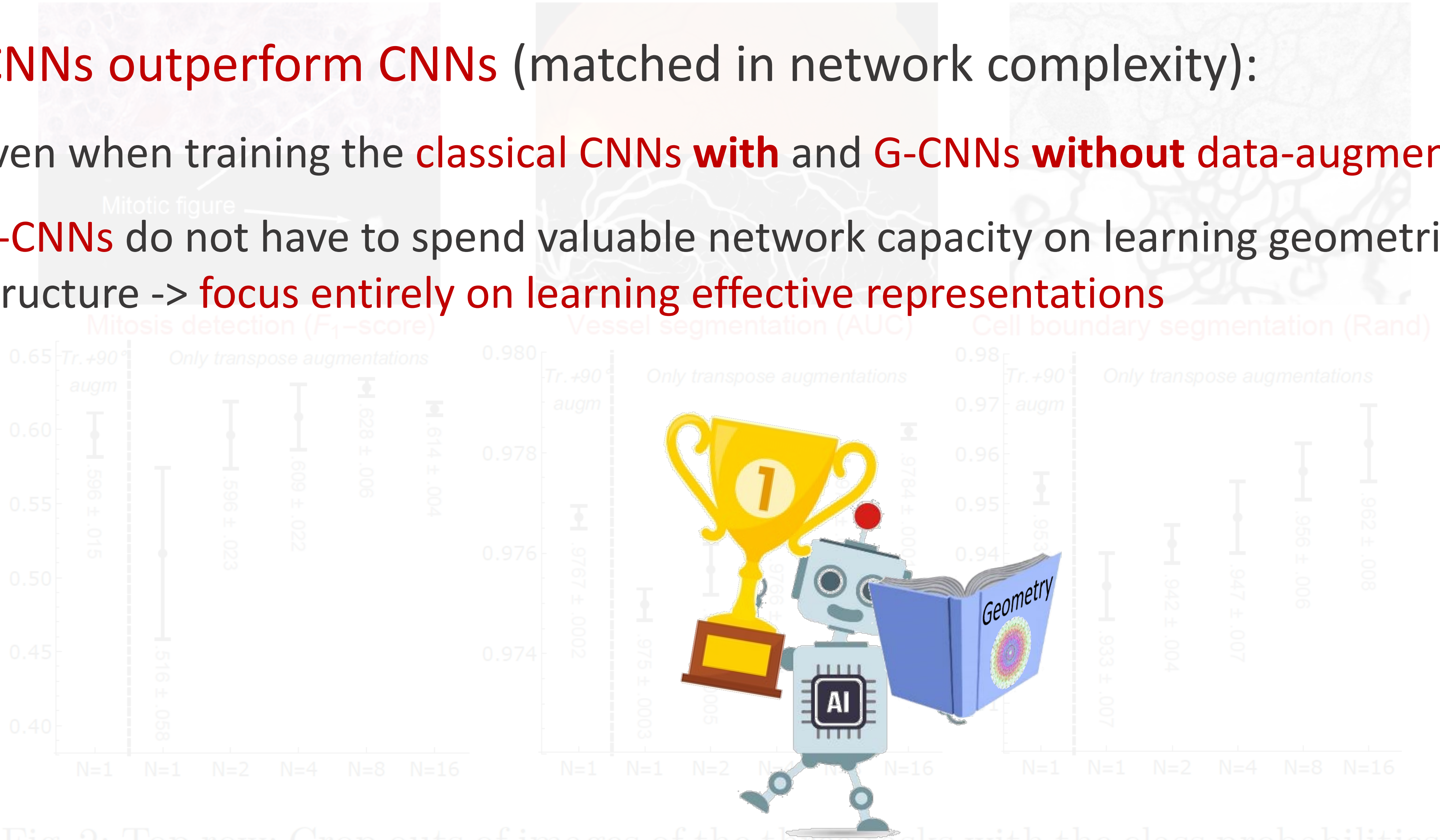


Fig. 2: Top row: Crop outs of images of the three tasks with the class probabilities generated by our method. Bottom row: Mean results (±1 std. dev.).

Related work on group equivariant networks

Based on the overview given in *Cohen-Geiger-Weiler 2018*

Group convolution networks
(domain extension)

LeCun et al 1990	$\mathbb{Z}^2$	translation networks
Mallat et al. 2013, 2015	SE(2)	Scattering transform & SVM
Bekkers et al. 2014-2018	SE(2)	via B-splines, 2 layer G-CNN
Cohen-Welling 2016	p4m	via 90° rotations + flips + theory!
Dieleman et al. 2016	p4m	via 90° rotations + flips
Weiler et al. 2017	SE(2)	via circular harmonics
Zhou et al. 2017	SE(2)	via bilinear interpolation
Bekkers et al. 2018	SE(2)	via bilinear interpolation
Hoogeboom et al. 2018	S(2,6)	hexagonal grids
Winkels-Cohen 2018	SE(3,N) + m	90° rotations + flips
Worrall-Brostow 2018	SE(3,N)	90° rotations
Cohen et al. 2018	SO(3)	via spherical harmonics

Steerable filter networks
(co-domain extension)

Worrall et al. 2017	SE(2)	irrep
Marcos et al. 2017	SE(2)	vector field networks
Kondor 2018	SE(3)	irrep, N-body nets
Thomas et al. 2018	SE(3)	irrep, point clouds
Weiler et al. 2018	SE(3)	irrep
Esteves	SO(3)/SO(2)	irrep
Kondor-Trivedi 2018	SO(d)	irrep (on compact quotient sp.)

Continuous

Discrete

Can we use the theory in practice for other groups?

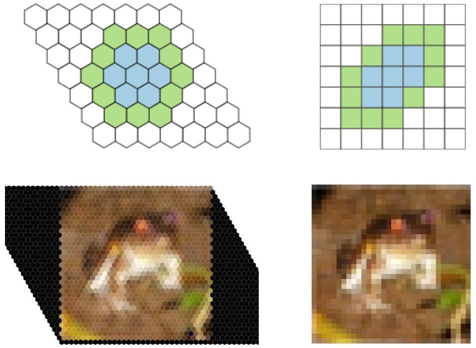
G-CNNs are currently limited to compact groups:

Discrete <> no interpolation

Continuous <> Fourier theory on groups

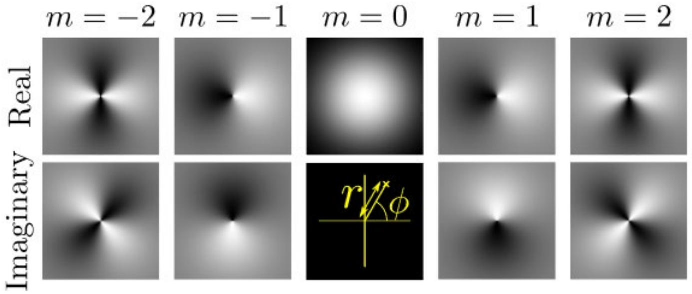
HexaConv

Hoogeboom, Peters, Cohen, Welling – ICLR 2018



Circular/Spherical harmonics

Worall, Garbin, Turmukhambetov, Brostow – CVPR 2017



Why this limitation? Available tools. We need to implement transformations (and sampling) of the convolution kernels.

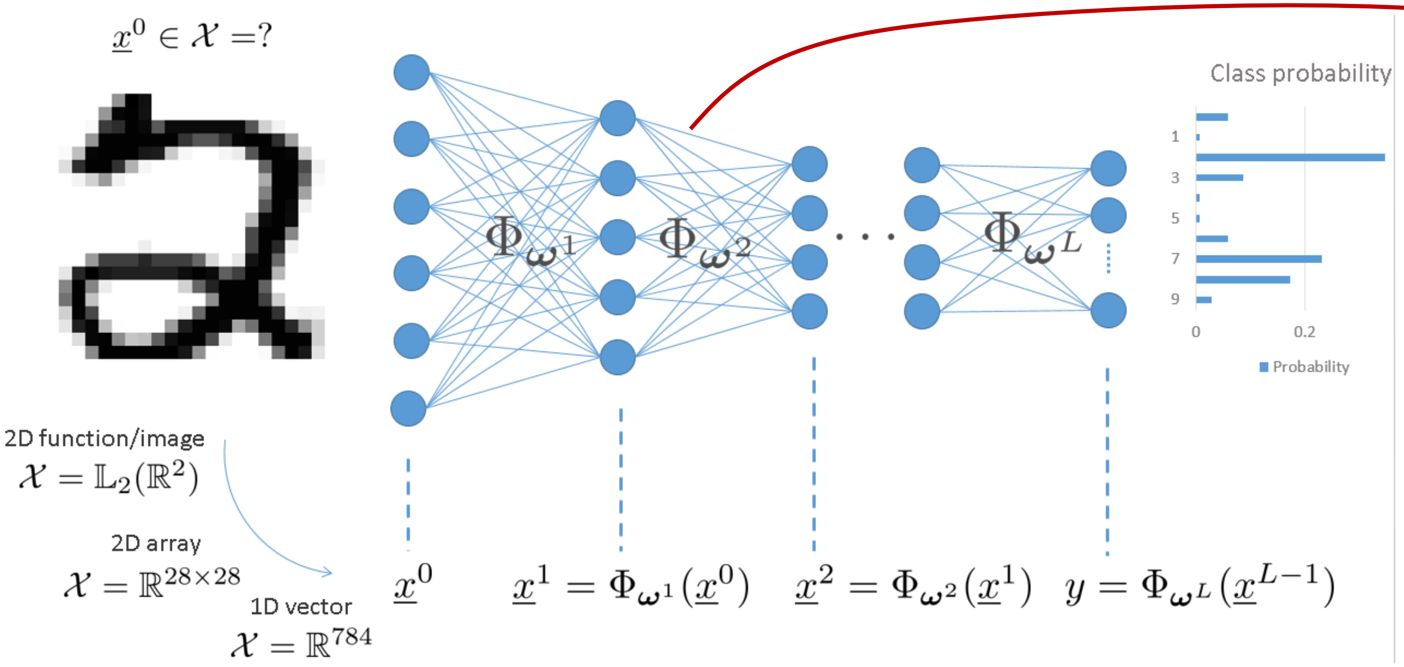
Solution? A new flexible class of basis functions that enables to implement G-convs for arbitrary Lie groups.

B-Splines on Lie groups

Equivariance ⟺ G-CNNs

If you want equivariance G-CNNs are the way to go

Classical artificial neural networks

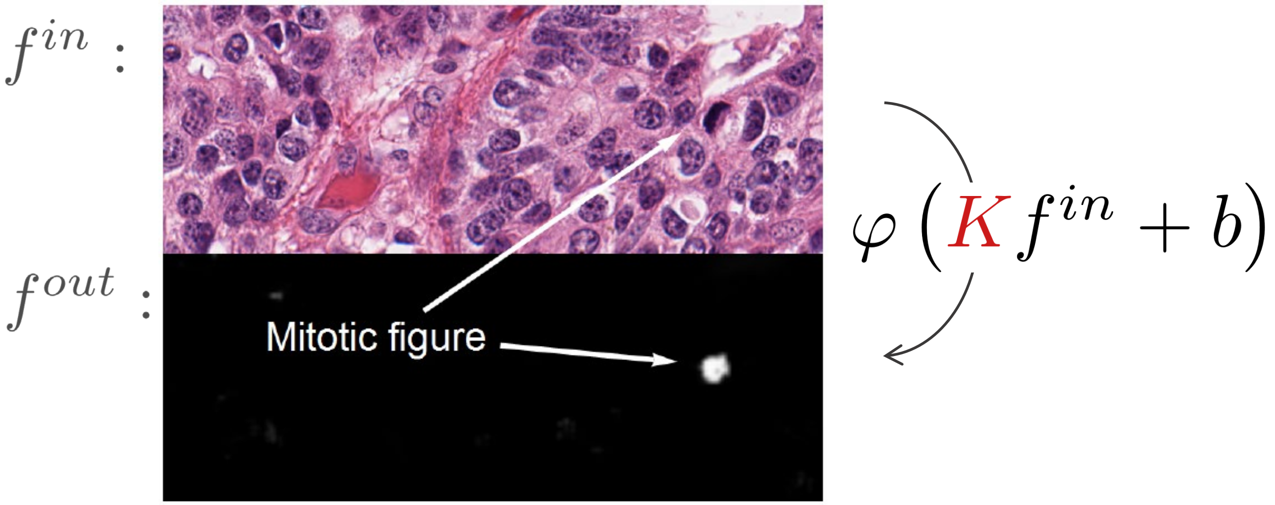


$$\boxed{\underline{x}^l = \varphi\left(K_{\mathbf{w}^l}\,\underline{x}^{l-1} + \underline{b}^l\right)}$$

$\underline{x}^{l-1} \in \mathbb{R}^{N^{l-1}}$	The input vector
$\underline{x}^{l} \in \mathbb{R}^{N^{l}}$	The output vector
$K_{\mathbf{w}} : \mathbb{R}^{N^l \times N^{l-1}}$	A linear mapping parameterized by weights $\mathbf{w}$
$\underline{b} \in \mathbb{R}^{N^l}$	A bias term
$\varphi : \mathbb{R} \rightarrow \mathbb{R}$	An activation function (applied element wise)
$\omega = (\mathbf{w}, \underline{b})$	The trainable parameters

Artificial NNs in the continuous world

Images as functions in $\mathbb{L}_2(\mathbb{R}^2)$



$$\varphi\left(K f^{in} + b\right)$$

Linear (and bounded) mappings between feature maps are kernel operators *(Dunford-Pettis)*

$$(K f)(y) = \int_X k(y,x) f(x) \mathrm{d}x$$

Equivariance constraint on *K* implies group convolution!

$f^{in} \in \mathcal{X} = \mathbb{L}_2(X)$	The input "vector": function on space X
$f^{out} \in \mathcal{Y} = \mathbb{L}_2(Y)$	The output "vector": function on space Y
$K_{\mathbf{w}} : \mathcal{X} \to \mathcal{Y}$	A linear mapping parameterized by weights $\mathbf{w}$
$b \in \mathbb{R}$	A bias term
$\varphi : \mathbb{R} \to \mathbb{R}$	An activation function (applied element wise)
$\omega = (\mathbf{w}, \underline{b})$	The trainable parameters

Artificial NNs in the continuous world

Images as functions in $\mathbb{L}_2(\mathbb{R}^2)$

Linear (and bounded) mappings between feature maps are kernel operators (Dunford-Pettis)

$$(Kf)(y) = \int_X k(y,x)f(x)\mathrm{d}x$$

f^{in} :

f^{out} :

*Bekkers 2019, Thm 1**

**Work with Remco Duits at TU/e.*

See also: Duits 2005 – Thm 25, Cohen, Geiger, Weiler 2018 - Thm 6.1, Kondor, Trivedi 2018 - Thm 1

Theorem 1. *Let operator $\mathcal{K} : \mathbb{L}_2(X) \to \mathbb{L}_2(Y)$ be linear and bounded, let X, Y be homogeneous spaces on which Lie group G act transitively, and $\mathrm{d}\mu_X$ a Radon measure on X, then*

1. *$\mathcal{K}$ is a kernel operator, i.e., $\exists_{\tilde{k}\in\mathbb{L}_1(Y\times X)} : \; (\mathcal{K}f)(y) = \int_X \tilde{k}(y,x)f(x)\mathrm{d}\mu_X$,*
2. *with equivariance constraint $\forall g \in G : \mathcal{K} \circ \mathcal{L}_g^{G\to\mathbb{L}_2(X)} = \mathcal{L}_g^{G\to\mathbb{L}_2(Y)} \circ \mathcal{K}$ the map is defined by a one-argument kernel*

$$\tilde{k}(y,x) = \qquad \tilde{k}(y_0, g_y^{-1} \odot x) =: \qquad k(g_y^{-1} \odot x) \tag{3}$$

for any $g_y \in G$ such that $y = g_y \odot y_0$ for some fixed origin $y_0 \in Y$,

3. *if $Y \equiv G/H$ is the quotient of G with $H = \mathrm{Stab}_G(y_0)$ then the kernel is constrained via*

$$\forall_{h\in H}, \forall_{x\in X} : \qquad k(x) = \qquad k(h^{-1} \odot x), \tag{4}$$

Our options for SE(2) equivariance

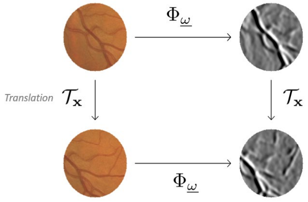
With

$$Y \equiv SE(2)/SO(2)$$

Equivariance requires

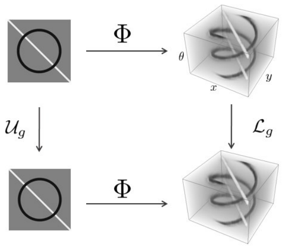
$$k(\mathbf{R}_\theta \mathbf{x}) = k(\mathbf{x})$$

2D cross-correlations $\mathcal{K} : \mathbb{L}_2(\mathbb{R}^2) \to \mathbb{L}_2(\mathbb{R}^2)$



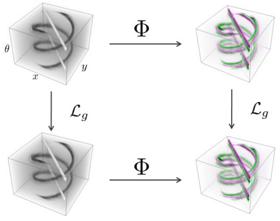
$$(\mathcal{K}f)(\mathbf{y}) = (\mathcal{T}_{\mathbf{x}}k, f)_{\mathbb{L}_2(\mathbb{R}^2)} = \int_{\mathbb{R}^2} k(\mathbf{x}-\mathbf{y})f(x)\mathrm{d}\mathbf{x}$$

SE(2) lifting correlations $\mathcal{K} : \mathbb{L}_2(\mathbb{R}^2) \to \mathbb{L}_2(SE(2))$



$$(\mathcal{K}f)(g) = (\mathcal{U}_g k, f)_{\mathbb{L}_2(\mathbb{R}^2)} = \int_{\mathbb{R}^2} k(\mathbf{R}_\theta^{-1}(\mathbf{x}'-\mathbf{x}))f(\mathbf{x}')\mathrm{d}\mathbf{x}'$$

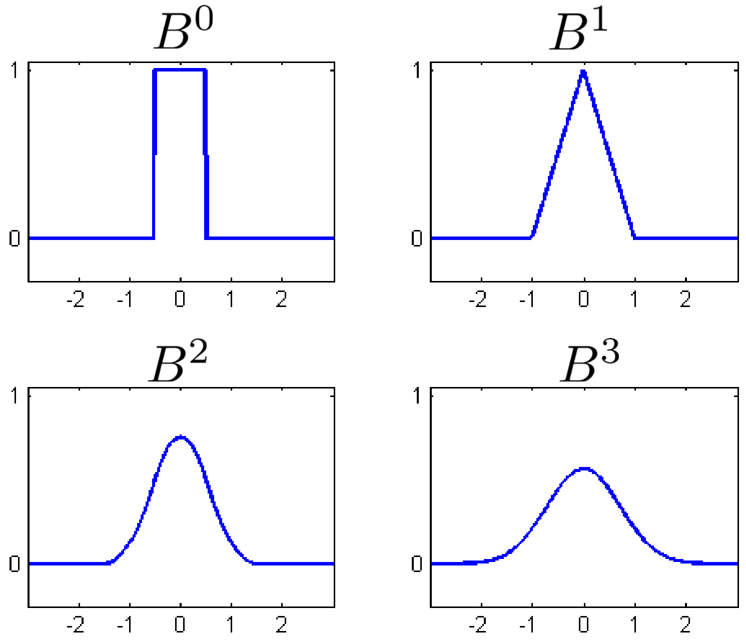
SE(2) G-correlations $\mathcal{K} : \mathbb{L}_2(SE(2)) \to \mathbb{L}_2(SE(2))$



$$(\mathcal{K}F)(g) = (\mathcal{L}_g K, F)_{\mathbb{L}_2(SE(2))} = \int_{SE(2)} K(\mathbf{R}_\theta^{-1}(\mathbf{x}'-\mathbf{x}), \theta'-\theta)F(\mathbf{x}',\theta')\mathrm{d}g'$$

B-Splines on Lie groups

B-Splines on $\mathbb{R}^d$



Piecewise polynomial!
Finite support!

1D basis function

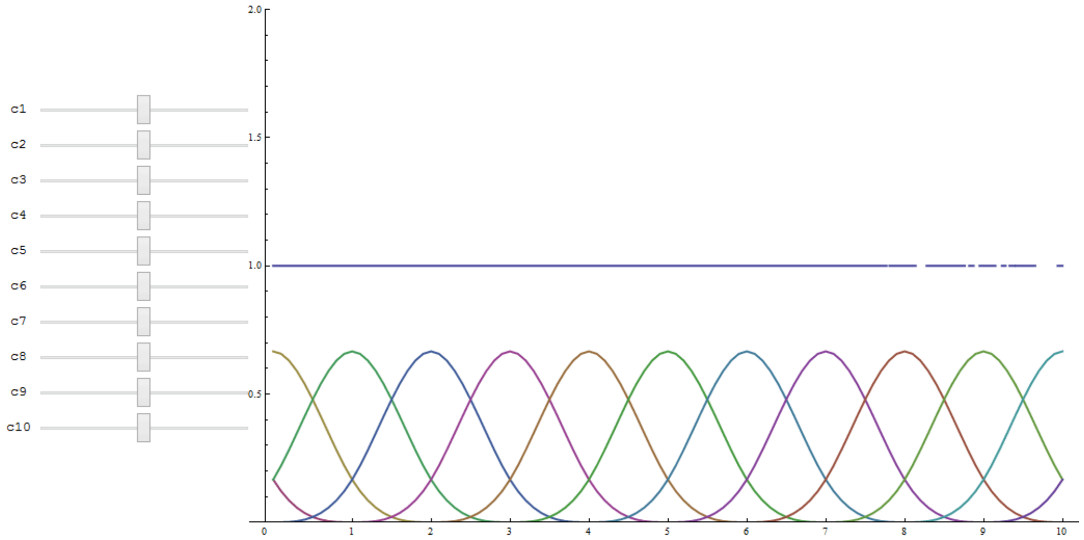
$$B^n(x) := \left(1_{[-\frac{1}{2},\frac{1}{2}]} *^{(n)} 1_{[-\frac{1}{2},\frac{1}{2}]}\right)(x),$$

Basis function on $\mathbb{R}^d$

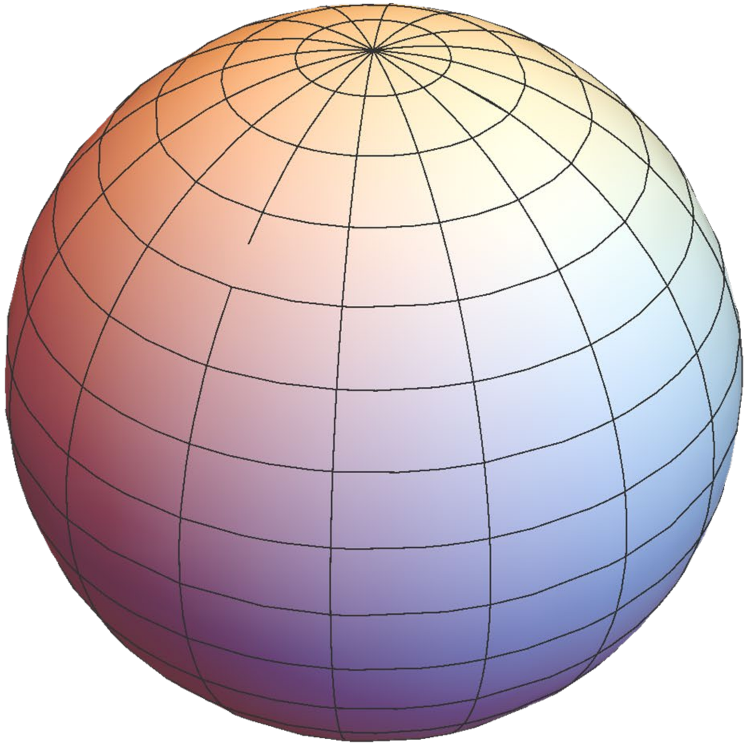
$$B^{\mathbb{R}^d,n}(\mathbf{x}) := \underbrace{(B^n \otimes \cdots \otimes B^n)(\mathbf{x})}_{d\ \text{times}} = B^n(x_0)B^n(x_1)\dots B^n(x_d)$$

Uniform B-Spline on $\mathbb{R}^d$

$$f(\mathbf{x}) := \sum_{i=1}^{N} c_i B^{\mathbb{R}^d,n}\left(\frac{\mathbf{x}-\mathbf{x}_i}{s_{\mathbf{x}}}\right)$$



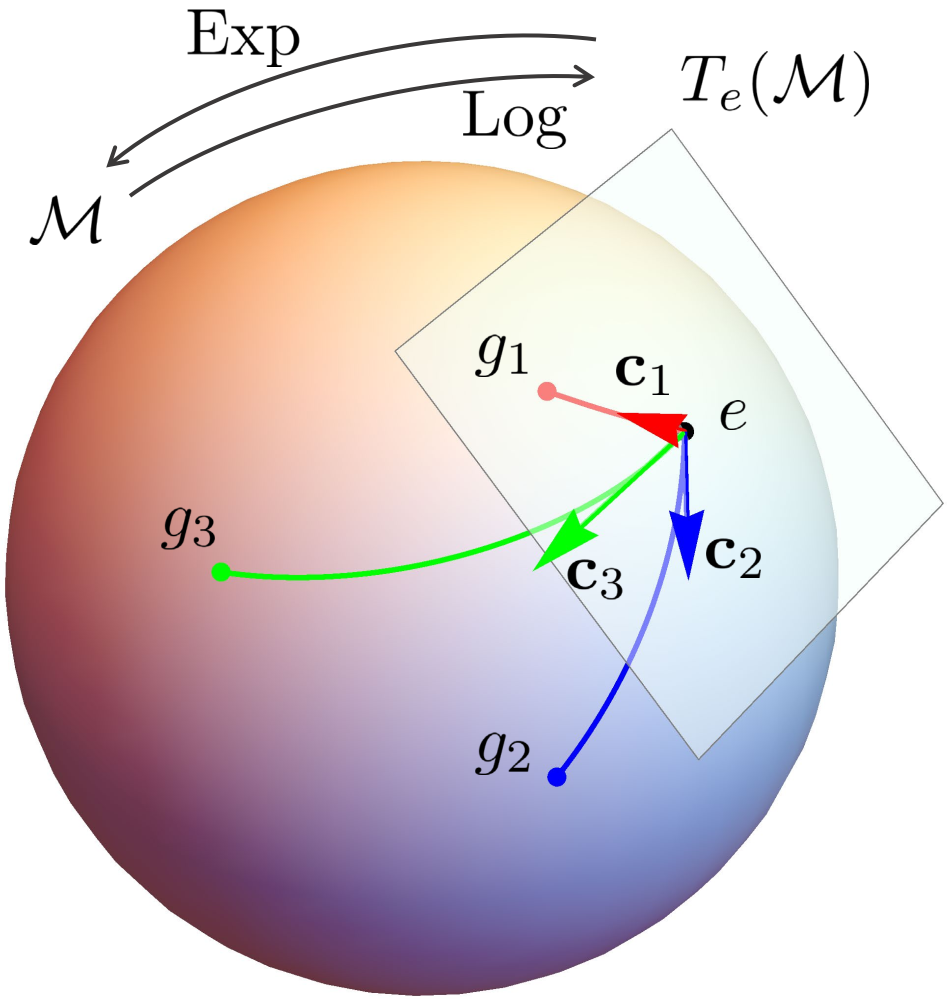
How to define B-Splines on manifolds?

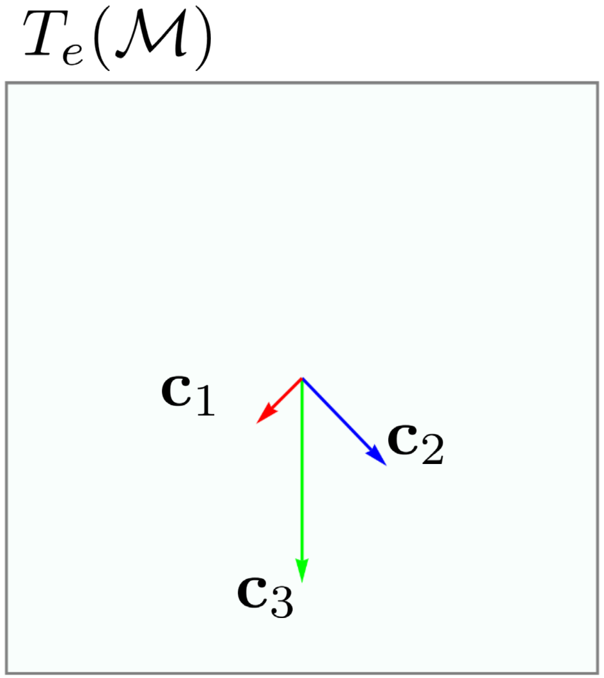


What is the meaning of “uniform” on a manifold?

What parameterization to use?

The exponential and logarithmic map



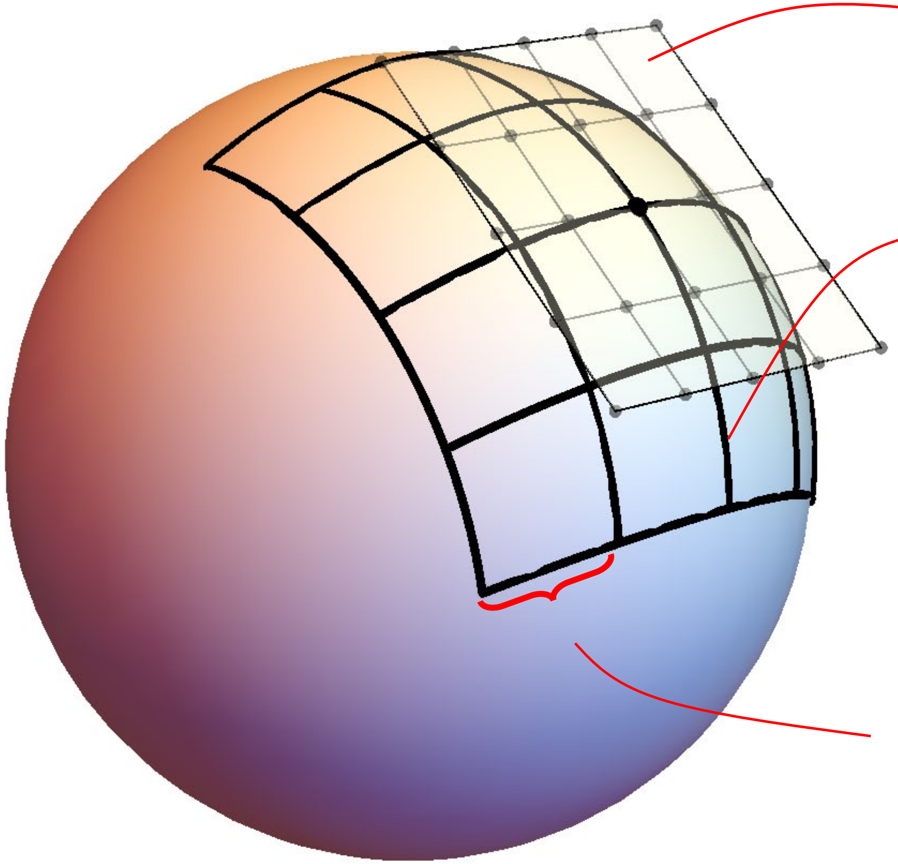


The distance from a point g to the origin e is given by the length of its “initial velocity vector”

$$\|\mathrm{Log}\, g\|$$

Link: SO(3)/SO(2)

A grid on the Lie algebra maps to a grid on *G*



Now we can define B-splines on the vector space of the Lie algebra.

This then defines a function on the group.

Equidistant w.r.t. the default distance on the group $\|\mathrm{Log}(h_i^{-1} \cdot h_j)\|$

B-Splines on Lie groups $G = \mathbb{R}^d \rtimes H$

Via the Logarithmic map

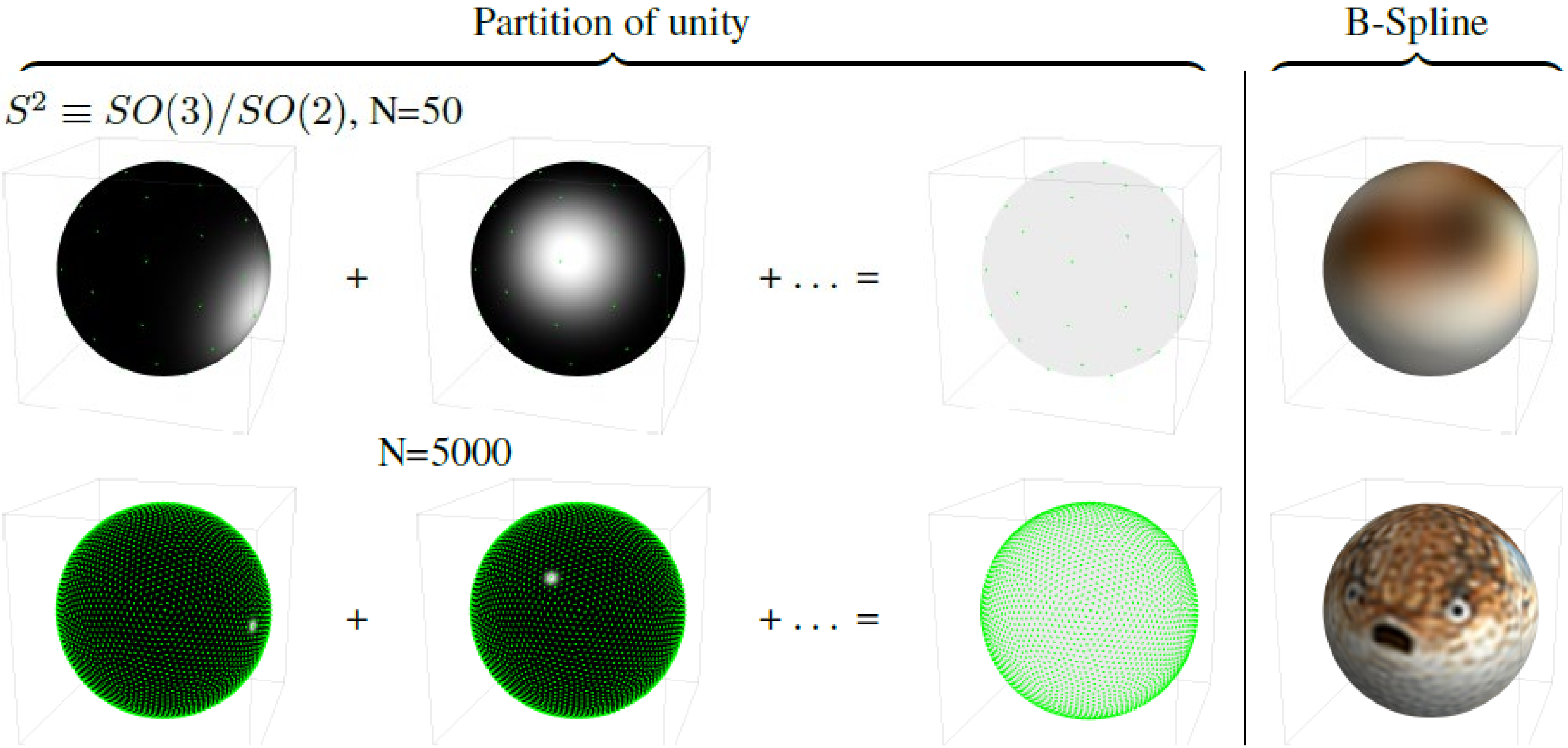
$$k(g) = \sum_{i=1}^{N} w_i B\left(\frac{x - x_i}{s_x}\right) B\left(\frac{\text{Log}(h^{-1} \cdot h_i)}{s_h}\right)$$

Examples of B-Splines on *H*

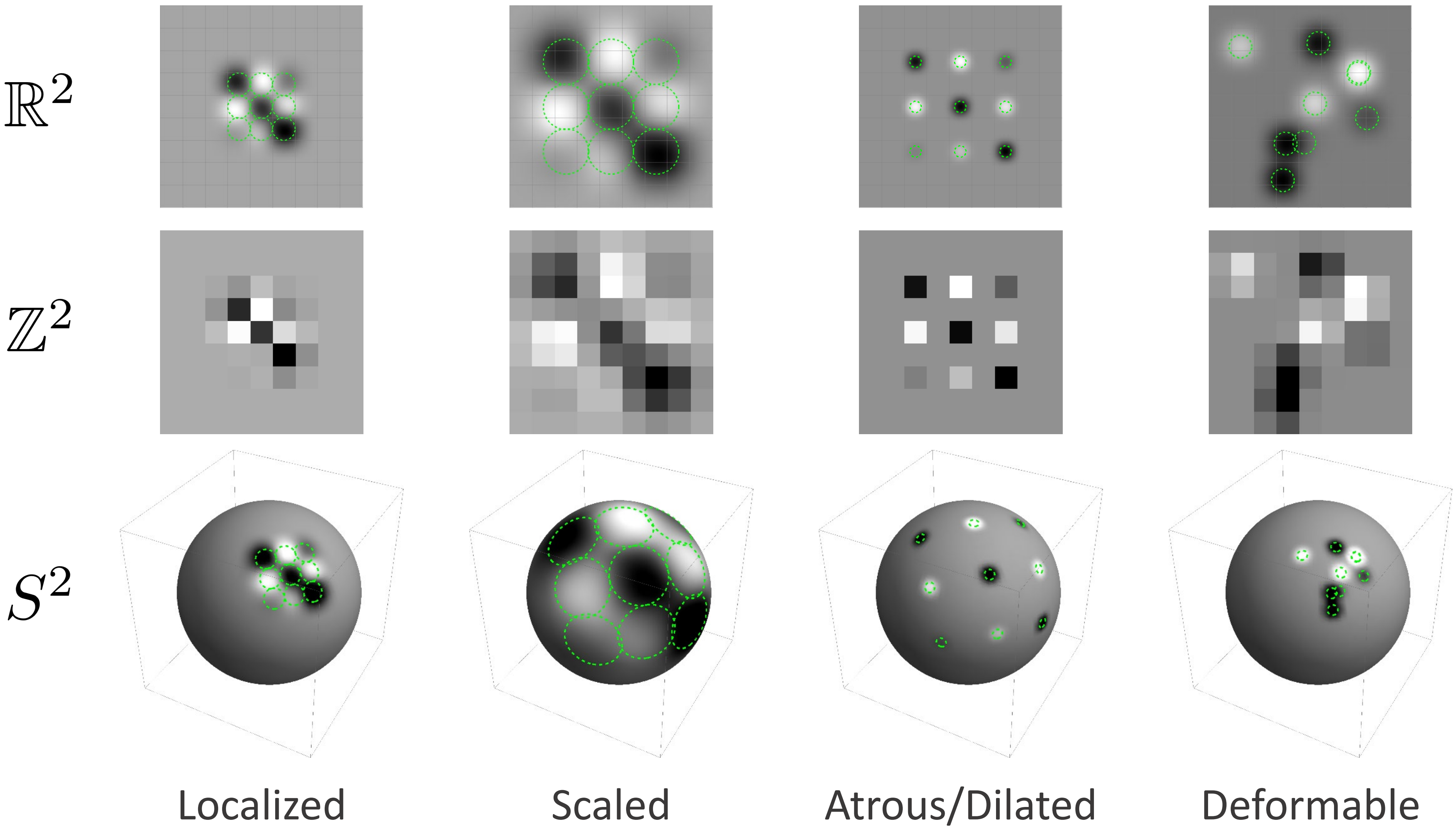
3D Rotations (quotient)

Isotropic (quotient)

Approximately uniform



Unique properties of B-spline kernels



Properties of B-splines on Lie groups

- Enables to construction a basis on any Lie group

- To build full G-CNNs for groups of type $G = \mathbb{R}^d \rtimes H$ we only need:
  - The group product and inverse of H
  - Its action on $\mathbb{R}^d$
  - The logarithmic map (which is analytic)

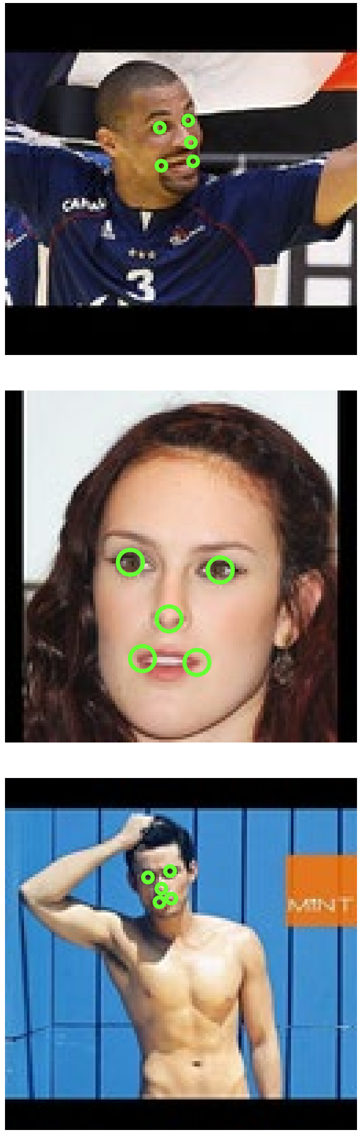
Modular code (released soon…)

```
<< import gsplinets
<< layers = gsplinets.layers('SE2')
```

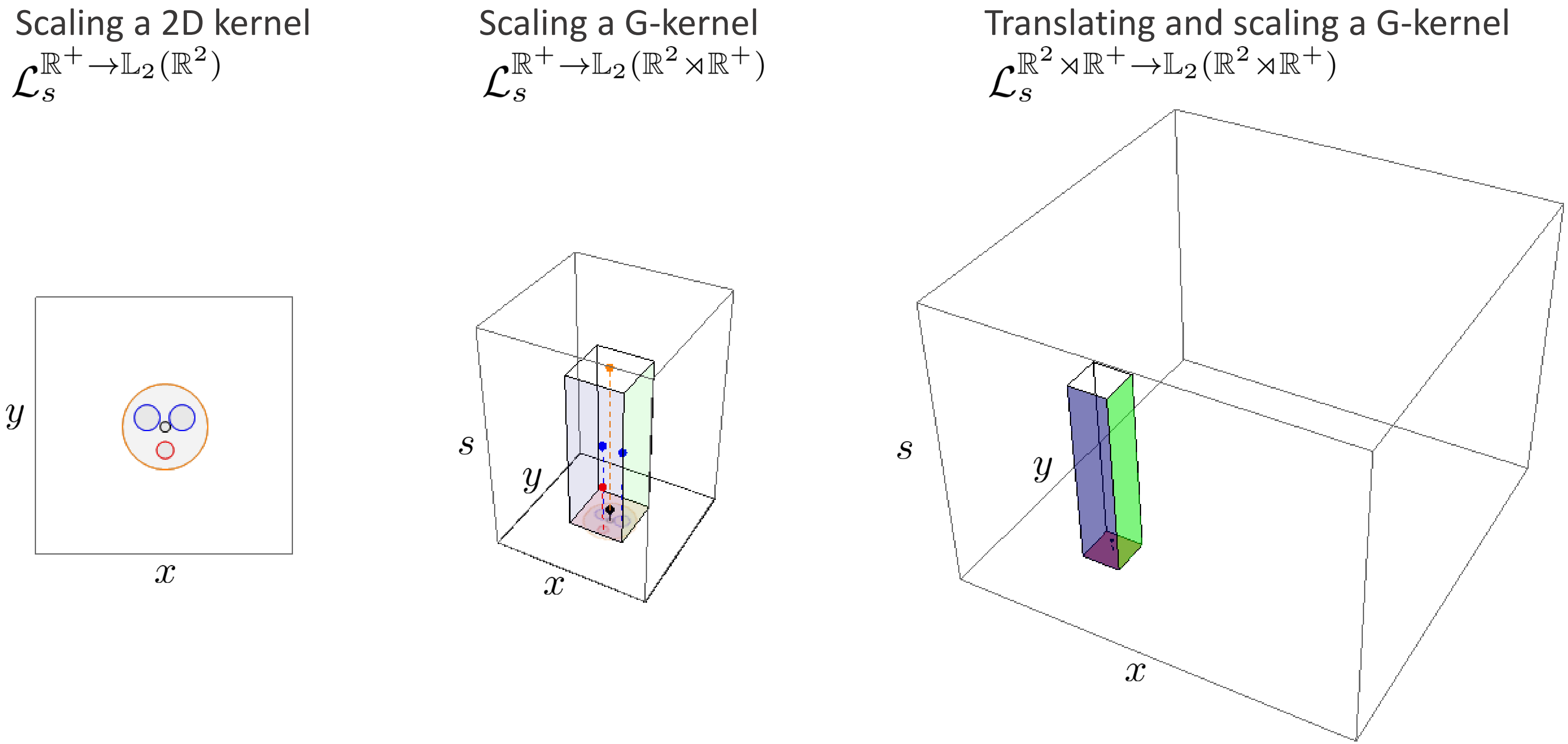
- Enables heuristics from conventional CNN architectures:
  - Dense/"fully connecting" convolution kernels on H
  - Localized convolutions on H
  - Atrous convolutions on H
  - Deformable kernels (also optimize over the centers of the splines)
  - …

Results

Case 1 (Scaling invariance): Facial landmark detection | CelebA database | 6 G-CNN layers



Principle behind scale-translation G-CNNs

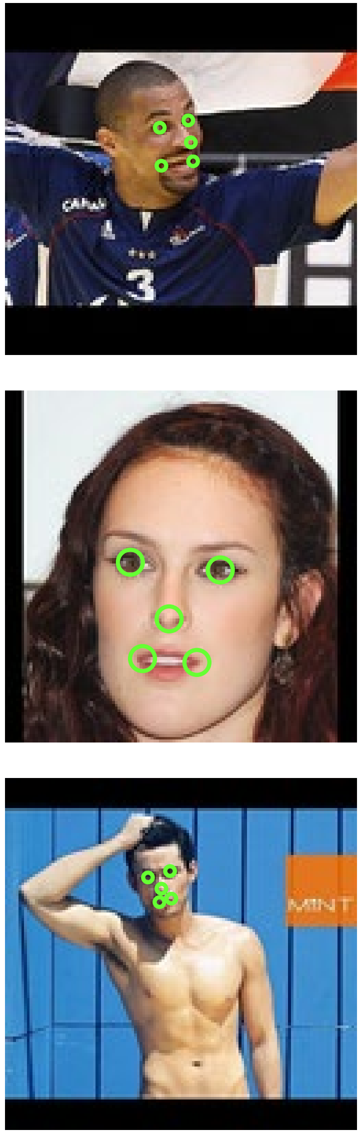


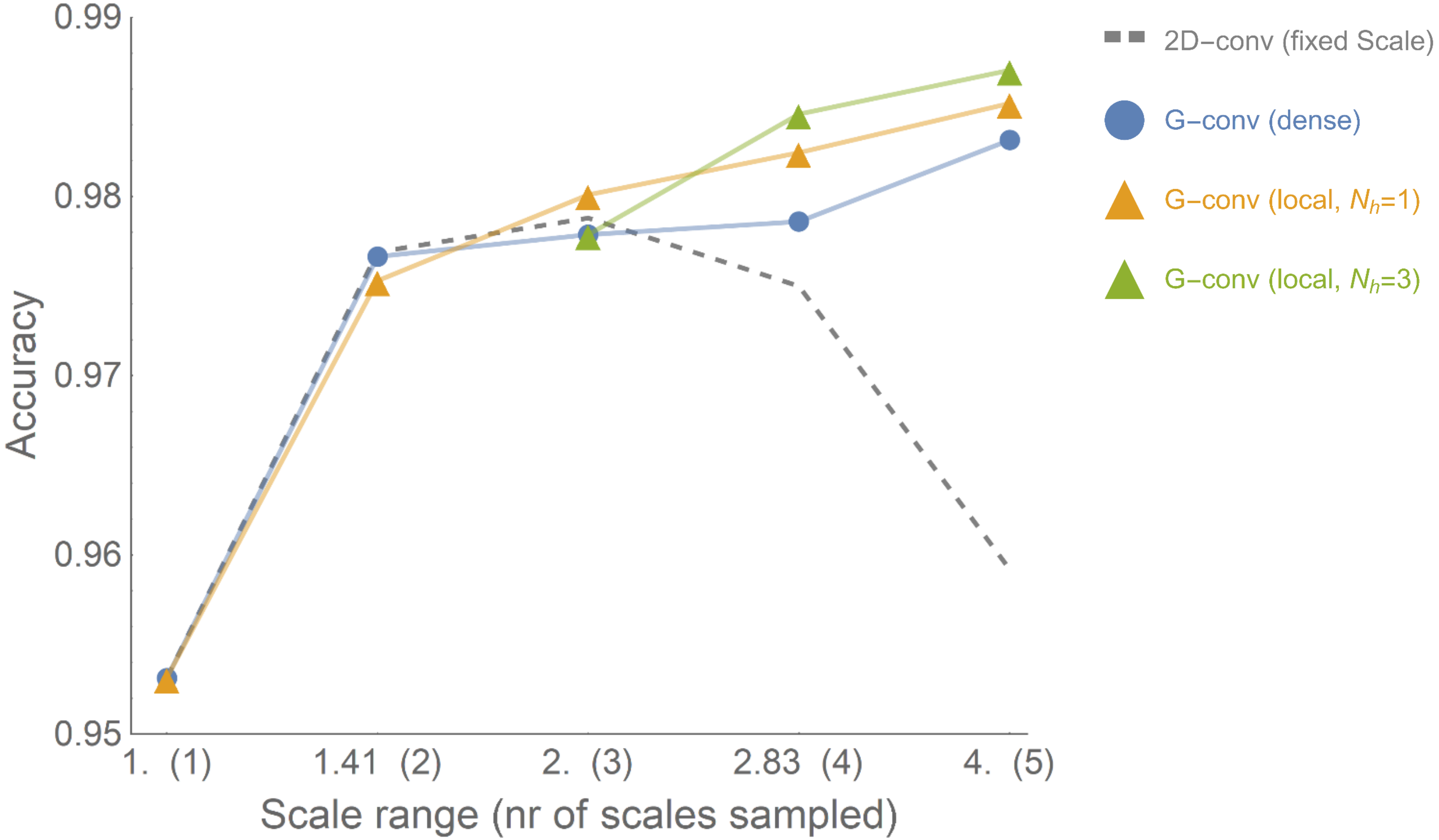
Scaling a 2D kernel
$\mathcal{L}_s^{\mathbb{R}^+\to\mathbb{L}_2(\mathbb{R}^2)}$

Scaling a G-kernel
$\mathcal{L}_s^{\mathbb{R}^+\to\mathbb{L}_2(\mathbb{R}^2\rtimes\mathbb{R}^+)}$

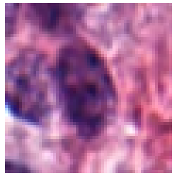
Translating and scaling a G-kernel
$\mathcal{L}_s^{\mathbb{R}^2\rtimes\mathbb{R}^+\to\mathbb{L}_2(\mathbb{R}^2\rtimes\mathbb{R}^+)}$

Case 1 (Scaling invariance): Facial landmark detection | CelebA database | 6 G-CNN layers



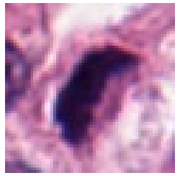


Case 2 (Rotation invariance): Cancer detection | PCAM database | 4 G-CNN layers



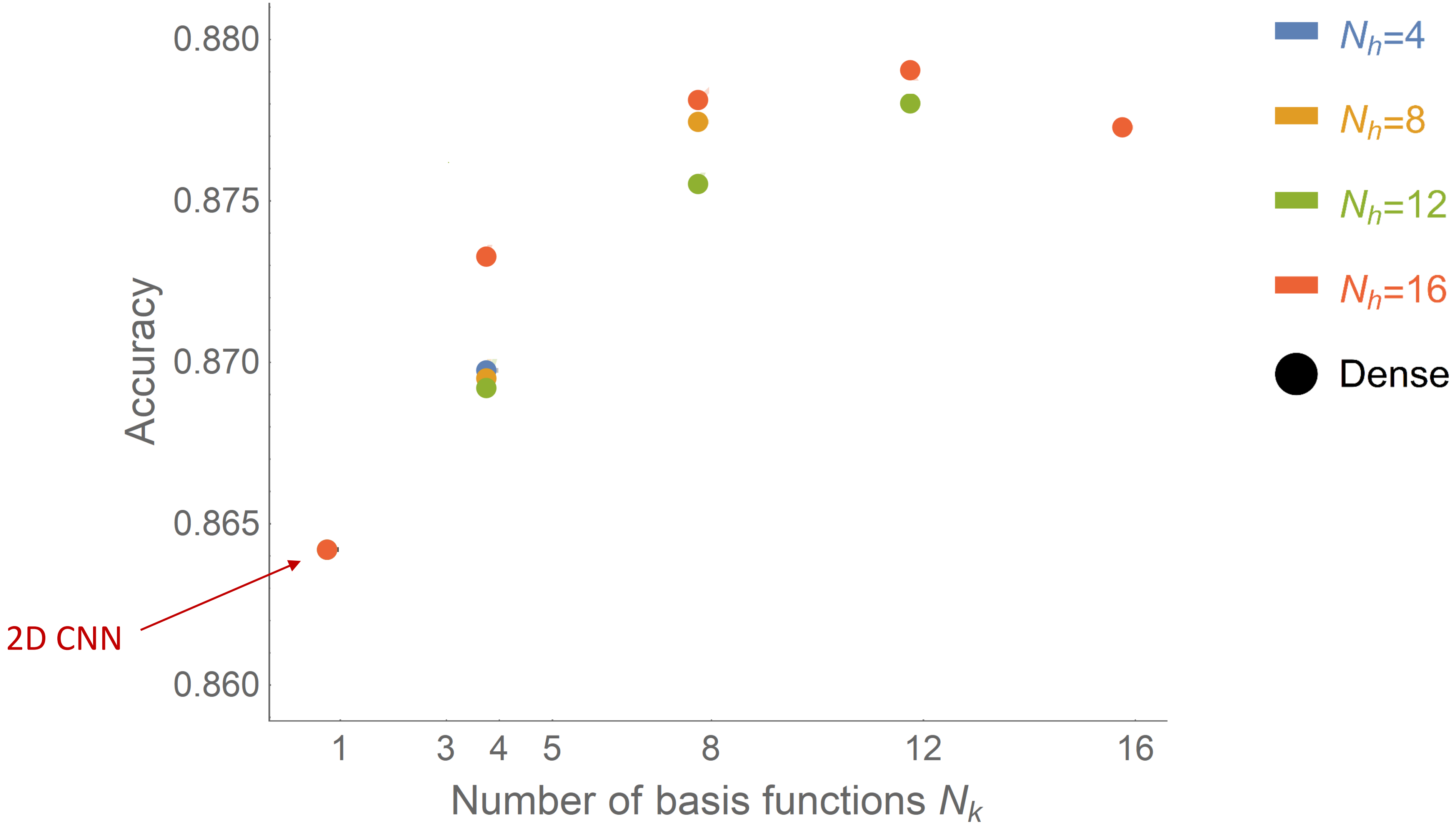
normal

normal



mitotic

...



2D CNN

Conclusion

Conclusion

- G-CNNs “naturally” arise from NNs under equivariance constraints

- G-CNNs improve upon classic CNNs by
  - Making data augmentation w.r.t. the group obsolete
  - No trainable weights need to be spend on learning geometry behavior
  - Additional geometry structure allows to deal with context (recognition by components, relative poses)

- B-Splines can be used to build G-CNNs for a large class of transf. groups

- They enable unique properties
  - Localized G-convs
  - Atrous G-convs
  - Deformable G-convs
  - Flexibility in kernel resolution (# basis functions) vs sampling resolution (# grid points)

- Experimental results
  - G-CNNs outperform 2D CNNs
  - Localized G-CNNs generally outperform full/dense G-CNNs
  - Atrous G-CNNs generally outperform full/dense G-CNNs

Thank you for your attention!

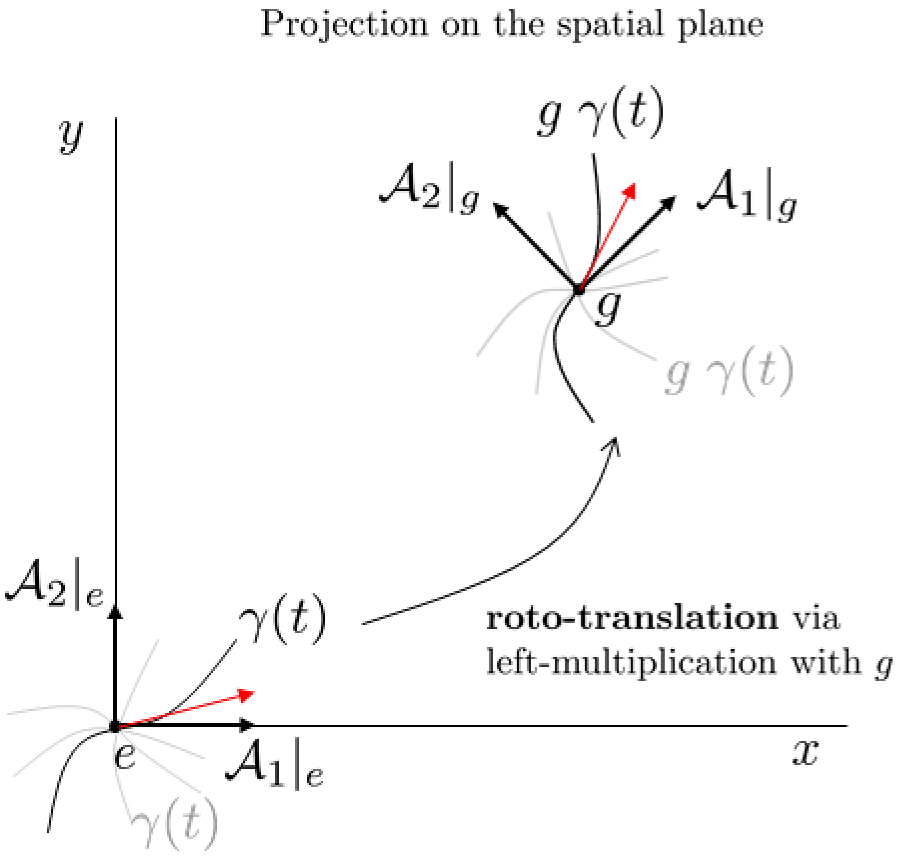
Ph.D. position on this topic coming up at AMLab, University of Amsterdam

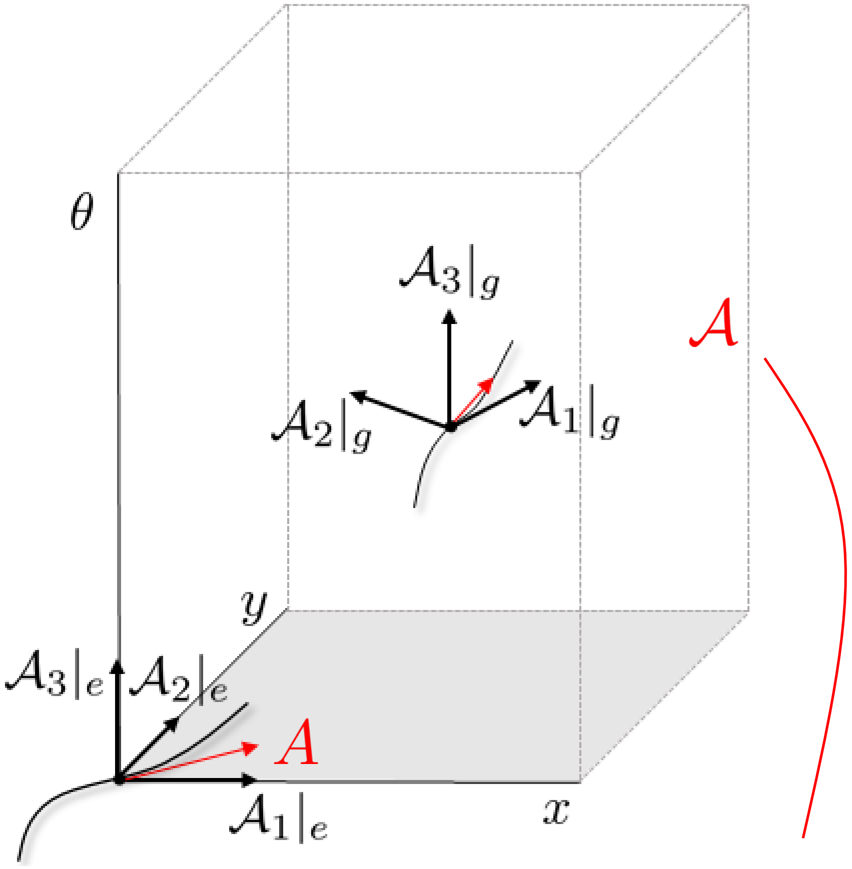
Amsterdam Machine Learning Lab
Informatics Institute
University of Amsterdam

Backup slides

On SE(2) and SO(3) and Exp/Log map

Left-invariant vector fields (push forward of left mult.)





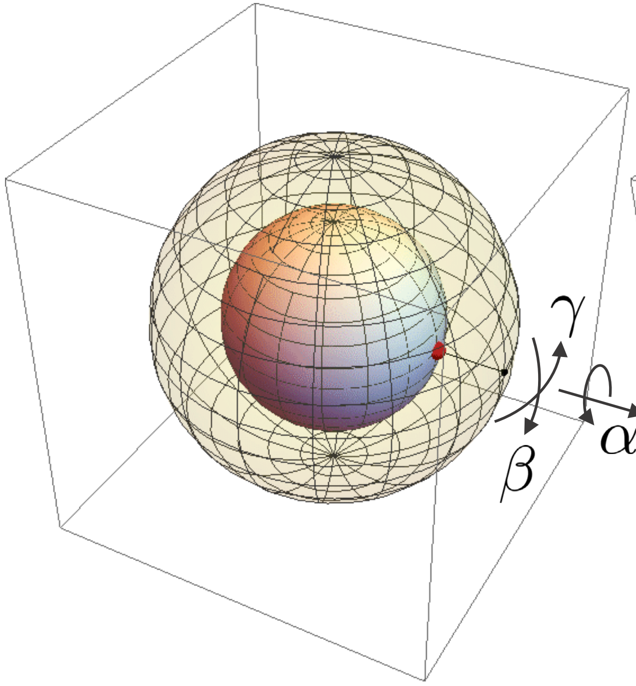
Left-invariant vector field

A tangent space at the origin defines a left-invariant tangent bundle on the group

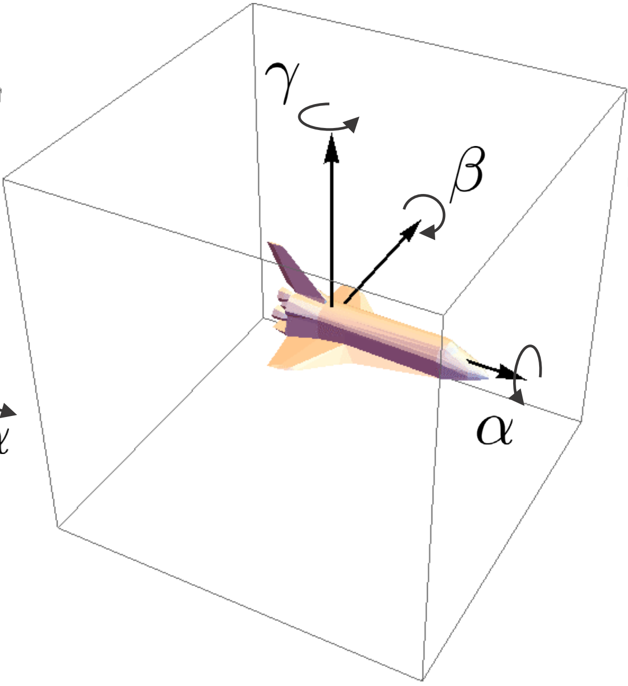
The 3D Rotation group and the sphere as a quotient

The 3D rotation group

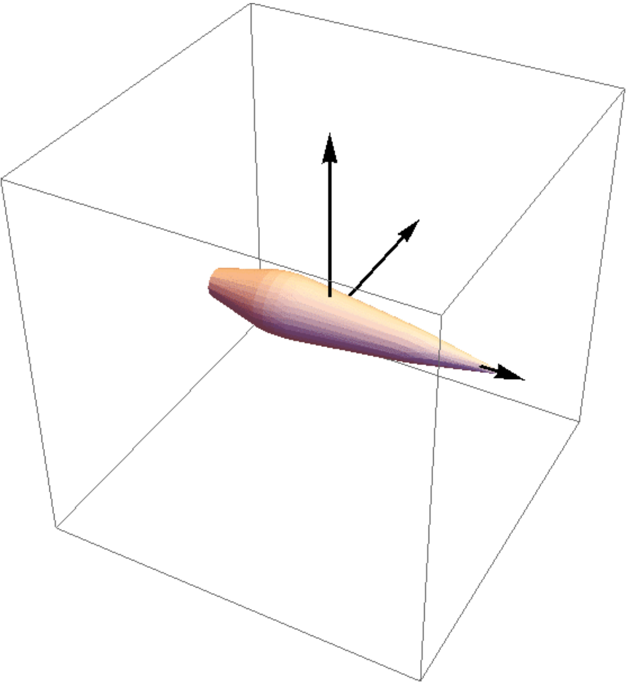
Representation in parameter space (XYZ-Euler angles)



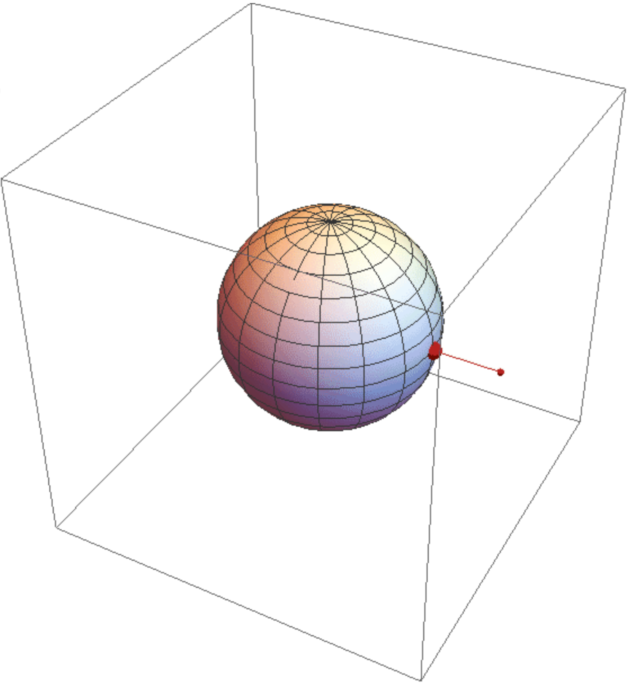
Rotation by $R \in SO(3)$
$R = R_{\mathbf{e}_z,\gamma} R_{\mathbf{e}_y,\beta} R_{\mathbf{e}_x,\alpha}$



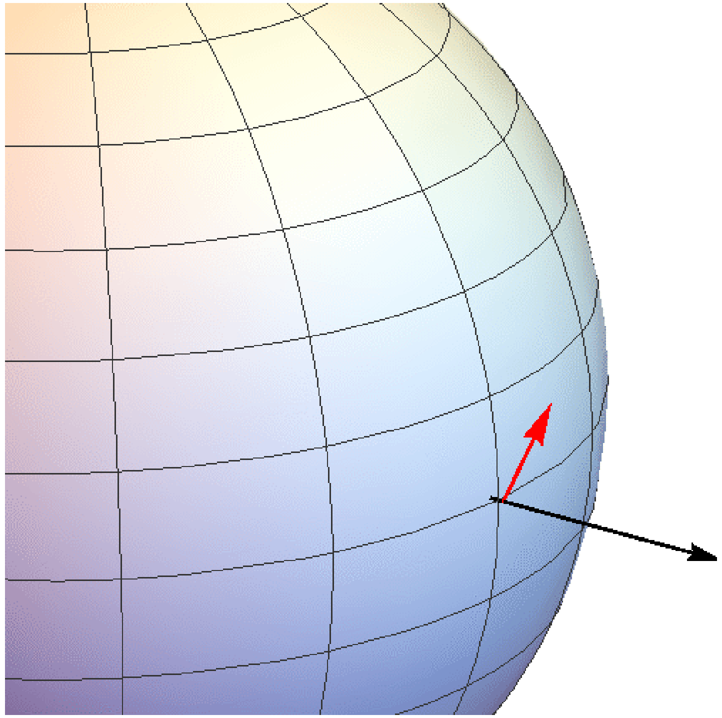
The 2-sphere as a quotient group



$S^2 \equiv SO(3)/SO(2)$

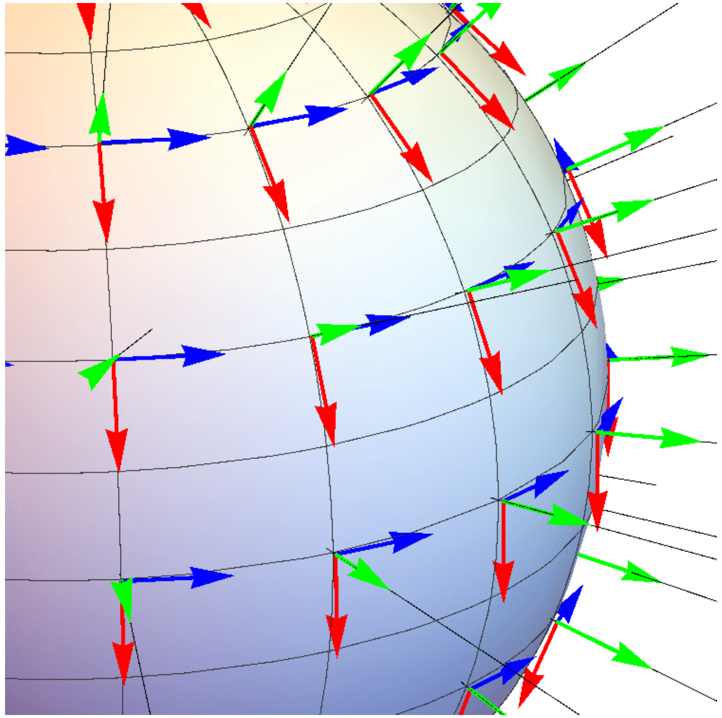


Some animations on vector fields

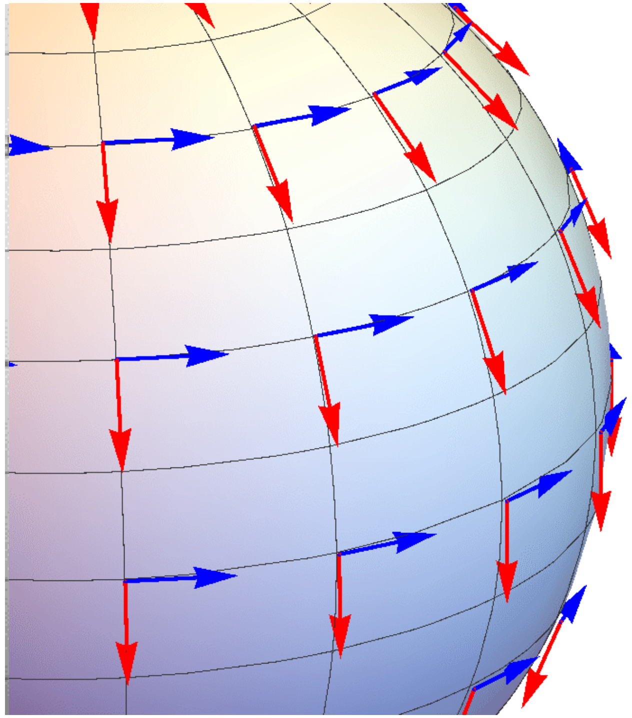


The group structure can be used to “transport” vectors.

A vector at the origin defines a whole vector field!

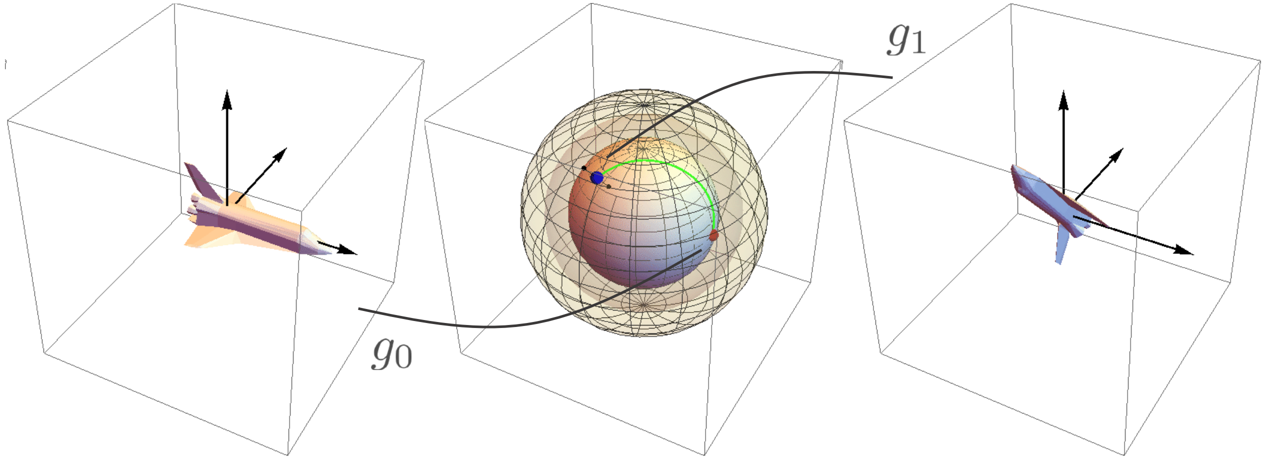


This generates a frame of reference attached to each $g \in G$

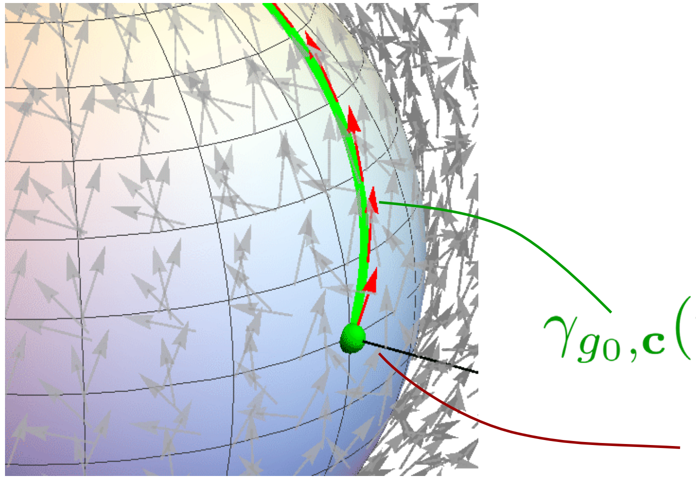


In a quotient group this frame is not unique…

The exponential map: integrating along a vector field

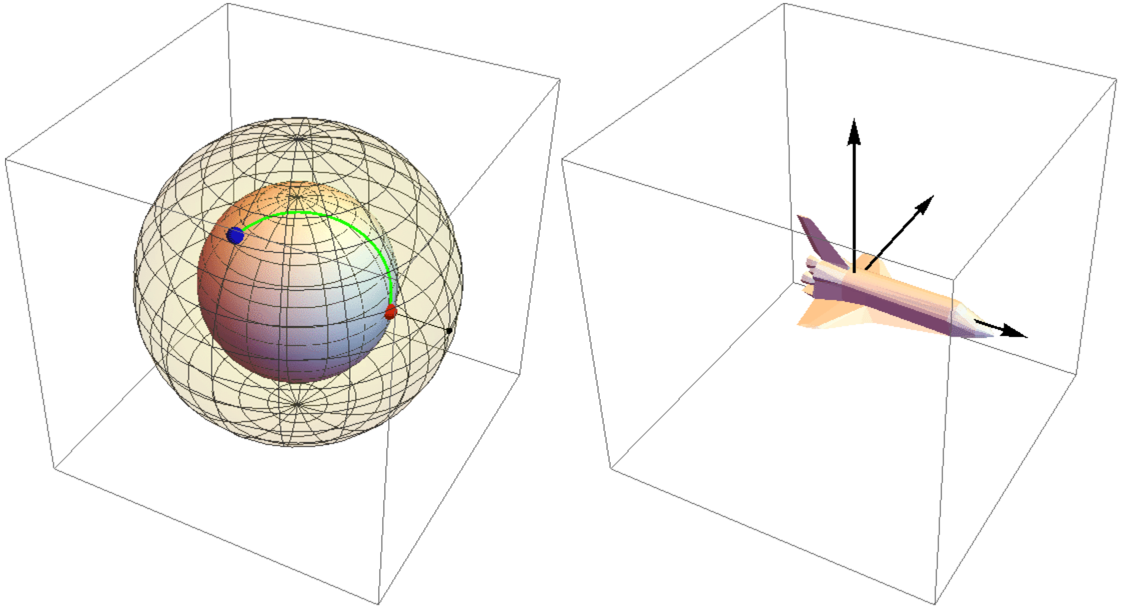


Interpolation from g_0 to g_1 via exp



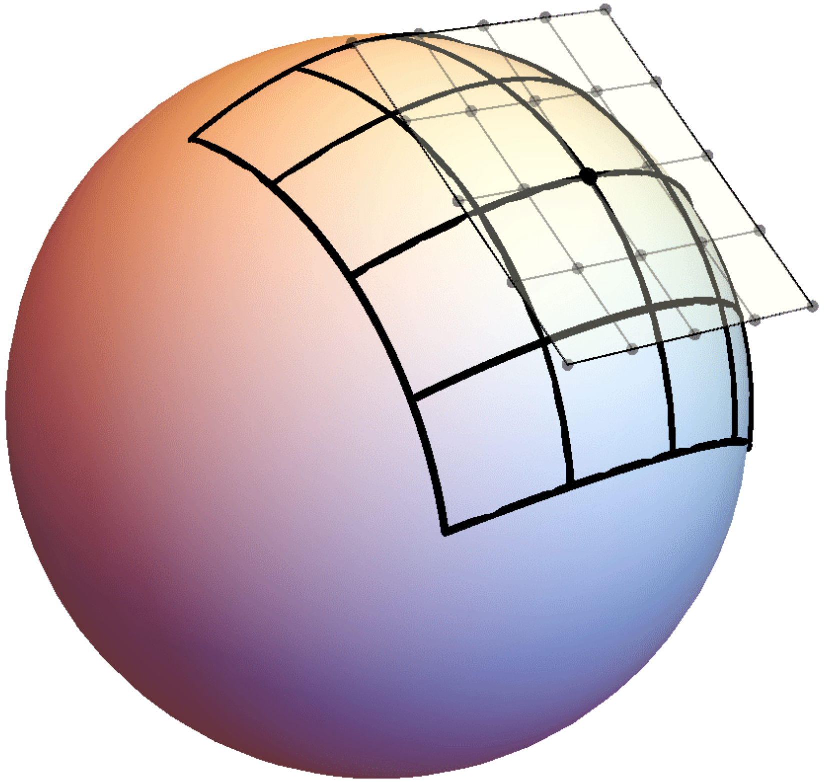
$$\gamma_{g_0,\mathbf{c}}(t) = g_0 \cdot \exp \mathbf{c}\, t$$

$$\mathbf{c} = \log g_0^{-1} \cdot g_1$$



Link: B-Splines on S2

B-splines on quotient groups require symmetry constraints



Backup slides

Equivariance diagram with actual results

Real example (rotation invariant features)