Group Equivariant Deep Learning

Lecture 1 - Regular group convolutions

Lecture 1.5 - A brief history of G-CNNs

Erik Bekkers, Amsterdam Machine Learning Lab, University of Amsterdam

This mini-course serves as a module with the UvA Master AI course Deep Learning 2 https://uvadl2c.github.io/

G-CNNs rule!

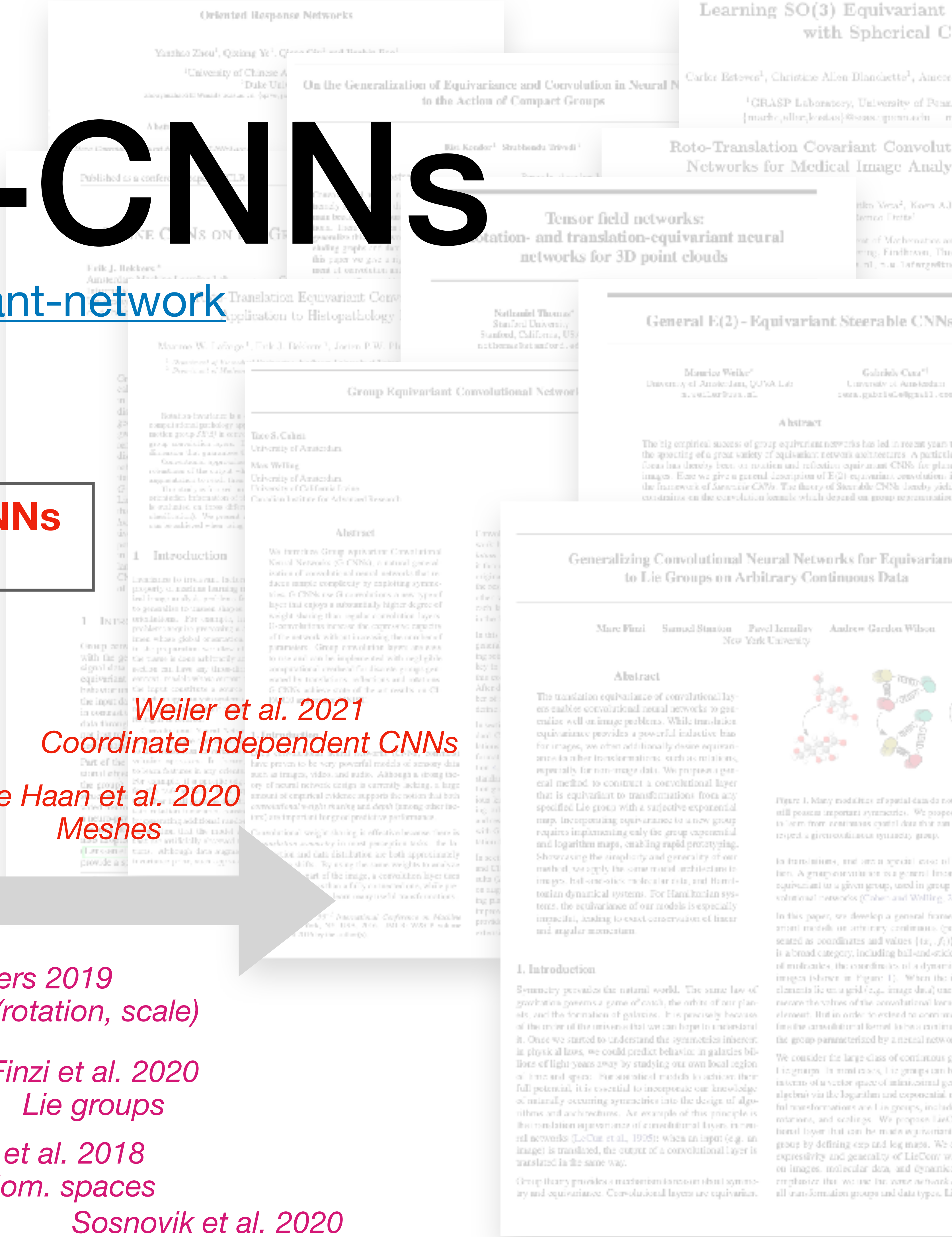
- The right inductive bias: guaranteed equivariance
  (no loss of information)
- Performance gains that can't be obtained by data-augmentation alone
  (both local and global equivariance/invariance)
- Increased sample efficiency
  (increased weight sharing, no geometric augmentation necessary)

G-CNNs rule!

- The right inductive bias: guaranteed equivariance
  (no loss of information)
- Performance gains that can't be obtained by data-augmentation alone
  (both local and global equivariance/invariance)
- Increased sample efficiency
  (increased weight sharing, no geometric augmentation necessary)

A brief history of G-CNNs

https://github.com/Chen-Cai-OSU/awesome-equivariant-network



Discrete G-CNNs
(square/cube symmetries)

Gauge-equivariant CNNs
(manifolds)

Cohen-Welling 2016
p4m

Winkels-Cohen 2018
3D discrete roto-transl.

Hoogeboom et al. 2018
SE(2,6)

Dieleman et al. 2016
p4m

Worrall-Brostow 2018
3D discrete roto-transl.

Cohen et al. 2019
Icosahedron

Weiler et al. 2021
Coordinate Independent CNNs

de Haan et al. 2020
Meshes

NNs → CNNs → G-CNNs

LeCun et al. 1990

Mallat et al. 2013, 2015
SE(2)

Bekkers et al. 2014-2018
SE(2)

Weiler et al. 2017
SE(2)

Zhou et al. 2017
SE(2)

Worrall et al. 2017
SE(2)

Esteves 2017
SO(3)

Kondor 2018
SE(3)

Thomas et al. 2018
SE(3)

Cohen et al. 2018
SE(3)

Kondor-Trivedi 2018
SE(3)

Weiler et al. 2018
SE(3)

Weiler-Cesa 2019
SE(2)

Bekkers 2019
Lie groups (rotation, scale)

Finzi et al. 2020
Lie groups

Chakraborty et al. 2018
Riemannian Hom. spaces

Sosnovik et al. 2020
Scale-translation

Continuous rotation G-CNNs
(regular discretizations)

Continuous rotation G-CNNs
(steerable)

Continuous G-CNNs
(Lie groups)

Cesa-Lang-Weiler 2022
$G = \mathbb{R}^d \rtimes H$ *with* H *compact*

https://quva-lab.github.io/escnn/

A brie… NNs

http… …ork

Disc… (square/…

Cohen-Welling p4m

Die…

NNs → CNNs →

LeCun et al. 1990

Mallat et al. 20… SE(2)

Bekke…

W…

Continuous… (regular…

Group Equivariant Convolutional Networks

Taco S. Cohen T.S.COHEN@UVA.NL
University of Amsterdam

Max Welling M.WELLING@UVA.NL
University of Amsterdam
University of California Irvine
Canadian Institute for Advanced Research

Abstract

We introduce Group equivariant Convolutional Neural Networks (G-CNNs), a natural generalization of convolutional neural networks that reduces sample complexity by exploiting symmetries. G-CNNs use G-convolutions, a new type of layer that enjoys a substantially higher degree of weight sharing than regular convolution layers. G-convolutions increase the expressive capacity of the network without increasing the number of parameters. Group convolution layers are easy to use and can be implemented with negligible computational overhead for discrete groups generated by translations, reflections and rotations. G-CNNs achieve state of the art results on CIFAR10 and rotated MNIST.

1. Introduction

Deep convolutional neural networks (CNNs, convnets) have proven to be very powerful models of sensory data such as images, video, and audio. Although a strong theory of neural network design is currently lacking, a large amount of empirical evidence supports the notion that both *convolutional weight sharing* and *depth* (among other factors) are important for good predictive performance.

Convolutional weight sharing is effective because there is a *translation symmetry* in most perception tasks: the label function and data distribution are both approximately invariant to shifts. By using the same weights to analyze or model each part of the image, a convolution layer uses far fewer parameters than a fully connected one, while preserving the capacity to learn many useful transformations.

Proceedings of the 33rd International Conference on Machine Learning, New York, NY, USA, 2016. JMLR: W&CP volume 48.

Convolution layers can be used effectively in a *deep* network because all the layers in such a network are *translation equivariant*: shifting the image and then feeding it through a number of layers is the same as feeding the original image through the same layers and then shifting the resulting feature maps (at least up to edge-effects). In other words, the symmetry (translation) is preserved by each layer, which makes it possible to exploit it not just in the first, but also in higher layers of the network.

In this paper we show how convolutional networks can be generalized to exploit larger groups of symmetries, including rotations and reflections. The notion of equivariance is key to this generalization, so in section 2 we will discuss this concept and its role in deep representation learning. After discussing related work in section 3, we recall a number of mathematical concepts in section 4 that allow us to define and analyze the G-convolution in a generic manner.

In section 5, we analyze the equivariance properties of standard CNNs, and show that they are equivariant to translations but may fail to equivary with more general transformations. Using the mathematical framework from section 4, we can define G-CNNs (section 6) by analogy to standard CNNs (the latter being the G-CNN for the translation group). We show that G-convolutions, as well as various kinds of layers used in modern CNNs, such as pooling, arbitrary pointwise nonlinearities, batch normalization and residual blocks are all equivariant, and thus compatible with G-CNNs. In section 7 we provide concrete implementation details for group convolutions.

In section 8 we report experimental results on MNIST-rot and CIFAR10, where G-CNNs achieve state of the art results (2.28% error on MNIST-rot, and 4.19% resp. 6.46% on augmented and plain CIFAR10). We show that replacing planar convolutions with G-convolutions consistently improves results without additional tuning. In section 9 we provide a discussion of these results and consider several extensions of the method, before concluding in section 10.

Learning SO(3) Equivariant … with Spherical C…

Roto-Translation Covariant Convolut… Networks for Medical Image Analy…

Tensor field networks: …tation- and translation-equivariant neural networks for 3D point clouds

General E(2)-Equivariant Steerable CNNs

Generalizing Convolutional Neural Networks for Equivarian… to Lie Groups on Arbitrary Continuous Data

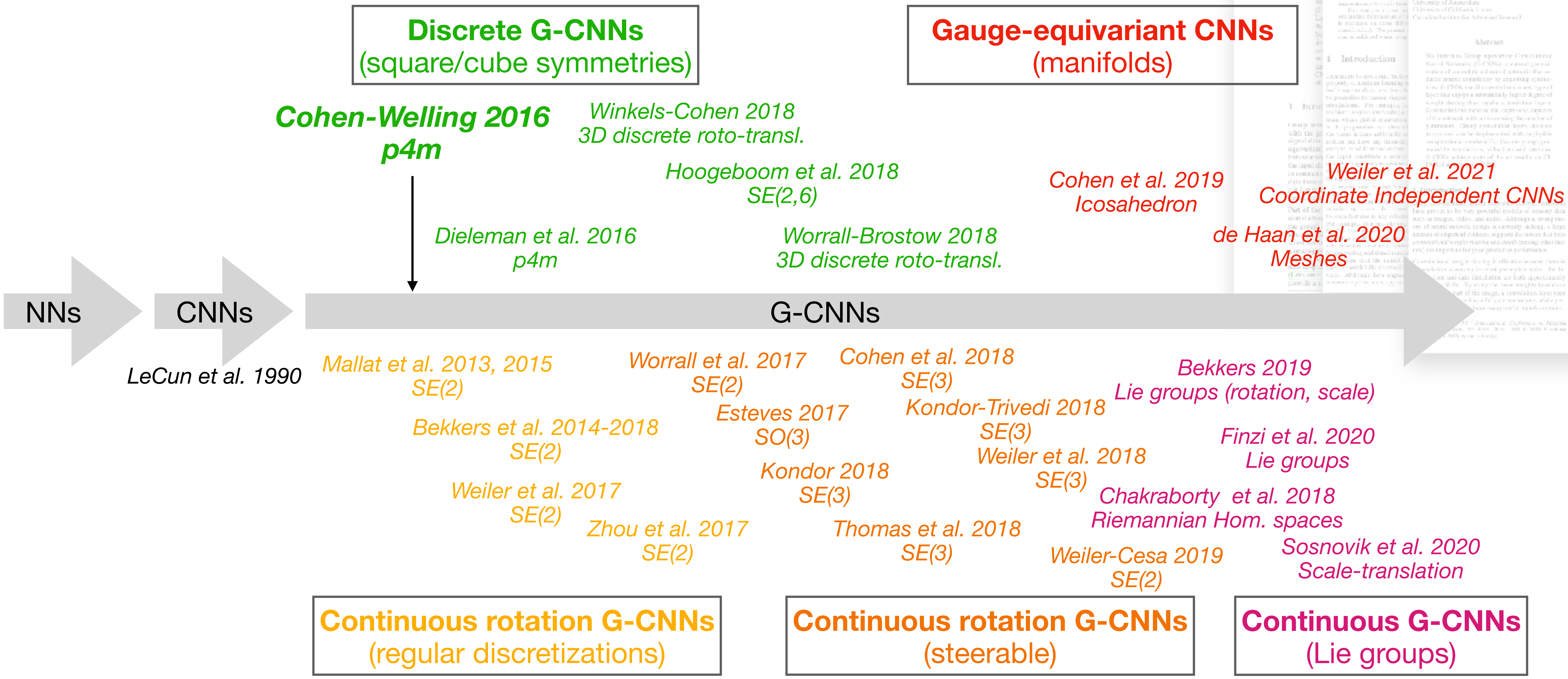
…ler et al. 2021
…e Independent CNNs
2020

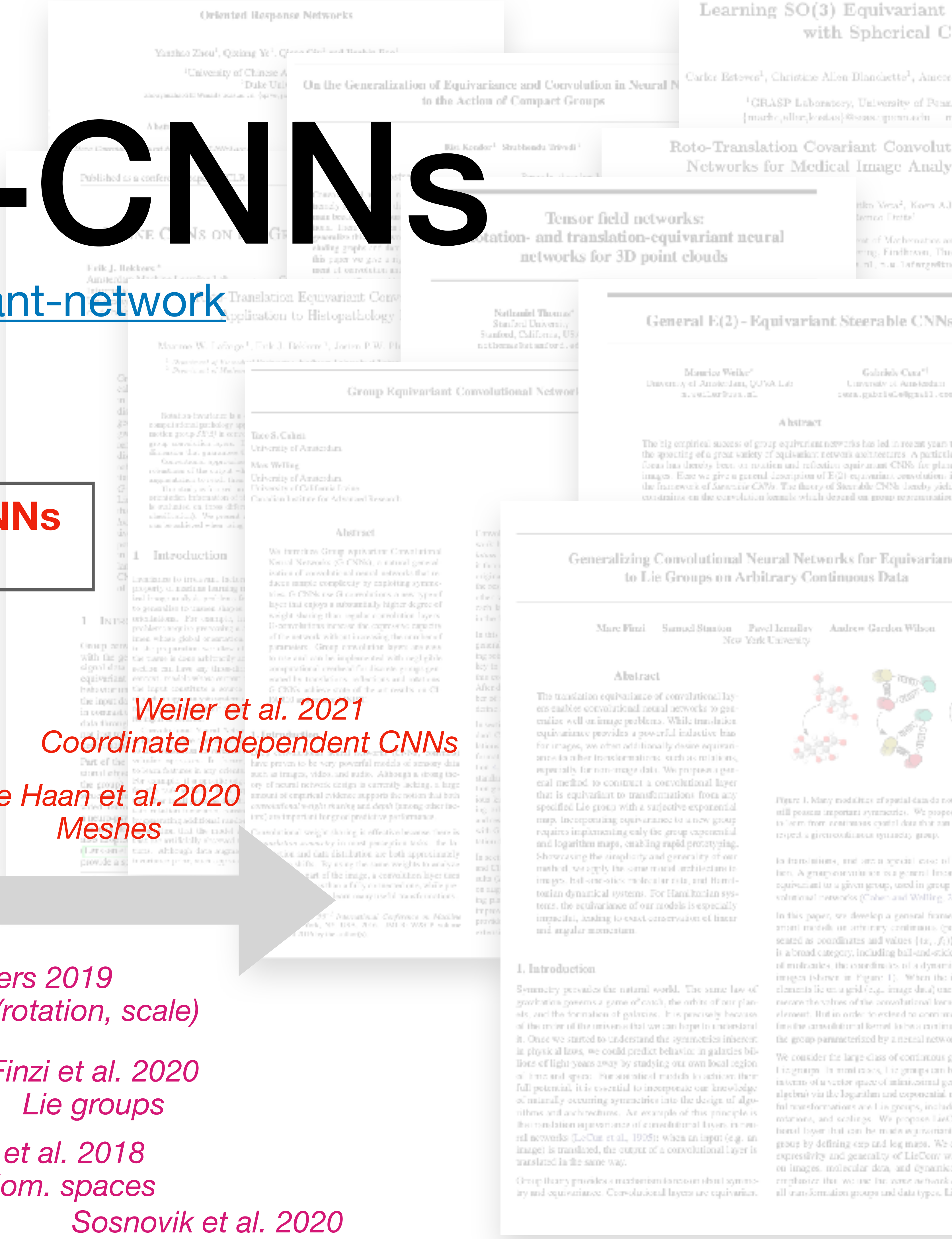
…k et al. 2020
…translation

…ous G-CNNs (…groups)

A brief history of G-CNNs

https://github.com/Chen-Cai-OSU/awesome-equivariant-network





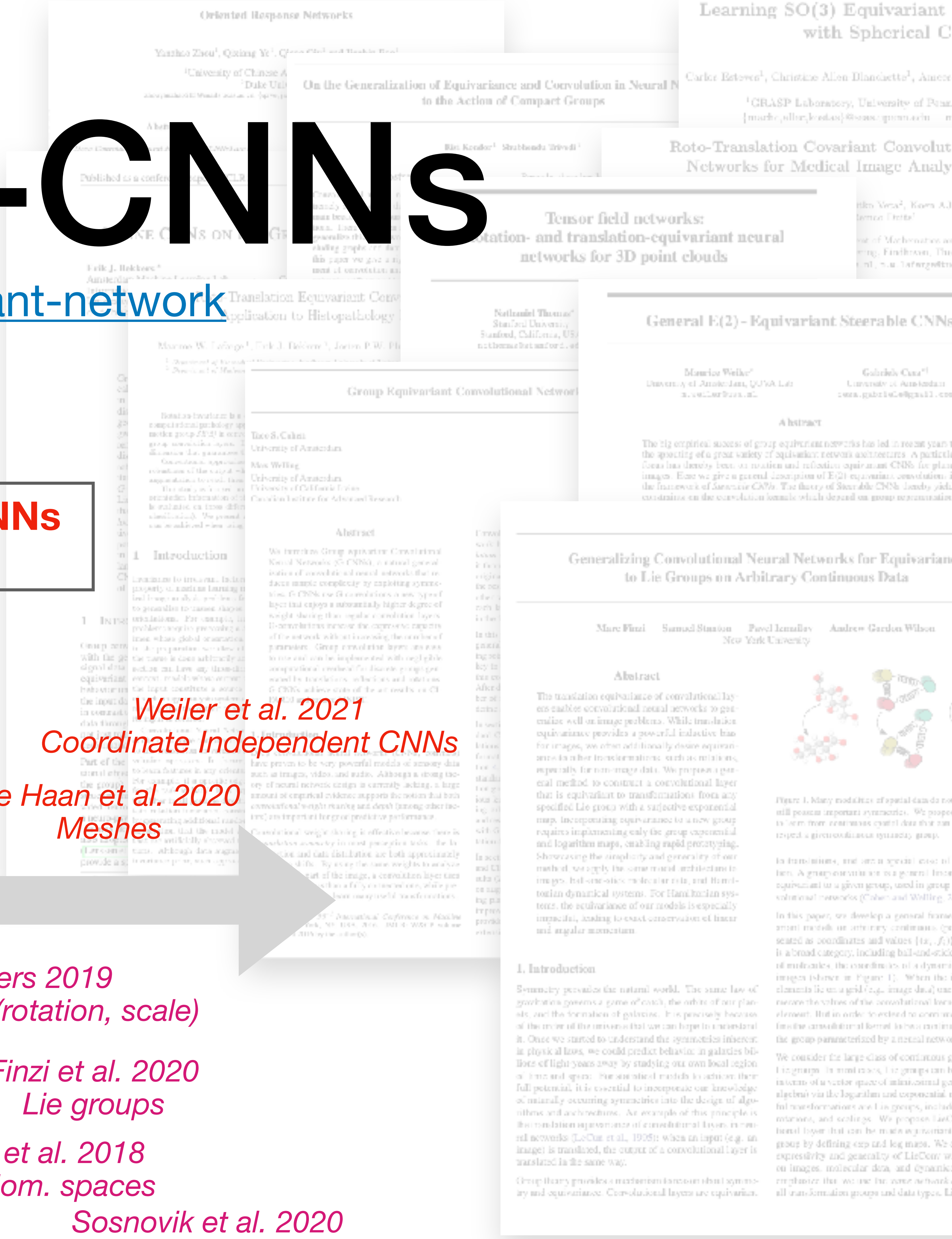
Cesa-Lang-Weiler 2022

$G = \mathbb{R}^d \rtimes H$ with H compact

https://quva-lab.github.io/escnn/

A brief history of G-CNNs

https://github.com/Chen-Cai-OSU/awesome-equivariant-network



Discrete G-CNNs
(square/cube symmetries)

Gauge-equivariant CNNs
(manifolds)

Cohen-Welling 2016
p4m

Winkels-Cohen 2018
3D discrete roto-transl.

Hoogeboom et al. 2018
SE(2,6)

Dieleman et al. 2016
p4m

Worrall-Brostow 2018
3D discrete roto-transl.

Cohen et al. 2019
Icosahedron

Weiler et al. 2021
Coordinate Independent CNNs

de Haan et al. 2020
Meshes

NNs → CNNs → G-CNNs

LeCun et al. 1990

Mallat et al. 2013, 2015
SE(2)

Bekkers et al. 2014-2018
SE(2)

Weiler et al. 2017
SE(2)

Zhou et al. 2017
SE(2)

Worrall et al. 2017
SE(2)

Esteves 2017
SO(3)

Kondor 2018
SE(3)

Thomas et al. 2018
SE(3)

Cohen et al. 2018
SE(3)

Kondor-Trivedi 2018
SE(3)

Weiler et al. 2018
SE(3)

Weiler-Cesa 2019
SE(2)

Bekkers 2019
Lie groups (rotation, scale)

Finzi et al. 2020
Lie groups

Chakraborty et al. 2018
Riemannian Hom. spaces

Sosnovik et al. 2020
Scale-translation

Continuous rotation G-CNNs
(regular discretizations)

Continuous rotation G-CNNs
(steerable)

Continuous G-CNNs
(Lie groups)

Cesa-Lang-Weiler 2022
$G = \mathbb{R}^d \rtimes H$ *with* H *compact*

https://quva-lab.github.io/escnn/

A brief history of G-CNNs

https://github.com/Chen-Cai-OSU/awesome-equivariant-network

NNs

arXiv:1804.04656v1 [cs.LG] 12 Apr 2018

3D G-CNNs for Pulmonary Nodule Detection

Marysia Winkels
University of Amsterdam / Aidence
marysia@aidence.com

Taco S. Cohen
University of Amsterdam
taco.cohen@gmail.com

Abstract

Convolutional Neural Networks (CNNs) require a large amount of annotated data to learn from, which is often difficult to obtain in the medical domain. In this paper we show that the sample complexity of CNNs can be significantly improved by using 3D roto-translation group convolutions (G-Convs) instead of the more conventional translational convolutions. These 3D G-CNNs were applied to the problem of false positive reduction for pulmonary nodule detection, and proved to be substantially more effective in terms of performance, sensitivity to malignant nodules, and speed of convergence compared to a strong and comparable baseline architecture with regular convolutions, data augmentation and a similar number of parameters. For every dataset size tested, the G-CNN achieved a FROC score close to the CNN trained on *ten times* more data.

1 Introduction

Lung cancer is currently the leading cause of cancer-related death worldwide, accounting for an estimated 1.7 million deaths globally each year and 270,000 in the European Union alone (1; 2), taking more victims than breast cancer, colon cancer and prostate cancer combined (3). This high mortality rate can be largely attributed to the fact that the majority of lung cancer is diagnosed when the cancer has already metastasised as symptoms generally do not present themselves until the cancer is at a late stage, making early detection difficult (4).

Screening of high risk groups could potentially increase early detection and thereby improve the survival rate (5; 6). However, the (cost-) effectiveness of screening would be largely dependent on the skill, alertness and experience level of the reading radiologists, as potentially malignant lesions are easy to overlook due to the rich vascular structure of the lung (see Figure 1). A way to reduce observational oversights would be to use second readings (7; 8), a practice in which two readers independently interpret an image and combine findings, but this would also drastically add to the already increasing workload of the radiologist (9), and increase the cost of care. Thus, a potentially much more cost-effective and accurate approach would be to introduce computer aided detection (CAD) software as a second reader to assist in the detection of lung nodules (10; 11).

Figure 1: Lung nodule on axial thorax CT

For medical image analysis, deep learning and in particular the Convolutional Neural Network (CNN) has become the methodology of choice. With regards to pulmonary nodule detection specifically, deep learning techniques for *candidate generation* and *false positive reduction* unambiguously outperform classical machine learning approaches (12; 13; 14). Convolutional neural networks, however, typically require a substantial amount of labeled data to train on – something that is scarce

Parts of this paper appeared previously in the first author's thesis.

1st Conference on Medical Imaging with Deep Learning (MIDL 2018), Amsterdam, The Netherlands.

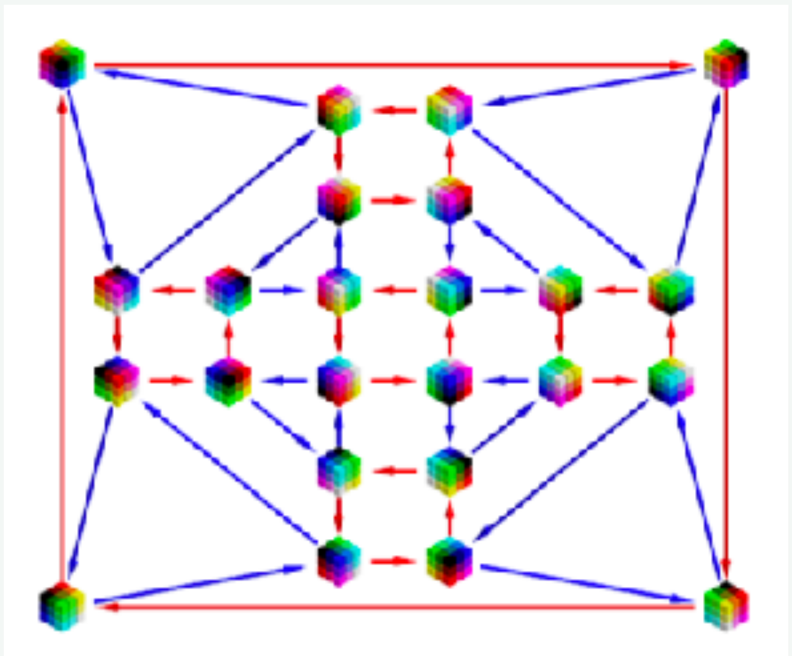


Figure 3: Cayley diagram for O. Red arrows correspond to Z-axis rotation, whereas blue arrows correspond to rotation around a diagonal axis. Best viewed in color.

This ECCV 2018 paper, provided here by the Computer Vision Foundation, is the author-created version. The content of this paper is identical to the content of the officially published ECCV 2018 LNCS version of the paper as available on SpringerLink: https://link.springer.com/conference/eccv

CubeNet: Equivariance to 3D Rotation and Translation

Daniel Worrall[1 0000-0002-9810-0709] and Gabriel Brostow[1 0000-0001-8472-3828]

Computer Science Department, University College London, UK
{d.worrall,g.brostow}@cs.ucl.ac.uk

Abstract. 3D Convolutional Neural Networks are sensitive to transformations applied to their input. This is a problem because a voxelized version of a 3D object, and its rotated clone, will look unrelated to each other after passing through to the last layer of a network. Instead, an idealized model would preserve a meaningful representation of the voxelized object, while explaining the pose-difference between the two inputs. An equivariant representation vector has two components: the invariant identity part, and a discernable encoding of the transformation. Models that can't explain pose-differences risk "diluting" the representation, in pursuit of optimizing a classification or regression loss function.

We introduce a Group Convolutional Neural Network with linear equivariance to translations *and* right angle rotations in three dimensions. We call this network *CubeNet*, reflecting its cube-like symmetry. By construction, this network helps preserve a 3D shape's global and local signature, as it is transformed through successive layers. We apply this network to a variety of 3D inference problems, achieving state-of-the-art on the ModelNet10 classification challenge, and comparable performance on the ISBI 2012 Connectome Segmentation Benchmark. To the best of our knowledge, this is the first 3D rotation equivariant CNN for voxel representations.

Keywords: Deep Learning, Equivariance, 3D Representations

1 Introduction

Convolutional neural networks (CNNs) are the go-to model for most prediction-ba... computer vision problems. However, most popularized CNNs are treated as bla... boxes, lacking interpretability and simple properties concerning the data doma... they act on. For instance, in 3D object recognition, we know that object categories *invariant* to object pose, but convolutional neural network filters are orientation, sc... reflection, and parity (point reflection) selective. This means that every activat... in any intermediate layer is sensitive to local pose, and ultimately the global out... of the network is too. A simple solution to obtain this sought-after invariance is augment the input data with transformed copies, spanning all possible variati... to which we seek to be invariant [2]. This method is simple and effective, but re... on an efficient and realistic data augmentation pipeline. There is also the argum...

8 D. Worrall and G. Brostow

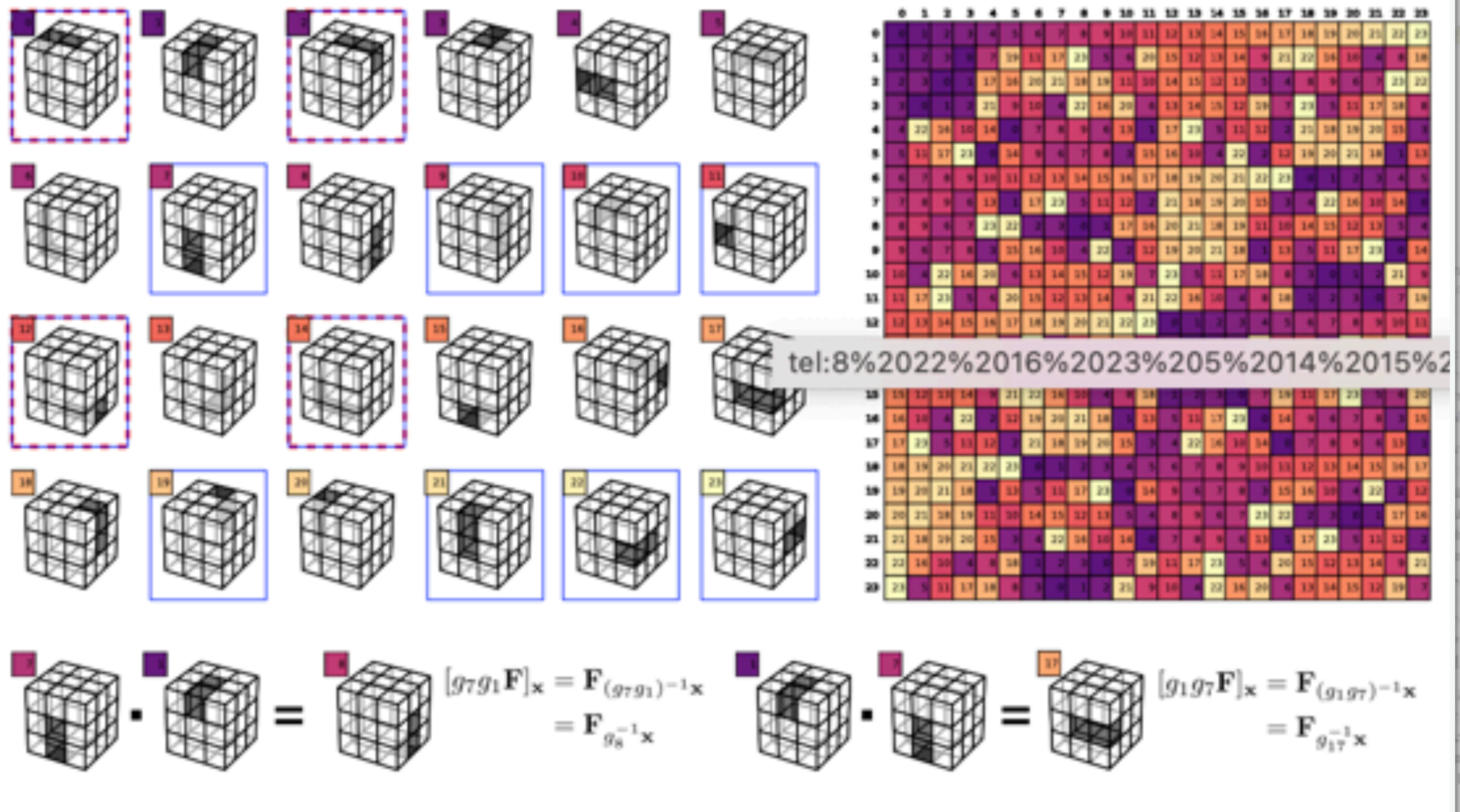


Fig. 2. (Best viewed in color) LEFT: The 24 rotations of the cube group S_4, applied to the a cube $\mathbf{F_x}$ are shown. For instance, rotation g_{22} applied to the cube returns $\mathbf{F}_{g_{22}^{-1}\mathbf{x}}$, shown by the #22 in the bottom row. The 12 cubes wrapped in thin blue boxes are the rotational tetrahedral group T_4. The 4 cubes wrapped in thick dashed red lines are the Klein four-group V. RIGHT: The Cayley table of the cube group, representing how rotations are composed. For instance, on the BOTTOM LEFT, we have the example of composing rotation g_7 with rotation g_1. The composition is performed by i) first applying g_7 to the cube to yield $\mathbf{F}_{g_7^{-1}\mathbf{x}}$ then ii) applying g_1 to $\mathbf{F}_{g_7^{-1}\mathbf{x}}$, returning $\mathbf{F}_{g_1^{-1}g_7^{-1}\mathbf{x}}$. The first transformation is easy to visualize - it is by #7 in the grid of cubes. The transformation g_1 is a rotation by 90° counter-clockwise about the vertical axis, thus for the composition we rotate $\mathbf{F}_{g_7^{-1}\mathbf{x}}$ 90° counter-clockwise about the z-axis. This results in $F_{g_8^{-1}\mathbf{x}}$. This result is stored in the Cayley table by placing the first rotation down the left column and the second rotation along the top row. The intersection of row **7** with column **1** is the rotation **8**. On the BOTTOM RIGHT, we show the composition $g_7 g_1 = g_{17} \neq g_8 = g_1 g_7$, demonstrating the non-commutativity property of the cube group and 3D rotations in general.

Cesa-Lang-Weiler 2022
$G = \mathbb{R}^d \rtimes H$ with H compa...

https://quva-lab.github.io/escnn/

A brief history of G-CNNs

https://github.com/Chen-Cai-OSU/awesome-equivariant-network

Published as a conference paper at ICLR 2018

arXiv:1803.02108v1 [cs.LG] 6 Mar 2018

HexaConv

Emiel Hoogeboom*, Jorn W.T. Peters* & Taco S. Cohen
University of Amsterdam
{e.hoogeboom,j.w.t.peters,t.s.cohen}@uva.nl

Max Welling
University of Amsterdam & CIFAR
m.welling@uva.nl

Abstract

The effectiveness of convolutional neural networks stems in large part from their ability to exploit the translation invariance that is inherent in many learning problems. Recently, it was shown that CNNs can exploit other sources of invariance, such as rotation invariance, by using *group convolutions* instead of planar convolutions. However, for reasons of performance and ease of implementation, it has been necessary to limit the group convolution to transformations that can be applied to the filters without interpolation. Thus, for images with square pixels, only integer translations, rotations by multiples of 90 degrees, and reflections are admissible.

Whereas the square tiling provides a 4-fold rotational symmetry, a hexagonal tiling of the plane has a 6-fold rotational symmetry. In this paper we show how one can efficiently implement planar convolution and group convolution over hexagonal lattices, by re-using existing highly optimized convolution routines. We find that, due to the reduced anisotropy of hexagonal filters, planar HexaConv provides better accuracy than planar convolution with square filters, given a fixed parameter budget. Furthermore, we find that the increased degree of symmetry of the hexagonal grid increases the effectiveness of group convolutions, by allowing for more parameter sharing. We show that our method significantly outperforms conventional CNNs on the AID aerial scene classification dataset, even outperforming ImageNet pretrained models.

1 Introduction

For sensory perception tasks, neural networks have mostly replaced handcrafted features. Instead of defining features by hand using domain knowledge, it is now possible to learn them, resulting in improved accuracy and saving a considerable amount of work. However, successful generalization is still critically dependent on the inductive bias encoded in the network architecture, whether this bias is understood by the network architect or not.

The canonical example of a successful network architecture is the Convolutional Neural Network (CNN, ConvNet). Through convolutional weight sharing, these networks exploit the fact that a given visual pattern may appear in different locations in the image with approximately equal likelihood. Furthermore, this translation symmetry is preserved throughout the network, because a translation of the input image leads to a translation of the feature maps at each layer: convolution is translation equivariant.

Very often, the true label function (the mapping from image to label that we wish to learn) is invariant to more transformations than just translations. Rotations are an obvious example, but standard translational convolutions cannot exploit this symmetry, because they are not rotation equivariant. As it turns out, a convolution operation can be defined for almost any group of transformation — not just translations. By simply replacing convolutions with group convolutions (wherein filters are not

*Equal contribution

1

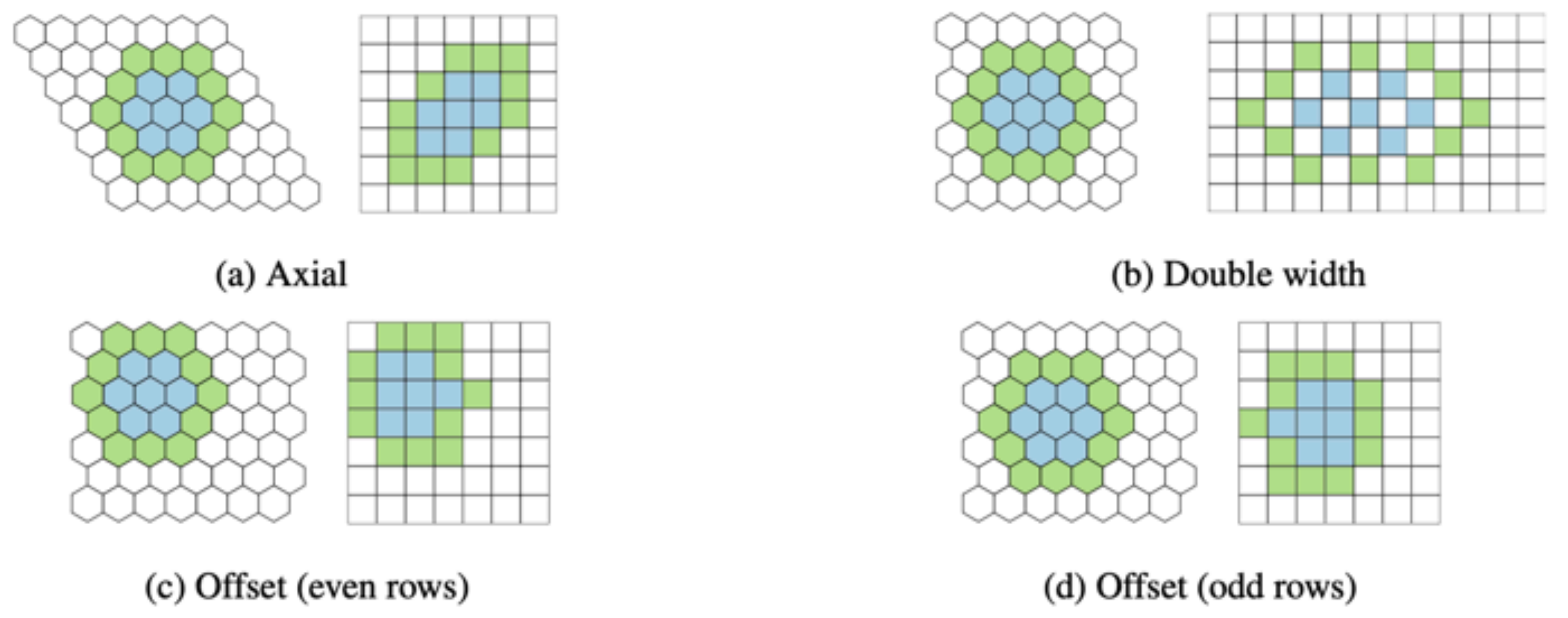


Figure 4: Hexagonal convolution filters (left) represented in 2D memory (right) for filters of size three (blue) and five (blue and green). Standard 2D convolution using both feature map and filter stored according to the coordinate system is equivalent to convolution on the hexagonal lattice. Note that for the offset coordinate system two separate planar convolution are required — one for even and one for odd rows.

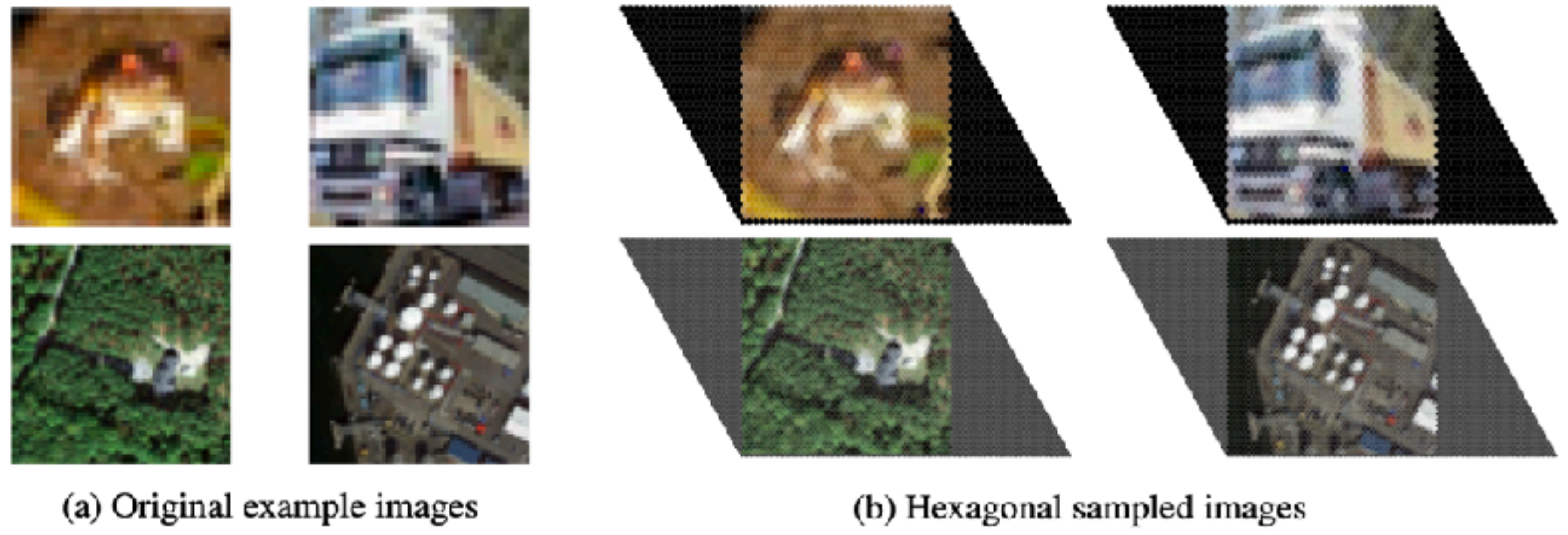
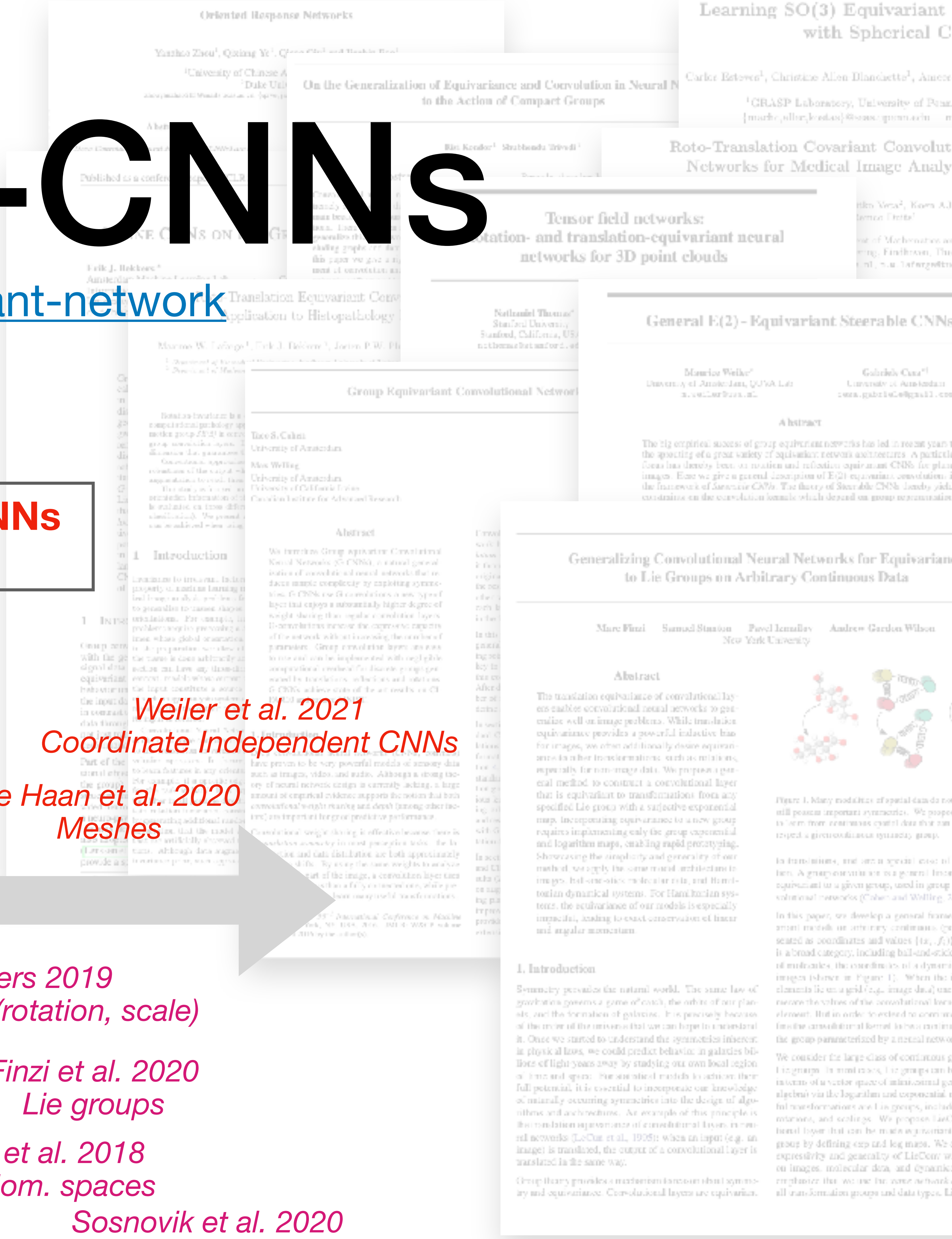


Figure 6: CIFAR-10 (top) and AID (bottom) examples sampled from Cartesian to hexagonal axial coordinates. Zero padding enlarges the images in axial systems.

NNs

A brief history of G-CNNs

https://github.com/Chen-Cai-OSU/awesome-equivariant-network



Discrete G-CNNs
(square/cube symmetries)

Gauge-equivariant CNNs
(manifolds)

Cohen-Welling 2016
p4m

Winkels-Cohen 2018
3D discrete roto-transl.

Hoogeboom et al. 2018
SE(2,6)

Dieleman et al. 2016
p4m

Worrall-Brostow 2018
3D discrete roto-transl.

Cohen et al. 2019
Icosahedron

Weiler et al. 2021
Coordinate Independent CNNs

de Haan et al. 2020
Meshes

NNs → CNNs → G-CNNs

LeCun et al. 1990

Mallat et al. 2013, 2015
SE(2)

Bekkers et al. 2014-2018
SE(2)

Weiler et al. 2017
SE(2)

Zhou et al. 2017
SE(2)

Worrall et al. 2017
SE(2)

Esteves 2017
SO(3)

Kondor 2018
SE(3)

Thomas et al. 2018
SE(3)

Cohen et al. 2018
SE(3)

Kondor-Trivedi 2018
SE(3)

Weiler et al. 2018
SE(3)

Weiler-Cesa 2019
SE(2)

Bekkers 2019
Lie groups (rotation, scale)

Finzi et al. 2020
Lie groups

Chakraborty et al. 2018
Riemannian Hom. spaces

Sosnovik et al. 2020
Scale-translation

Continuous rotation G-CNNs
(regular discretizations)

Continuous rotation G-CNNs
(steerable)

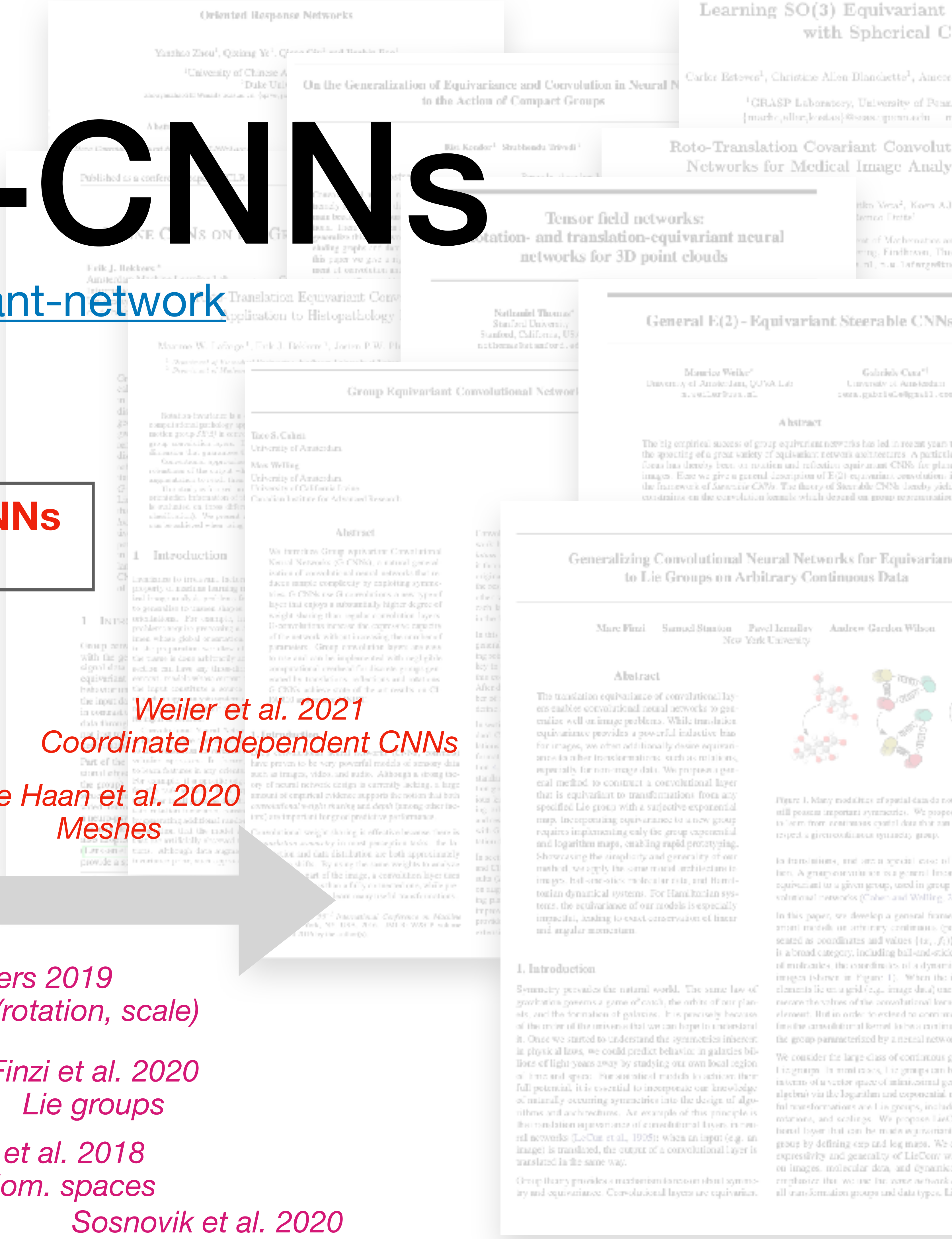
Continuous G-CNNs
(Lie groups)

Cesa-Lang-Weiler 2022
$G = \mathbb{R}^d \rtimes H$ *with* H *compact*

https://quva-lab.github.io/escnn/

A brief history of G-CNNs

https://github.com/Chen-Cai-OSU/awesome-equivariant-network



Discrete G-CNNs
(square/cube symmetries)

Gauge-equivariant CNNs
(manifolds)

Cohen-Welling 2016
p4m

Winkels-Cohen 2018
3D discrete roto-transl.

Hoogeboom et al. 2018
SE(2,6)

Dieleman et al. 2016
p4m

Worrall-Brostow 2018
3D discrete roto-transl.

Cohen et al. 2019
Icosahedron

Weiler et al. 2021
Coordinate Independent CNNs

de Haan et al. 2020
Meshes

NNs → CNNs → G-CNNs

LeCun et al. 1990

Mallat et al. 2013, 2015
SE(2)

Bekkers et al. 2014-2018
SE(2)

Weiler et al. 2017
SE(2)

Zhou et al. 2017
SE(2)

Worrall et al. 2017
SE(2)

Esteves 2017
SO(3)

Kondor 2018
SE(3)

Cohen et al. 2018
SE(3)

Kondor-Trivedi 2018
SE(3)

Weiler et al. 2018
SE(3)

Thomas et al. 2018
SE(3)

Weiler-Cesa 2019
SE(2)

Bekkers 2019
Lie groups (rotation, scale)

Finzi et al. 2020
Lie groups

Chakraborty et al. 2018
Riemannian Hom. spaces

Sosnovik et al. 2020
Scale-translation

Continuous rotation G-CNNs
(regular discretizations)

Continuous rotation G-CNNs
(steerable)

Continuous G-CNNs
(Lie groups)

Cesa-Lang-Weiler 2022
$G = \mathbb{R}^d \rtimes H$ *with* H *compact*

https://quva-lab.github.io/escnn/

A brief history of G-CNNs

arXiv:1804.03393v3 [cs.CV] 11 Jun 2018

Roto-Translation Covariant Convolutional Networks for Medical Image Analysis

Erik J Bekkers*[1], Maxime W Lafarge*[2], Mitko Veta[2], Koen AJ Eppenhof[2], Josien PW Pluim[2] and Remco Duits[1]

Eindhoven University of Technology, [1]Department of Mathematics and Computer Science and [2]Department of Biomedical Engineering, Eindhoven, The Netherlands
*Joint main authors - e.j.bekkers@tue.nl, m.w.lafarge@tue.nl

Abstract. We propose a framework for rotation and translation covariant deep learning using $SE(2)$ group convolutions. The group product of the special Euclidean motion group $SE(2)$ describes how a concatenation of two roto-translations results in a net roto-translation. We encode this geometric structure into convolutional neural networks (CNNs) via $SE(2)$ group convolutional layers, which fit into the standard 2D CNN framework, and which allow to generically deal with rotated input samples without the need for data augmentation.
We introduce three layers: a *lifting layer* which lifts a 2D (vector valued) image to an $SE(2)$-image, i.e., 3D (vector valued) data whose domain is $SE(2)$; a *group convolution layer* from and to an $SE(2)$-image; and a *projection layer* from an $SE(2)$-image to a 2D image. The lifting and group convolution layers are $SE(2)$ *covariant* (the output roto-translates with the input). The final projection layer, a maximum intensity projection over rotations, makes the full CNN rotation *invariant*.
We show with three different problems in histopathology, retinal imaging, and electron microscopy that with the proposed group CNNs, state-of-the-art performance can be achieved, without the need for data augmentation by rotation and with increased performance compared to standard CNNs that do rely on augmentation.

Keywords: Group convolutional network, roto-translation group, mitosis detection, vessel segmentation, cell boundary segmentation

1 Introduction

In this work we generalize $\mathbb{R}^2$ convolutional neural networks (CNNs) to $SE(2)$ group CNNs (G-CNNs) in which the data lives on position orientation space, and in which the convolution layers are defined in terms of representations of the special Euclidean motion group $SE(2)$. In essence this means that we replace the convolutions (with translations of a kernel) by $SE(2)$ group convolutions (with roto-translations of a kernel). The advantage of the proposed approach compared to standard $\mathbb{R}^2$ CNNs is that rotation covariance is encoded in the network design and does not have to be learned by the convolution kernels. E.g., a feature that may appear in the data under several orientations does not have

en-Cai-OSU/aweso

Gauge-e

2018
ransl.

oom et al. 2018
SE(2,6)

Worrall-Brostow 2018
3D discrete roto-transl.

G-

2017

eves 2
SO(3

Kond

Contents lists available at ScienceDirect
Medical Image Analysis
journal homepage: www.elsevier.com/locate/media

Roto-translation equivariant convolutional networks: Application to histopathology image analysis

Maxime W. Lafarge[a,*], Erik J. Bekkers[b], Josien P.W. Pluim[a], Remco Duits[b], Mitko Veta[a]

[a]Department of Biomedical Engineering, Eindhoven University of Technology, Eindhoven, the Netherlands
[b]Department of Mathematics and Computer Science, Eindhoven University of Technology, Eindhoven, the Netherlands

ARTICLE INFO

Article history:
Received 4 December 2019
Revised 3 June 2020
Accepted 14 August 2020
Available online 31 October 2020

Keywords:
Group convolutional neural network
Roto-translation equivariance
Computational pathology
Mitosis detection
Tumor detection
Nuclei segmentation

ABSTRACT

Rotation-invariance is a desired property of machine-learning models for medical image analysis and in particular for computational pathology applications. We propose a framework to encode the geometric structure of the special Euclidean motion group $SE(2)$ in convolutional networks to yield translation and rotation equivariance via the introduction of $SE(2)$-group convolution layers. This structure enables models to learn feature representations with a discretized orientation dimension that guarantees that their outputs are invariant under a discrete set of rotations.

Conventional approaches for rotation invariance rely mostly on data augmentation, but this does not guarantee the robustness of the output when the input is rotated. At that, trained conventional CNNs may require test-time rotation augmentation to reach their full capability.

This study is focused on histopathology image analysis applications for which it is desirable that the arbitrary global orientation information of the imaged tissues is not captured by the machine learning models. The proposed framework is evaluated on three different histopathology image analysis tasks (mitosis ... omparative analysis for each problem ... when using the proposed framework.

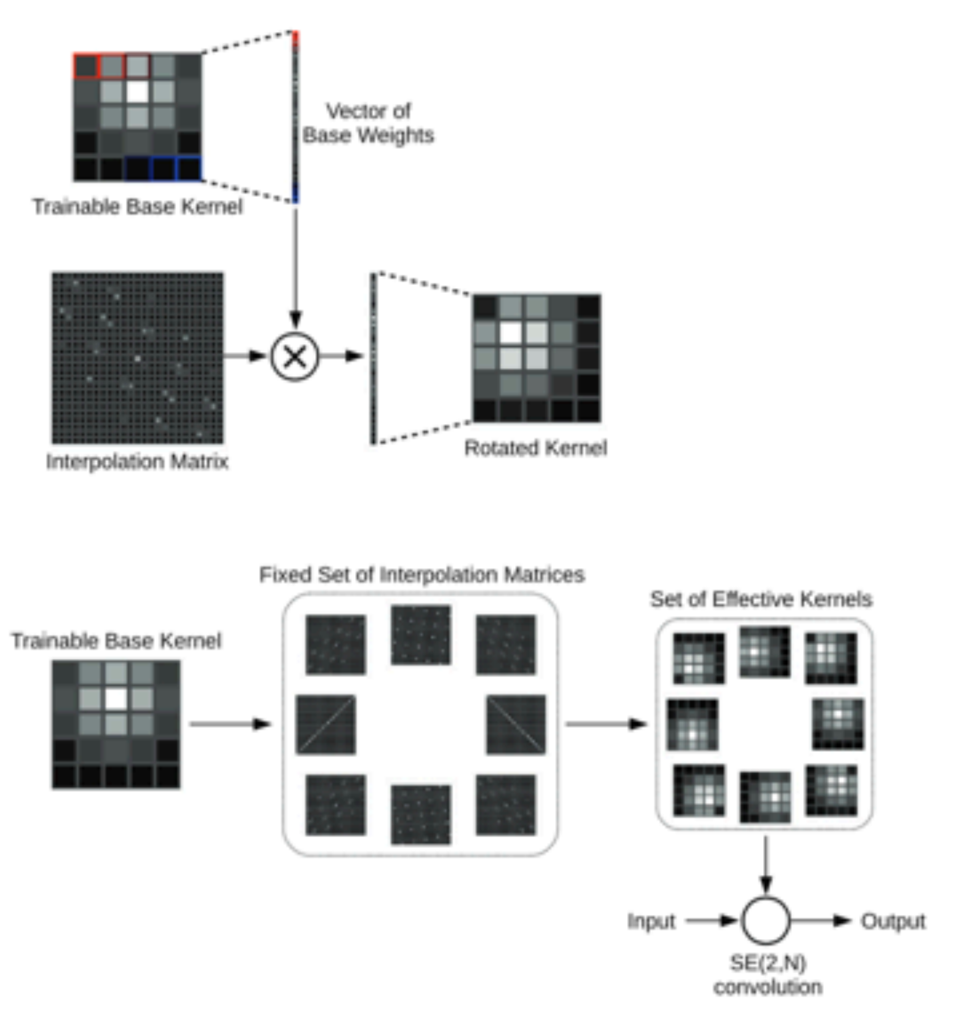


Fig. 2. Illustration of the process generating a rotated set of effective kernels from a trainable vector of base weights via the introduction of fixed interpolation matrix in the computational pipeline.

experiments that we used to analyze and validate them. In the construction of the G-CNNs we adhere to the following principle of group equivariant architecture design.

G-CNN design principle A sequence of layers starting with a lifting layer (Eq. (7)) and followed by one or more group convolution layers (Eq. (9)), possibly intertwined with point-wise non-linearities, results in the encoding of roto-translation equivariant feature maps. If such a block is followed by a projection layer (Eq. (10)) then the entire block results in a encoding of features that is guaranteed to be rotationally invariant. Our implementation of the G-CNN layers is available at https://github.com/tueimage/se2cnn.

4.1. Applications and model architectures

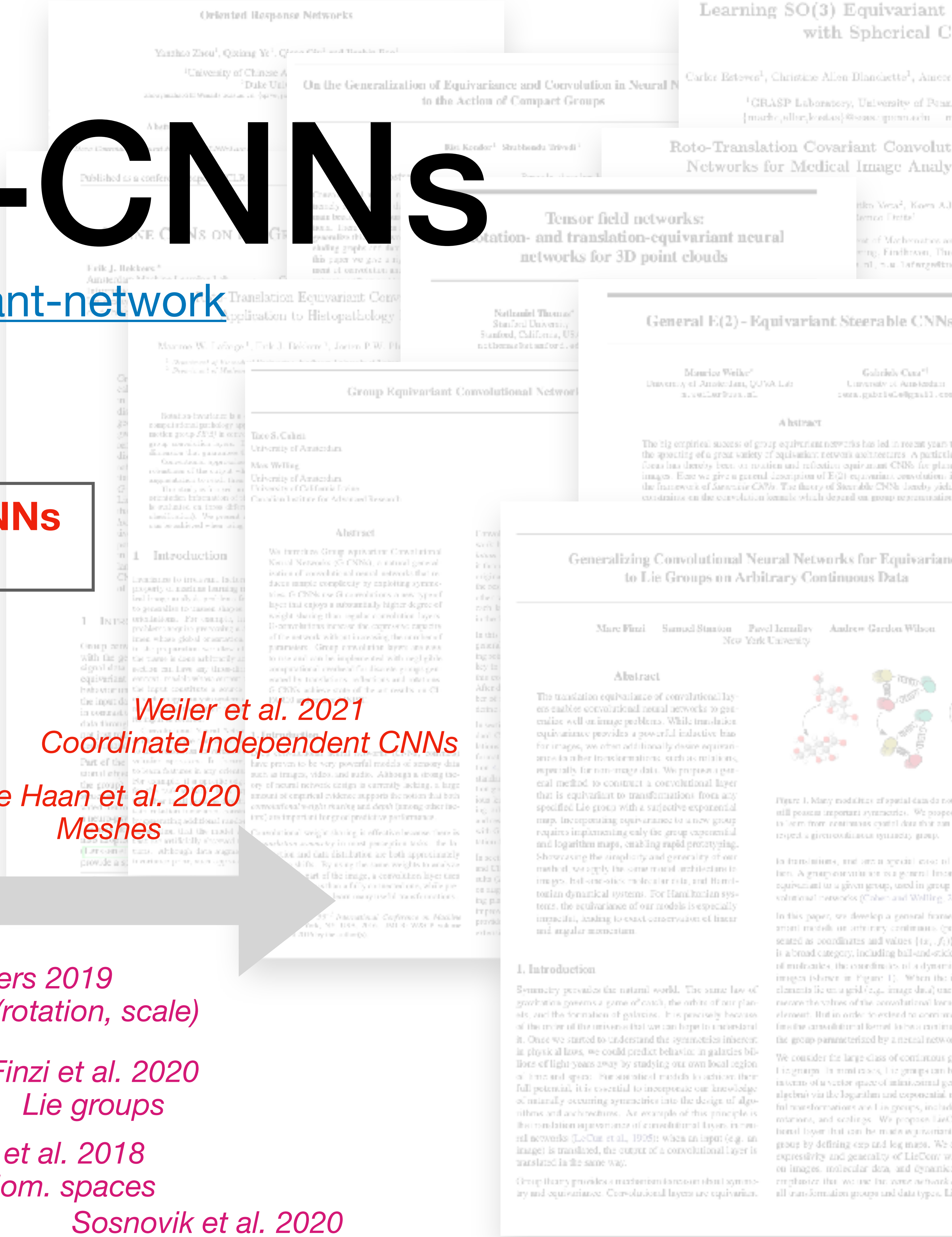
For each task introduced in Section 3.1 we conducted two experiments: first, we trained a set of variations of a baseline CNN, by changing the orientation sampling level N of their SE(2,N) layers, while keeping the total number of weights of each model approximately the same. Second, we trained each model with the reduced data regime counterparts of the training sets introduced in Section 3.1. For each task we opted for straight-forward architectures with a low number of parameters that were in-line with methods reported in the literature. This way, we propose new G-CNN baselines that facilitate comparative experiments and that can be extended to more sophisticated architectures.

Mitosis detection We used the mitosis classification model orig-

https://quva-lab.github.io/escnn/

A brief history of G-CNNs

https://github.com/Chen-Cai-OSU/awesome-equivariant-network



Discrete G-CNNs
(square/cube symmetries)

Gauge-equivariant CNNs
(manifolds)

Cohen-Welling 2016
p4m

Winkels-Cohen 2018
3D discrete roto-transl.

Hoogeboom et al. 2018
SE(2,6)

Dieleman et al. 2016
p4m

Worrall-Brostow 2018
3D discrete roto-transl.

Cohen et al. 2019
Icosahedron

Weiler et al. 2021
Coordinate Independent CNNs

de Haan et al. 2020
Meshes

NNs → CNNs → G-CNNs

LeCun et al. 1990

Mallat et al. 2013, 2015
SE(2)

Bekkers et al. 2014-2018
SE(2)

Weiler et al. 2017
SE(2)

Zhou et al. 2017
SE(2)

Worrall et al. 2017
SE(2)

Esteves 2017
SO(3)

Kondor 2018
SE(3)

Thomas et al. 2018
SE(3)

Cohen et al. 2018
SE(3)

Kondor-Trivedi 2018
SE(3)

Weiler et al. 2018
SE(3)

Weiler-Cesa 2019
SE(2)

Bekkers 2019
Lie groups (rotation, scale)

Finzi et al. 2020
Lie groups

Chakraborty et al. 2018
Riemannian Hom. spaces

Sosnovik et al. 2020
Scale-translation

Continuous rotation G-CNNs
(regular discretizations)

Continuous rotation G-CNNs
(steerable)

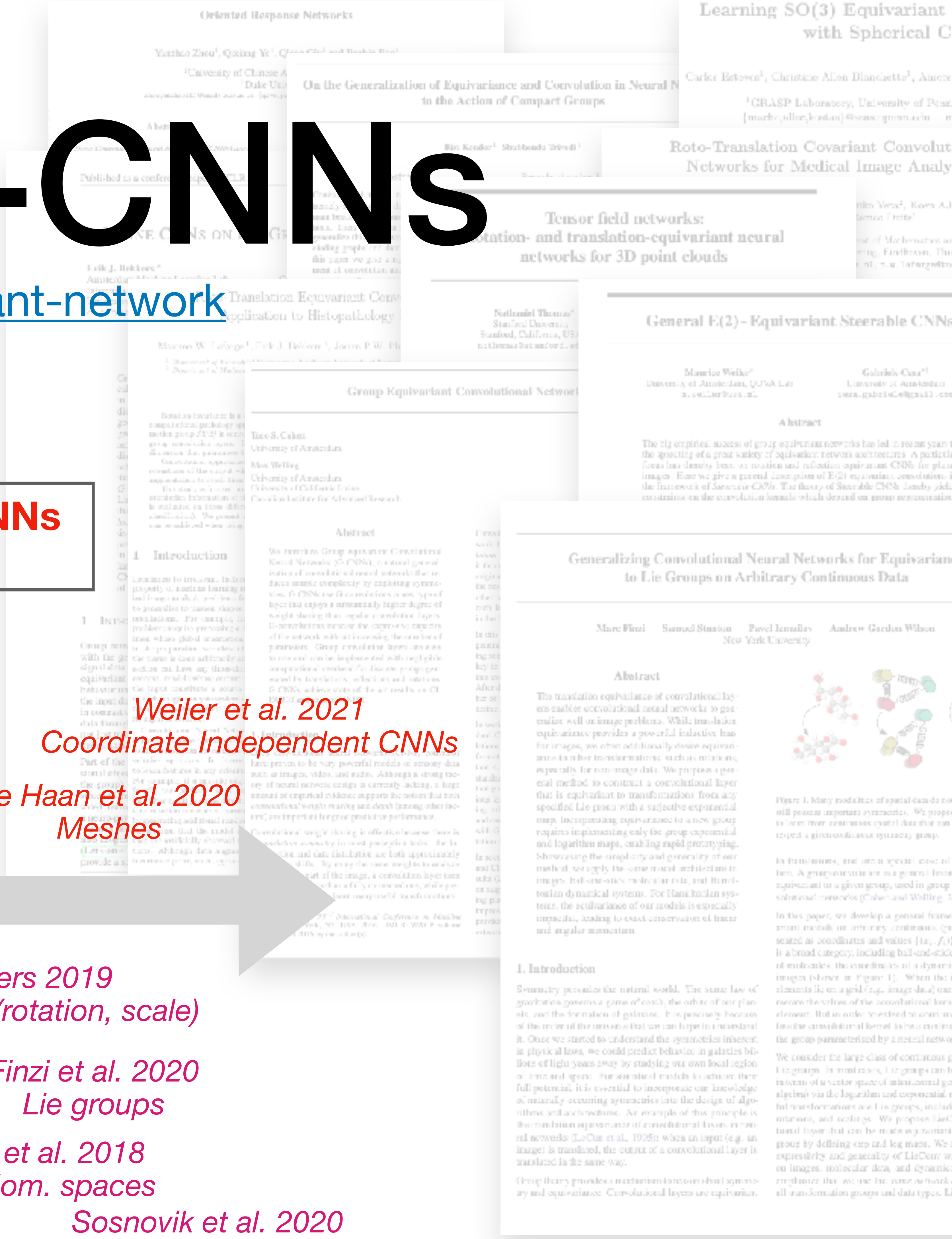
Continuous G-CNNs
(Lie groups)

Cesa-Lang-Weiler 2022
$G = \mathbb{R}^d \rtimes H$ *with* H *compact*

https://quva-lab.github.io/escnn/

A brief history of G-CNNs

https://github.com/Chen-Cai-OSU/awesome-equivariant-network



Discrete G-CNNs
(square/cube symmetries)

Gauge-equivariant CNNs
(manifolds)

Cohen-Welling 2016
p4m

Winkels-Cohen 2018
3D discrete roto-transl.

Hoogeboom et al. 2018
SE(2,6)

Dieleman et al. 2016
p4m

Worrall-Brostow 2018
3D discrete roto-transl.

Cohen et al. 2019
Icosahedron

Weiler et al. 2021
Coordinate Independent CNNs

de Haan et al. 2020
Meshes

NNs → CNNs → G-CNNs

LeCun et al. 1990

Mallat et al. 2013, 2015
SE(2)

Bekkers et al. 2014-2018
SE(2)

Weiler et al. 2017
SE(2)

Zhou et al. 2017
SE(2)

Worrall et al. 2017
SE(2)

Esteves 2017
SO(3)

Kondor 2018
SE(3)

Cohen et al. 2018
SE(3)

Kondor-Trivedi 2018
SE(3)

Weiler et al. 2018
SE(3)

Thomas et al. 2018
SE(3)

Weiler-Cesa 2019
SE(2)

Bekkers 2019
Lie groups (rotation, scale)

Finzi et al. 2020
Lie groups

Chakraborty et al. 2018
Riemannian Hom. spaces

Sosnovik et al. 2020
Scale-translation

Continuous rotation G-CNNs
(regular discretizations)

Continuous rotation G-CNNs
(steerable)

Continuous G-CNNs
(Lie groups)

Cesa-Lang-Weiler 2022
$G = \mathbb{R}^d \rtimes H$ *with* H *compact*

https://quva-lab.github.io/escnn/

A brief history of G-CNNs

[awesome-equivariant-network]

NNs → C

LeCun

This CVPR paper is the Open Access version, provided by the Computer Vision Foundation. Except for this watermark, it is identical to the version available on IEEE Xplore.

Harmonic Networks: Deep Translation and Rotation Equivariance

Daniel E. Worrall, Stephan J. Garbin, Daniyar Turmukhambetov and Gabriel J. Brostow
{d.worrall, s.garbin, d.turmukhambetov, g.brostow}@cs.ucl.ac.uk
University College London*

Abstract

Translating or rotating an input image should not affect the results of many computer vision tasks. Convolutional neural networks (CNNs) are already translation equivariant: input image translations produce proportionate feature map translations. This is not the case for rotations. Global rotation equivariance is typically sought through data augmentation, but patch-wise equivariance is more difficult. We present Harmonic Networks or H-Nets, a CNN exhibiting equivariance to patch-wise translation and 360-rotation. We achieve this by replacing regular CNN filters with circular harmonics, returning a maximal response and orientation for every receptive field patch.

H-Nets use a rich, parameter-efficient and fixed computational complexity representation, and we show that deep feature maps within the network encode complicated rotational invariants. We demonstrate that our layers are general enough to be used in conjunction with the latest architectures and techniques, such as deep supervision and batch normalization. We also achieve state-of-the-art classification on rotated-MNIST, and competitive results on other benchmark challenges.

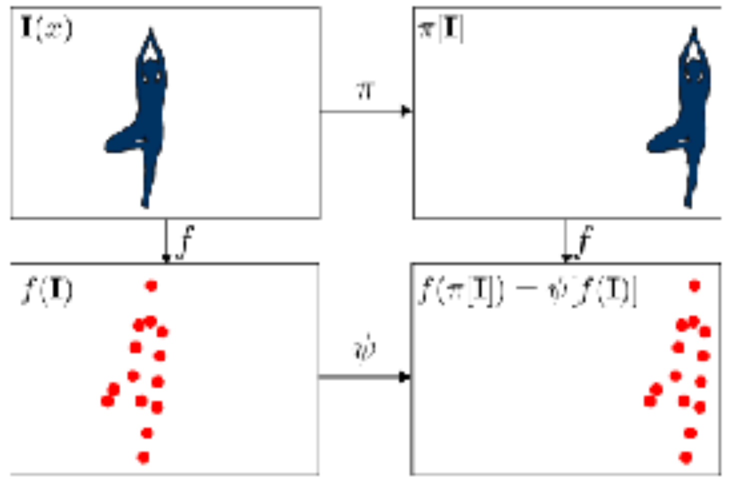


Figure 1. Patch-wise translation equivariance in CNNs arises from translational weight tying, so that a translation π of the input image $\mathbf{I}$, leads to a corresponding translation ψ of the feature maps $f(\mathbf{I})$, where $\pi \neq \psi$ in general, due to pooling effects. However, for rotations, CNNs do not yet have a feature space transformation ψ 'hard-baked' into their structure, and it is complicated to discover what ψ may be, if it exists at all. Harmonic Networks have a hard-baked representation, which allows for easier interpretation of feature maps—see Figure 3.

consider detecting a deformable object, such as a butterfly. The pose of the wings is limited in range, and so there are only certain poses our detector should normally see. A transformation invariant detector, good at detecting wings, would detect them whether they were bigger, further apart, rotated, etc., and it would encode all these cases with the same representation. It would fail to notice nonsense situations, however, such as a butterfly with wings rotated past the usual range, because it has thrown that extra pose information away. An equivariant detector, on the other hand, does not dispose of local pose information, and so it hands on a richer and more useful representation to downstream processes. Equivariance conveys more information about an input to downstream processes, it also constrains the space of possible learned models to those that are valid under the rules of natural image formation [30]. This makes learning more reliable and helps with generalization. For instance, consider CNNs. The key insight is that the statistics of natural images, embodied in the correlations between pixels, are a) invariant to translation, and b) highly localized. Thus features at every layer in a CNN are computed on local receptive fields, where weights are shared

1. Introduction

...representing 360°-rotations ...ks (CNNs) [19]. Currently, ...ed by design to map an image ...ed versions of the image map ...ons of the same feature vector ...Figure 1. However, until now, ...en the feature vectors do not ...ful or easy to predict manner. ...relating input transformations ...is called *equivariance*. ...e is invariance, where feature ...ll transformations of the input. ...globally for a model, such as a ...l not to restrict all intermediate ...mation invariant. For example,

...uk/projects/harmonicNets/

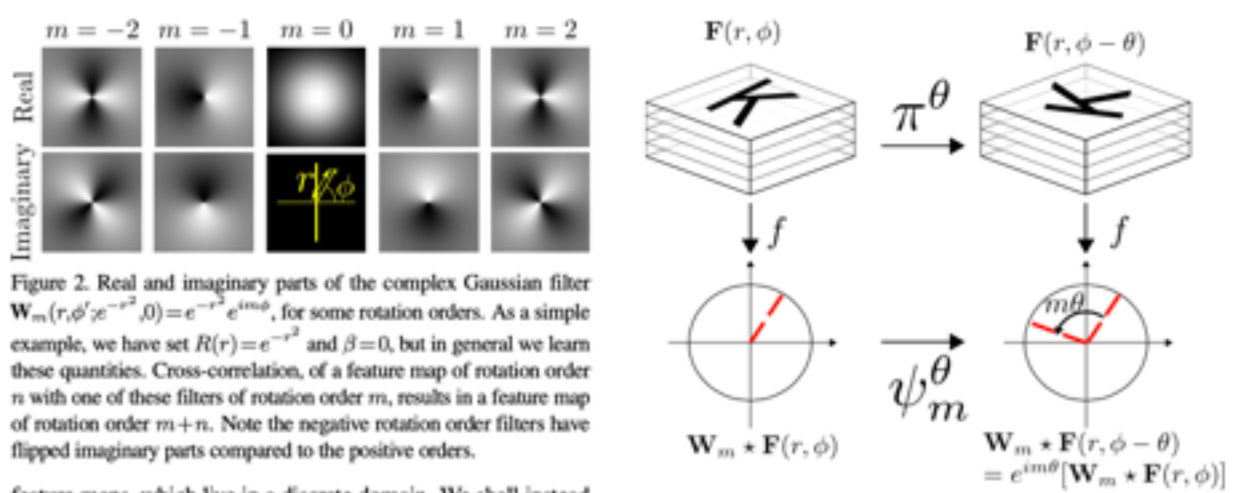


Figure 2. Real and imaginary parts of the complex Gaussian filter $\mathbf{W}_m(r, \phi; e^{-r^2}, 0) = e^{-r^2}e^{im\phi}$, for some rotation orders. As a simple example, we have set $R(r) = e^{-r^2}$ and $\beta = 0$, but in general we learn these quantities. Cross-correlation, of a feature map of rotation order n with one of these filters of rotation order m, results in a feature map of rotation order $m+n$. Note the negative rotation order filters have flipped imaginary parts compared to the positive orders.

feature maps, which live in a discrete domain. We shall instead use continuous spaces, because the analysis is easier. Later on in Section 4.2 we shall demonstrate how to convert back to the discrete domain for practical implementation, but for now we work entirely in continuous Euclidean space.

3.1. Equivariance

Equivariance is a useful property to have because transforma-

Figure 3. DOWN: Cross-correlation of the input patch with $\mathbf{W}_m$, yields a scalar complex-valued response. ACROSS-THEN-DOWN: Cross-correlation with the θ-rotated image yields another complex-valued response. BOTTOM: We transform from the unrotated response to the rotated response, through multiplication by $e^{im\theta}$.

Here r, ϕ are the spatial coordinates of image/feature maps, expressed in polar form, $m \in \mathbb{Z}$ is known as the *rotation order*,

Gauge-equivariant CNNs
(manifolds)

Cohen et al. 2019
Icosahedron

Weiler et al. 2021
Coordinate Independent CNNs

de Haan et al. 2020
Meshes

2018
transl.

. 2018

Bekkers 2019
Lie groups (rotation, scale)

dor-Trivedi 2018
SE(3)

Weiler et al. 2018
SE(3)

Finzi et al. 2020
Lie groups

Chakraborty et al. 2018
Riemannian Hom. spaces

l. 2018

Weiler-Cesa 2019
SE(2)

Sosnovik et al. 2020
Scale-translation

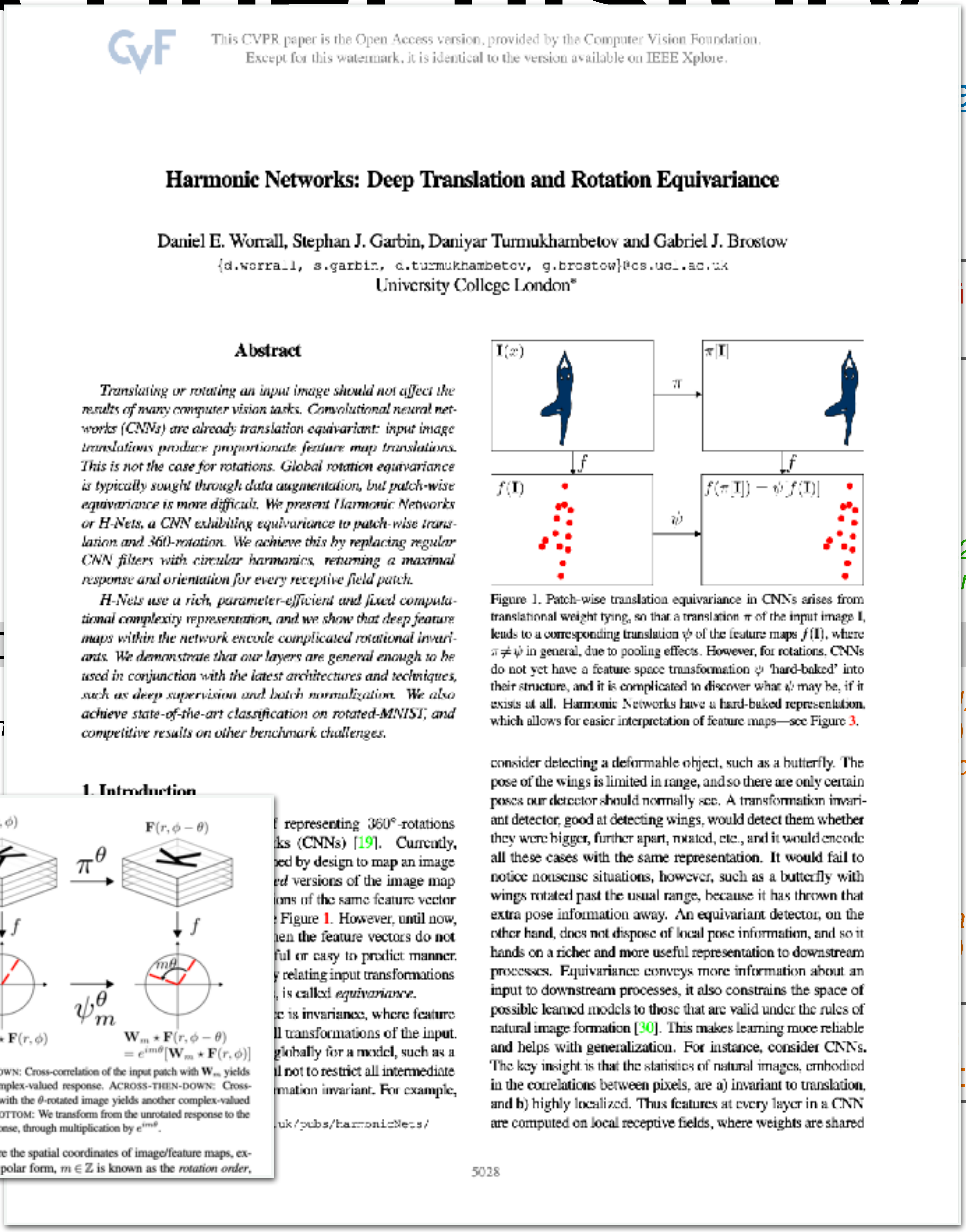
rotation G-CNNs
(steerable)

Continuous G-CNNs
(Lie groups)

Cesa-Lang-Weiler 2022
$G = \mathbb{R}^d \rtimes H$ with H compact

https://quva-lab.github.io/escnn/

A brief history of G-CNNs



This CVPR paper is the Open Access version, provided by the Computer Vision Foundation. Except for this watermark, it is identical to the version available on IEEE Xplore.

Harmonic Networks: Deep Translation and Rotation Equivariance

Daniel E. Worrall, Stephan J. Garbin, Daniyar Turmukhambetov and Gabriel J. Brostow
{d.worrall, s.garbin, d.turmukhambetov, g.brostow}@cs.ucl.ac.uk
University College London*

Abstract

Translating or rotating an input image should not affect the results of many computer vision tasks. Convolutional neural networks (CNNs) are already translation equivariant: input image translations produce proportionate feature map translations. This is not the case for rotations. Global rotation equivariance is typically sought through data augmentation, but patch-wise equivariance is more difficult. We present Harmonic Networks or H-Nets, a CNN exhibiting equivariance to patch-wise translation and 360-rotation. We achieve this by replacing regular CNN filters with circular harmonics, returning a maximal response and orientation for every receptive field patch.

H-Nets use a rich, parameter-efficient and fixed computational complexity representation, and we show that deep feature maps within the network encode complicated rotational invariants. We demonstrate that our layers are general enough to be used in conjunction with the latest architectures and techniques, such as deep supervision and batch normalization. We also achieve state-of-the-art classification on rotated-MNIST, and competitive results on other benchmark challenges.

Figure 1. Patch-wise translation equivariance in CNNs arises from translational weight tying, so that a translation π of the input image I, leads to a corresponding translation ψ of the feature maps $f(\mathbf{I})$, where $\pi \neq \psi$ in general, due to pooling effects. However, for rotations, CNNs do not yet have a feature space transformation ψ 'hard-baked' into their structure, and it is complicated to discover what ψ may be, if it exists at all. Harmonic Networks have a hard-baked representation, which allows for easier interpretation of feature maps—see Figure 3.

consider detecting a deformable object, such as a butterfly. The pose of the wings is limited in range, and so there are only certain poses our detector should normally see. A transformation invariant detector, good at detecting wings, would detect them whether they were bigger, further apart, rotated, etc., and it would encode all these cases with the same representation. It would fail to notice nonsense situations, however, such as a butterfly with wings rotated past the usual range, because it has thrown that extra pose information away. An equivariant detector, on the other hand, does not dispose of local pose information, and so it hands on a richer and more useful representation to downstream processes. Equivariance conveys more information about an input to downstream processes, it also constrains the space of possible learned models to those that are valid under the rules of natural image formation [30]. This makes learning more reliable and helps with generalization. For instance, consider CNNs. The key insight is that the statistics of natural images, embodied in the correlations between pixels, are a) invariant to translation, and b) highly localized. Thus features at every layer in a CNN are computed on local receptive fields, where weights are shared

5028

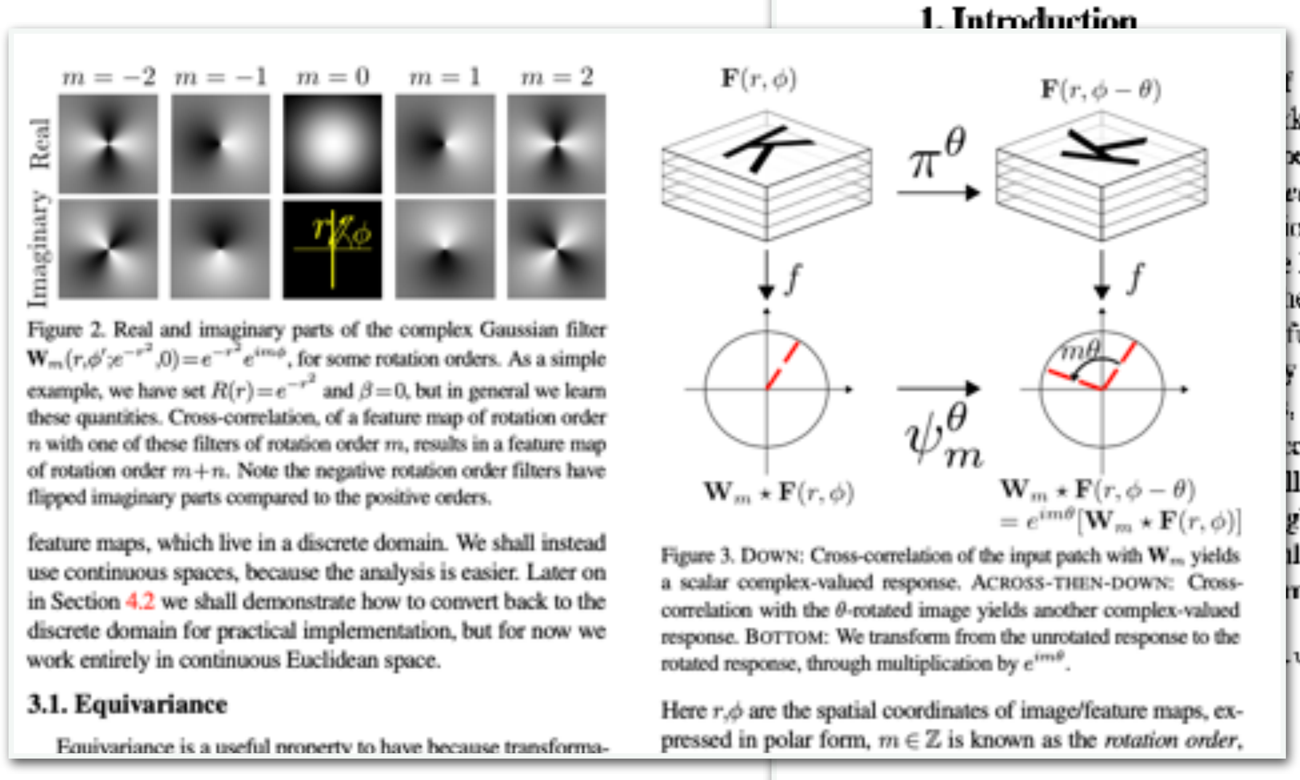


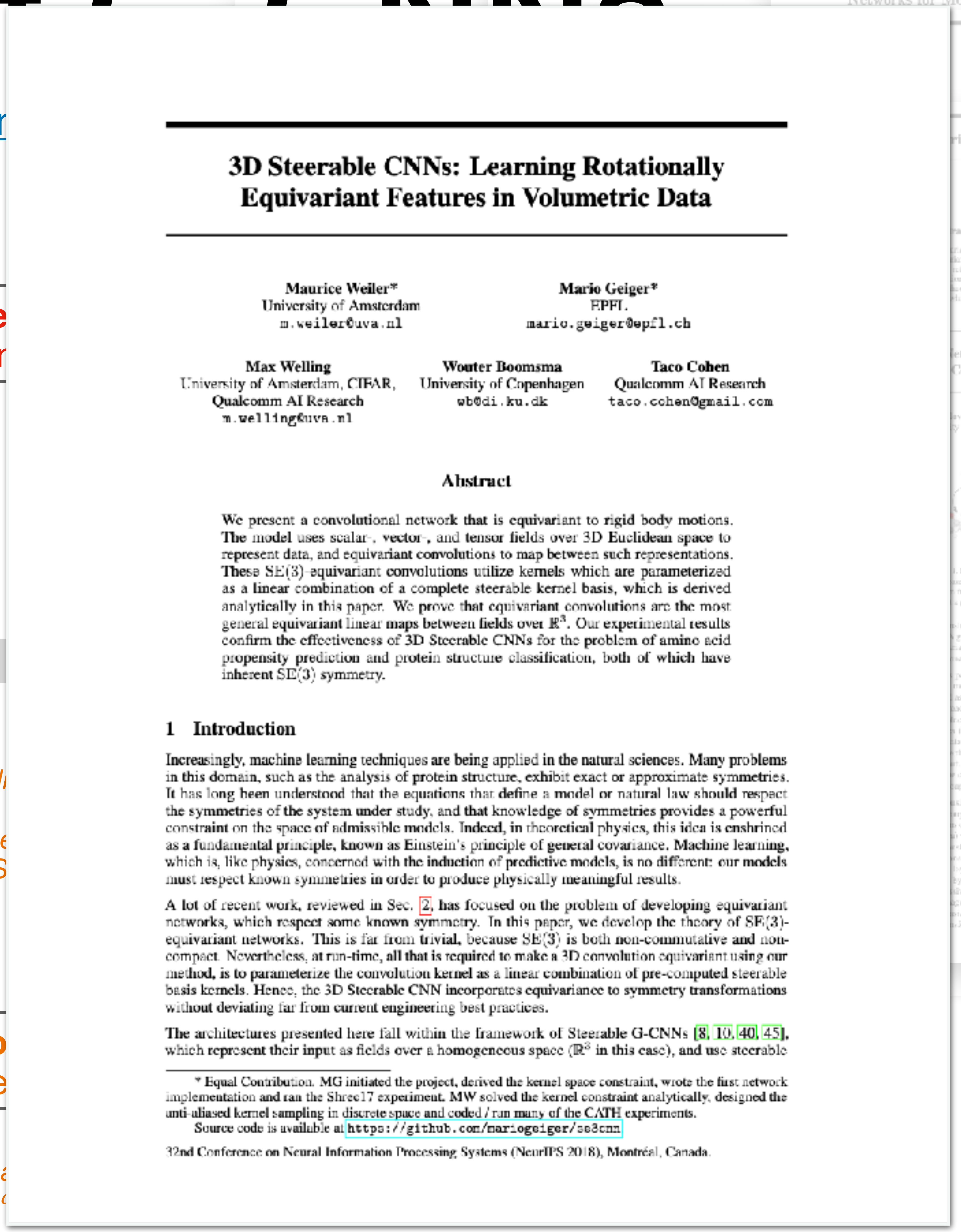
Figure 2. Real and imaginary parts of the complex Gaussian filter $\mathbf{W}_m(r,\phi;e^{-r^2},0) = e^{-r^2}e^{im\phi}$, for some rotation orders. As a simple example, we have set $R(r) = e^{-r^2}$ and $\beta = 0$, but in general we learn these quantities. Cross-correlation, of a feature map of rotation order n with one of these filters of rotation order m, results in a feature map of rotation order $m+n$. Note the negative rotation order filters have flipped imaginary parts compared to the positive orders.

feature maps, which live in a discrete domain. We shall instead use continuous spaces, because the analysis is easier. Later on in Section 4.2 we shall demonstrate how to convert back to the discrete domain for practical implementation, but for now we work entirely in continuous Euclidean space.

3.1. Equivariance

Figure 3. DOWN: Cross-correlation of the input patch with $\mathbf{W}_m$, yields a scalar complex-valued response. ACROSS-THEN-DOWN: Cross-correlation with the θ-rotated image yields another complex-valued response. BOTTOM: We transform from the unrotated response to the rotated response, through multiplication by $e^{im\theta}$.

Here r, ϕ are the spatial coordinates of image/feature maps, expressed in polar form, $m \in \mathbb{Z}$ is known as the *rotation order*,



3D Steerable CNNs: Learning Rotationally Equivariant Features in Volumetric Data

Maurice Weiler* University of Amsterdam m.weiler@uva.nl
Mario Geiger* EPFL mario.geiger@epfl.ch
Max Welling University of Amsterdam, CIFAR, Qualcomm AI Research m.welling@uva.nl
Wouter Boomsma University of Copenhagen wb@di.ku.dk
Taco Cohen Qualcomm AI Research taco.cohen@gmail.com

Abstract

We present a convolutional network that is equivariant to rigid body motions. The model uses scalar-, vector-, and tensor fields over 3D Euclidean space to represent data, and equivariant convolutions to map between such representations. These SE(3)-equivariant convolutions utilize kernels which are parameterized as a linear combination of a complete steerable kernel basis, which is derived analytically in this paper. We prove that equivariant convolutions are the most general equivariant linear maps between fields over $\mathbb{R}^3$. Our experimental results confirm the effectiveness of 3D Steerable CNNs for the problem of amino acid propensity prediction and protein structure classification, both of which have inherent SE(3) symmetry.

1 Introduction

Increasingly, machine learning techniques are being applied in the natural sciences. Many problems in this domain, such as the analysis of protein structure, exhibit exact or approximate symmetries. It has long been understood that the equations that define a model or natural law should respect the symmetries of the system under study, and that knowledge of symmetries provides a powerful constraint on the space of admissible models. Indeed, in theoretical physics, this idea is enshrined as a fundamental principle, known as Einstein's principle of general covariance. Machine learning, which is, like physics, concerned with the induction of predictive models, is no different: our models must respect known symmetries in order to produce physically meaningful results.

A lot of recent work, reviewed in Sec. 2, has focused on the problem of developing equivariant networks, which respect some known symmetry. In this paper, we develop the theory of SE(3)-equivariant networks. This is far from trivial, because SE(3) is both non-commutative and non-compact. Nevertheless, at run-time, all that is required to make a 3D convolution equivariant using our method, is to parameterize the convolution kernel as a linear combination of pre-computed steerable basis kernels. Hence, the 3D Steerable CNN incorporates equivariance to symmetry transformations without deviating far from current engineering best practices.

The architectures presented here fall within the framework of Steerable G-CNNs [8, 10, 40, 45], which represent their input as fields over a homogeneous space ($\mathbb{R}^3$ in this case), and use steerable

* Equal Contribution. MG initiated the project, derived the kernel space constraint, wrote the first network implementation and ran the Shrec17 experiment. MW solved the kernel constraint analytically, designed the anti-aliased kernel sampling in discrete space and coded / ran many of the CATH experiments.
Source code is available at https://github.com/mariogeiger/se3cnn

32nd Conference on Neural Information Processing Systems (NeurIPS 2018), Montréal, Canada.

https://quva-lab.github.io/escnn/

Published as a conference paper at ICLR 2022

A Program to Build E(n)-Equivariant Steerable CNNs

Gabriele Cesa
Qualcomm AI Research*
University of Amsterdam
gcesa@qti.qualcomm.com

Leon Lang
University of Amsterdam
l.lang@uva.nl

Maurice Weiler
University of Amsterdam
m.weiler.ml@gmail.com

Abstract

Equivariance is becoming an increasingly popular design choice to build data efficient neural networks by exploiting prior knowledge about the symmetries of the problem at hand. Euclidean steerable CNNs are one of the most common classes of equivariant networks. While the constraints these architectures need to satisfy are understood, existing approaches are tailored to specific (classes of) groups. No generally applicable method that is *practical* for implementation has been described so far. In this work, we generalize the Wigner-Eckart theorem proposed in Lang & Weiler (2020), which characterizes general G-steerable kernel spaces for compact groups G over their homogeneous spaces, to arbitrary G-spaces. This enables us to directly parameterize filters in terms of a band-limited basis on the whole space rather than on G's orbits, but also to easily implement steerable CNNs equivariant to a large number of groups. To demonstrate its generality, we instantiate our method on a variety of isometry groups acting on the Euclidean space $\mathbb{R}^3$. Our framework allows us to build E(3) and SE(3)-steerable CNNs like previous works, but also CNNs with arbitrary $G \leq O(3)$-steerable kernels. For example, we build 3D CNNs equivariant to the symmetries of platonic solids or choose $G = SO(2)$ when working with 3D data having only azimuthal symmetries. We compare these models on 3D shapes and molecular datasets, observing improved performance by matching the model's symmetries to the ones of the data.

1 Introduction

In machine learning, it is common for learning tasks to present a number of *symmetries*. A symmetry in the data occurs, for example, when some property (e.g., the label) does not change if a set of transformations is applied to the data itself, e.g. translations or rotations of images. Symmetries are algebraically described by *groups*. If prior knowledge about the symmetries of a task is available, it is usually beneficial to encode them in the models used (Shawe-Taylor, 1989; Cohen & Welling, 2016a). The property of such models is referred to as *equivariance* and is obtained by introducing some *equivariance constraints* in the architecture (see Eq. 2). A classical example are convolutional neural networks (CNNs), which achieve translation equivariance by constraining linear layers to be convolution operators. A wider class of equivariant models are Euclidean steerable CNNs (Cohen & Welling, 2016b; Weiler et al., 2018a; Weiler & Cesa, 2019; Jenner & Weiler, 2022), which guarantee equivariance to isometries $\mathbb{R}^n \rtimes G$ of a Euclidean space $\mathbb{R}^n$, i.e., to translations and a group G of origin-preserving transformations, such as rotations and reflections. As proven in Weiler et al. (2018a; 2021); Jenner & Weiler (2022), this requires convolutions with G-*steerable* (equivariant) kernels.

Our goal is developing a program to parameterize with minimal requirements arbitrary G-steerable kernel spaces, with compact G, which are required to implement $\mathbb{R}^n \rtimes G$ equivariant CNNs. Lang & Weiler (2020) provides a first step in this direction by generalizing the *Wigner-Eckart theorem* from quantum mechanics to obtain a general technique to parametrize G-steerable kernel spaces over *orbits* of a compact G. The theorem reduces the task of building steerable kernel bases to that of finding some pure representation theoretic ingredients. Since the equivariance constraint only relates points $g.x \in \mathbb{R}^n$ in the same *orbit* $G.x \subset \mathbb{R}^n$, a kernel can take independent values on different orbits. Fig. 1 shows

*Qualcomm AI Research is an initiative of Qualcomm Technologies, Inc.

README.md

General E(2)-Equivariant Steerable CNNs

Documentation | Experiments | Paper | Thesis

e2cnn is a PyTorch extension for equivariant deep learning.

Equivariant neural networks guarantee a specified transformation behavior of their feature spaces under transformations of their input. For instance, classical convolutional neural networks (*CNN*s) are by design equivariant to translations of their input. This means that a translation of an image leads to a corresponding translation of the network's feature maps. This package provides implementations of neural network modules which are equivariant under all *isometries* E(2) of the image plane $\mathbb{R}^2$, that is, under *translations*, *rotations* and *reflections*. In contrast to conventional CNNs, E(2)-equivariant models are guaranteed to generalize over such transformations, and are therefore more data efficient.

The f[...] chara[...] gray-[...]

Getting Started

e2cnn is easy to use since it provides a high level user interface which abstracts most intricacies of group and representation theory away. The following code snippet shows how to perform an equivariant convolution from an RGB-image to 10 *regular* feature fields (corresponding to a group convolution).

```
from e2cnn import gspaces                                          #  1
from e2cnn import nn                                               #  2
import torch                                                       #  3
                                                                   #  4
r2_act = gspaces.Rot2dOnR2(N=8)                                    #  5
feat_type_in  = nn.FieldType(r2_act,  3*[r2_act.trivial_repr])     #  6
feat_type_out = nn.FieldType(r2_act, 10*[r2_act.regular_repr])     #  7
                                                                   #  8
conv = nn.R2Conv(feat_type_in, feat_type_out, kernel_size=5)       #  9
relu = nn.ReLU(feat_type_out)                                      # 10
                                                                   # 11
x = torch.randn(16, 3, 32, 32)                                     # 12
x = nn.GeometricTensor(x, feat_type_in)                            # 13
                                                                   # 14
y = relu(conv(x))                                                  # 15
```

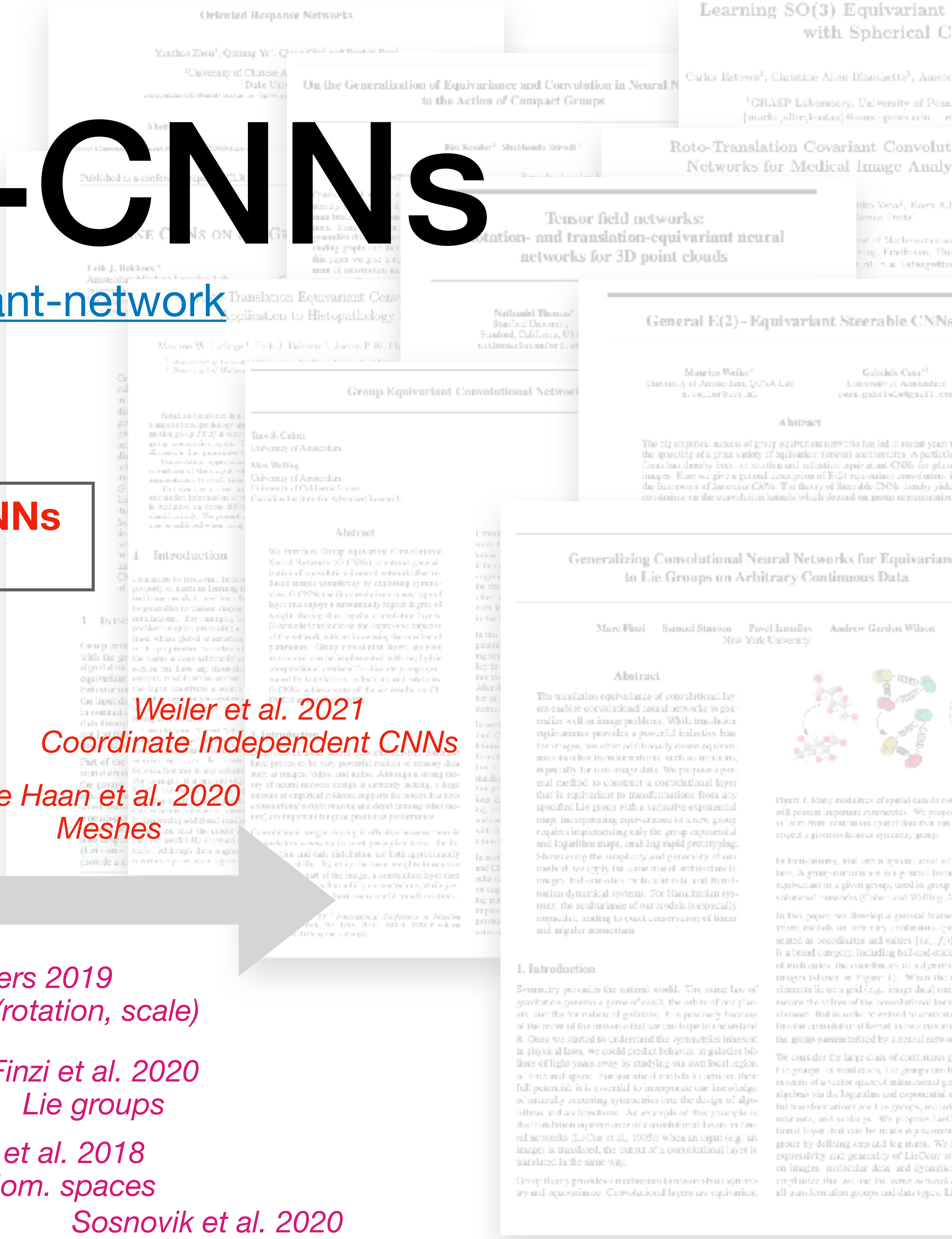
Line 5 specifies the symmetry group action on the image plane $\mathbb{R}^2$ under which the network should be equivariant. We choose the *cyclic group* C_8, which describes discrete rotations by multiples of $2\pi/8$. Line 6 specifies the input feature field types. The three color channels of an RGB image are thereby to be identified as three independent scalar fields, which transform under the *trivial representation* of C_8. Similarly, the output feature space is in line 7

n-C
018
ansl.
om et
E(2,6
Wo
3D
G-C
2017
es 2
SO(3)
Kondor 2018
SE(3)
Thoma
Continu

https://quva-lab.github.io/escnn/

A brief history of G-CNNs

https://github.com/Chen-Cai-OSU/awesome-equivariant-network



Discrete G-CNNs
(square/cube symmetries)

Gauge-equivariant CNNs
(manifolds)

Cohen-Welling 2016
p4m

Winkels-Cohen 2018
3D discrete roto-transl.

Hoogeboom et al. 2018
SE(2,6)

Dieleman et al. 2016
p4m

Worrall-Brostow 2018
3D discrete roto-transl.

Cohen et al. 2019
Icosahedron

Weiler et al. 2021
Coordinate Independent CNNs

de Haan et al. 2020
Meshes

NNs → CNNs → G-CNNs

LeCun et al. 1990

Mallat et al. 2013, 2015
SE(2)

Bekkers et al. 2014-2018
SE(2)

Weiler et al. 2017
SE(2)

Zhou et al. 2017
SE(2)

Worrall et al. 2017
SE(2)

Esteves 2017
SO(3)

Kondor 2018
SE(3)

Cohen et al. 2018
SE(3)

Kondor-Trivedi 2018
SE(3)

Weiler et al. 2018
SE(3)

Thomas et al. 2018
SE(3)

Weiler-Cesa 2019
SE(2)

Bekkers 2019
Lie groups (rotation, scale)

Finzi et al. 2020
Lie groups

Chakraborty et al. 2018
Riemannian Hom. spaces

Sosnovik et al. 2020
Scale-translation

Continuous rotation G-CNNs
(regular discretizations)

Continuous rotation G-CNNs
(steerable)

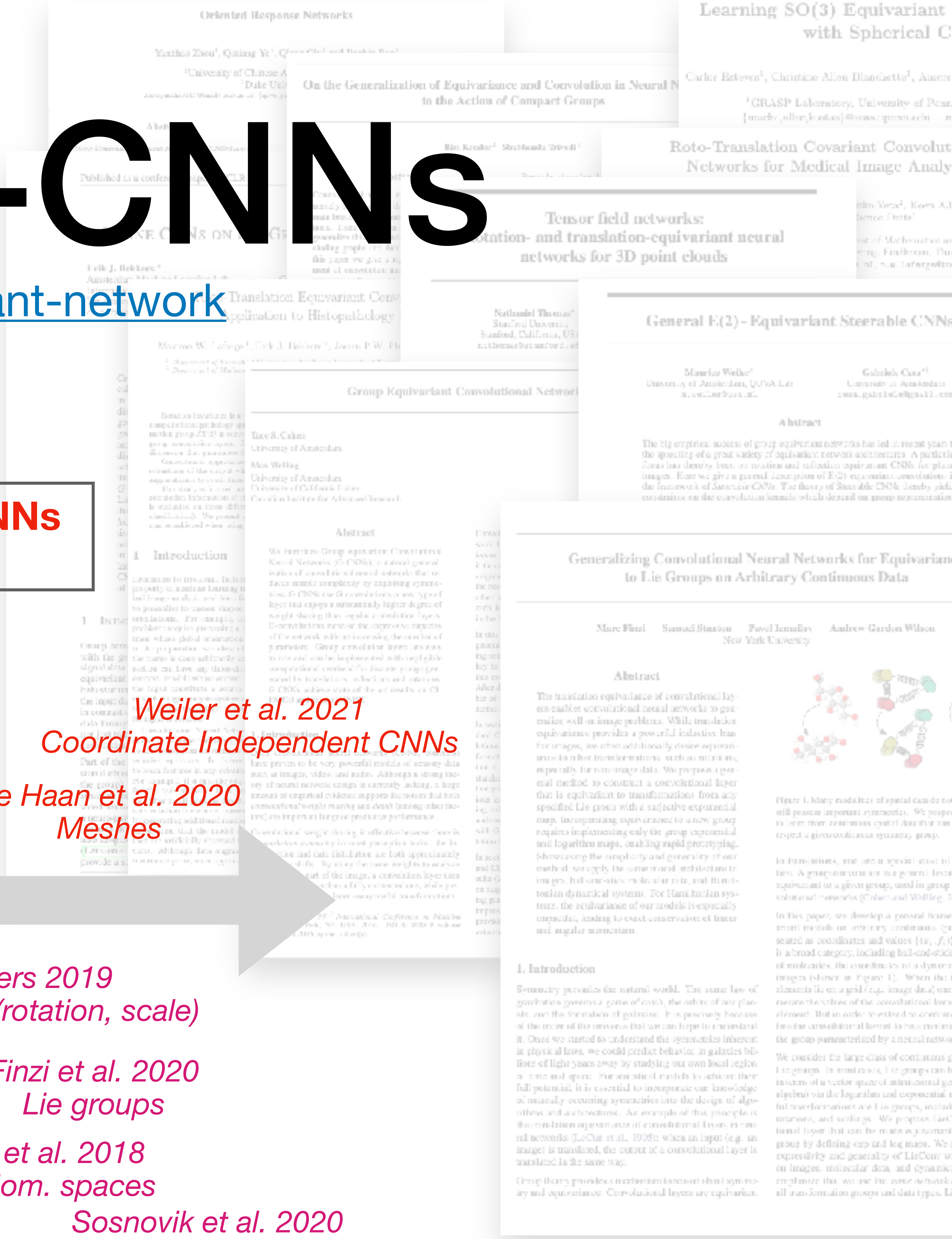
Continuous G-CNNs
(Lie groups)

Cesa-Lang-Weiler 2022
$G = \mathbb{R}^d \rtimes H$ *with* H *compact*

https://quva-lab.github.io/escnn/

A brief history of G-CNNs

https://github.com/Chen-Cai-OSU/awesome-equivariant-network



Discrete G-CNNs
(square/cube symmetries)

Gauge-equivariant CNNs
(manifolds)

Cohen-Welling 2016
p4m

Winkels-Cohen 2018
3D discrete roto-transl.

Hoogeboom et al. 2018
SE(2,6)

Dieleman et al. 2016
p4m

Worrall-Brostow 2018
3D discrete roto-transl.

Cohen et al. 2019
Icosahedron

Weiler et al. 2021
Coordinate Independent CNNs

de Haan et al. 2020
Meshes

NNs → CNNs → G-CNNs

LeCun et al. 1990

Mallat et al. 2013, 2015
SE(2)

Bekkers et al. 2014-2018
SE(2)

Weiler et al. 2017
SE(2)

Zhou et al. 2017
SE(2)

Worrall et al. 2017
SE(2)

Esteves 2017
SO(3)

Kondor 2018
SE(3)

Cohen et al. 2018
SE(3)

Kondor-Trivedi 2018
SE(3)

Weiler et al. 2018
SE(3)

Thomas et al. 2018
SE(3)

Weiler-Cesa 2019
SE(2)

Bekkers 2019
Lie groups (rotation, scale)

Finzi et al. 2020
Lie groups

Chakraborty et al. 2018
Riemannian Hom. spaces

Sosnovik et al. 2020
Scale-translation

Continuous rotation G-CNNs
(regular discretizations)

Continuous rotation G-CNNs
(steerable)

Continuous G-CNNs
(Lie groups)

Cesa-Lang-Weiler 2022
$G = \mathbb{R}^d \rtimes H$ *with* H *compact*

https://quva-lab.github.io/escnn/

A brief history of G-CNNs

com/Chen-Cai-OSU/awesome-equivariant-network

Published as a conference paper at ICLR 2020

B-Spline CNNs on Lie Groups

Erik J. Bekkers
Centre for Analysis and Scientific Computing
Department of Applied Mathematics and Computer Science
Eindhoven University of Technology, Eindhoven, the Netherlands
e.j.bekkers@tue.nl

Abstract

Group convolutional neural networks (G-CNNs) can be used to improve classical CNNs by equipping them with the geometric structure of groups. Central in the success of G-CNNs is the lifting of feature maps to higher dimensional disentangled representations in which data characteristics are effectively learned, geometric data-augmentations are made obsolete, and predictable behavior under geometric transformations (equivariance) is guaranteed via group theory. Currently, however, the practical implementations of G-CNNs are limited to either discrete groups (that leave the grid intact) or continuous compact groups such as rotations (that enable the use of Fourier theory). In this paper we lift these limitations and propose a modular framework for the design and implementation of *G-CNNs for arbitrary Lie groups*. In our approach the differential structure of Lie groups is used to expand convolution kernels in a generic basis of B-splines that is defined on the Lie algebra. This leads to a flexible framework that enables *localized*, *atrous*, and *deformable convolutions* in G-CNNs by means of respectively *localized*, *sparse* and *non-uniform B-spline* expansions. The impact and potential of our approach is studied on two benchmark datasets: cancer detection in histopathology slides in which rotation equivariance plays a key role and facial landmark localization in which scale equivariance is important. In both cases, G-CNN architectures outperform their classical 2D counterparts and the added value of atrous and localized group convolutions is studied in detail.

1 Introduction

Group convolutional neural networks (G-CNNs) are a class of neural networks that are equipped with the geometry of groups. This enables them to profit from the structure and symmetries in signal data such as images (Cohen & Welling, 2016). A key feature of G-CNNs is that they are equivariant with respect to transformations described by the group, i.e., they guarantee predictable behavior under such transformations and are insensitive to both local and global transformations on the input data. Classical CNNs are a special case of G-CNNs that are equivariant to translations and, in contrast to unconstrained NNs, they make advantage of (and preserve) the basic structure of signal data throughout the network (LeCun et al., 1990). By considering larger groups (i.e. considering not just translation equivariance) additional geometric structure can be utilized in order to improve performance and data efficiency (see G-CNN literature in Sec. 2).

Part of the success of G-CNNs can be attributed to the lifting of feature maps to higher dimensional objects that are generated by matching kernels under a range of poses (transformations in the group). This leads to a disentanglement with respect to the pose and together with the group structure this enables a flexible way of learning high level representations in terms of low-level activated neurons observed in specific configurations, which we conceptually illustrate in Fig. 1. From a neuro-psychological viewpoint, this resembles a hierarchical composition from low- to high-level features akin to the recognition-by-components model by Biederman (1987), a viewpoint which is also adopted in work on capsule networks (Hinton et al., 2011; Sabour et al., 2017). In particular in (Lenssen et al., 2018) the group theoretical connection is made explicit with equivariant capsules that provide a sparse index/value representation of feature maps on groups (Gens & Domingos, 2014).

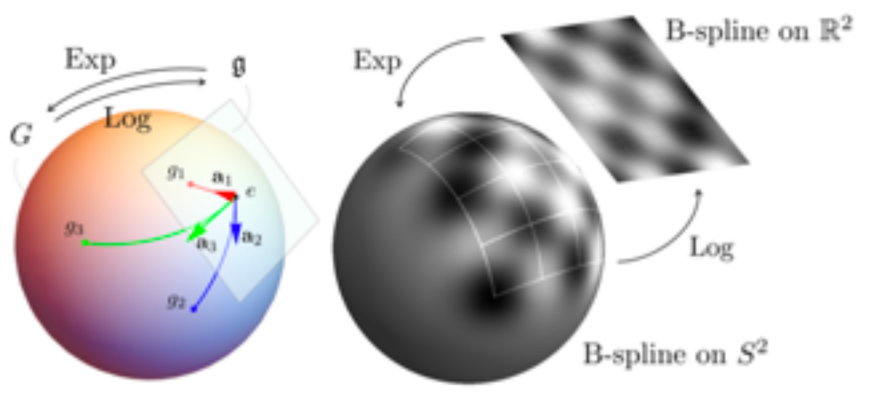


Figure 2: The Log-map allows us to map elements from curved manifolds such as the 2-sphere to a flat Euclidean tangent space. For Lie groups the Log-map is analytic, globally defined, and it provides us with a flexible tool to define group convolution kernels via B-splines. In our Lie group context the 2-sphere is treated as the quotient $SO(3)/SO(2)$. Technical details are given in Sec. 3 and App. B.

ries)

Gauge-equivariant CNNs (manifolds)

els-Cohen 2018
crete roto-transl.

Hoogeboom et al. 2018
SE(2,6)

Worrall-Brostow 2018
3D discrete roto-transl.

Cohen et al. 2019
Icosahedron

Weiler et al. 2021
Coordinate Independent CNNs

de Haan et al. 2020
Meshes

G-CNNs

Vorrall et al. 2017
SE(2)

Cohen et al. 2018
SE(3)

Esteves 2017
SO(3)

Kondor-Trivedi 2018
SE(3)

Kondor 2018
SE(3)

Weiler et al. 2018
SE(3)

et al. 2017
SE(2)

Thomas et al. 2018
SE(3)

Bekkers 2019
Lie groups (rotation, scale)

Finzi et al. 2020
Lie groups

Chakraborty et al. 2018
Riemannian Hom. spaces

Weiler-Cesa 2019
SE(2)

Sosnovik et al. 2020
Scale-translation

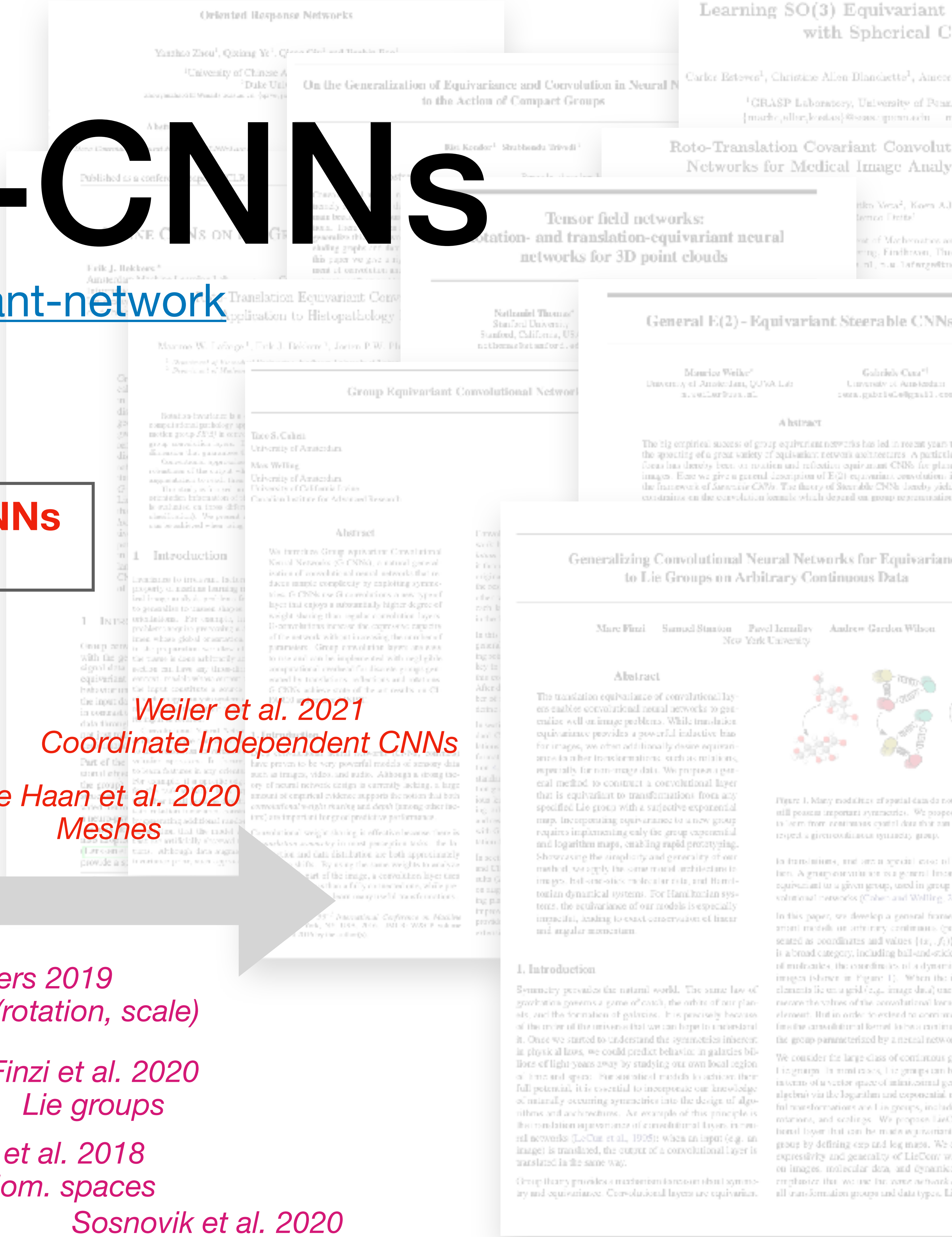
Continuous G-CNNs (Lie groups)

2022
compact

ub.io/escnn/

A brief history of G-CNNs

https://github.com/Chen-Cai-OSU/awesome-equivariant-network



NNs → CNNs → G-CNNs

LeCun et al. 1990

Discrete G-CNNs
(square/cube symmetries)

Cohen-Welling 2016
p4m

Winkels-Cohen 2018
3D discrete roto-transl.

Hoogeboom et al. 2018
SE(2,6)

Dieleman et al. 2016
p4m

Worrall-Brostow 2018
3D discrete roto-transl.

Gauge-equivariant CNNs
(manifolds)

Cohen et al. 2019
Icosahedron

Weiler et al. 2021
Coordinate Independent CNNs

de Haan et al. 2020
Meshes

Mallat et al. 2013, 2015
SE(2)

Bekkers et al. 2014-2018
SE(2)

Weiler et al. 2017
SE(2)

Zhou et al. 2017
SE(2)

Worrall et al. 2017
SE(2)

Esteves 2017
SO(3)

Kondor 2018
SE(3)

Cohen et al. 2018
SE(3)

Kondor-Trivedi 2018
SE(3)

Weiler et al. 2018
SE(3)

Thomas et al. 2018
SE(3)

Weiler-Cesa 2019
SE(2)

Bekkers 2019
Lie groups (rotation, scale)

Finzi et al. 2020
Lie groups

Chakraborty et al. 2018
Riemannian Hom. spaces

Sosnovik et al. 2020
Scale-translation

Continuous rotation G-CNNs
(regular discretizations)

Continuous rotation G-CNNs
(steerable)

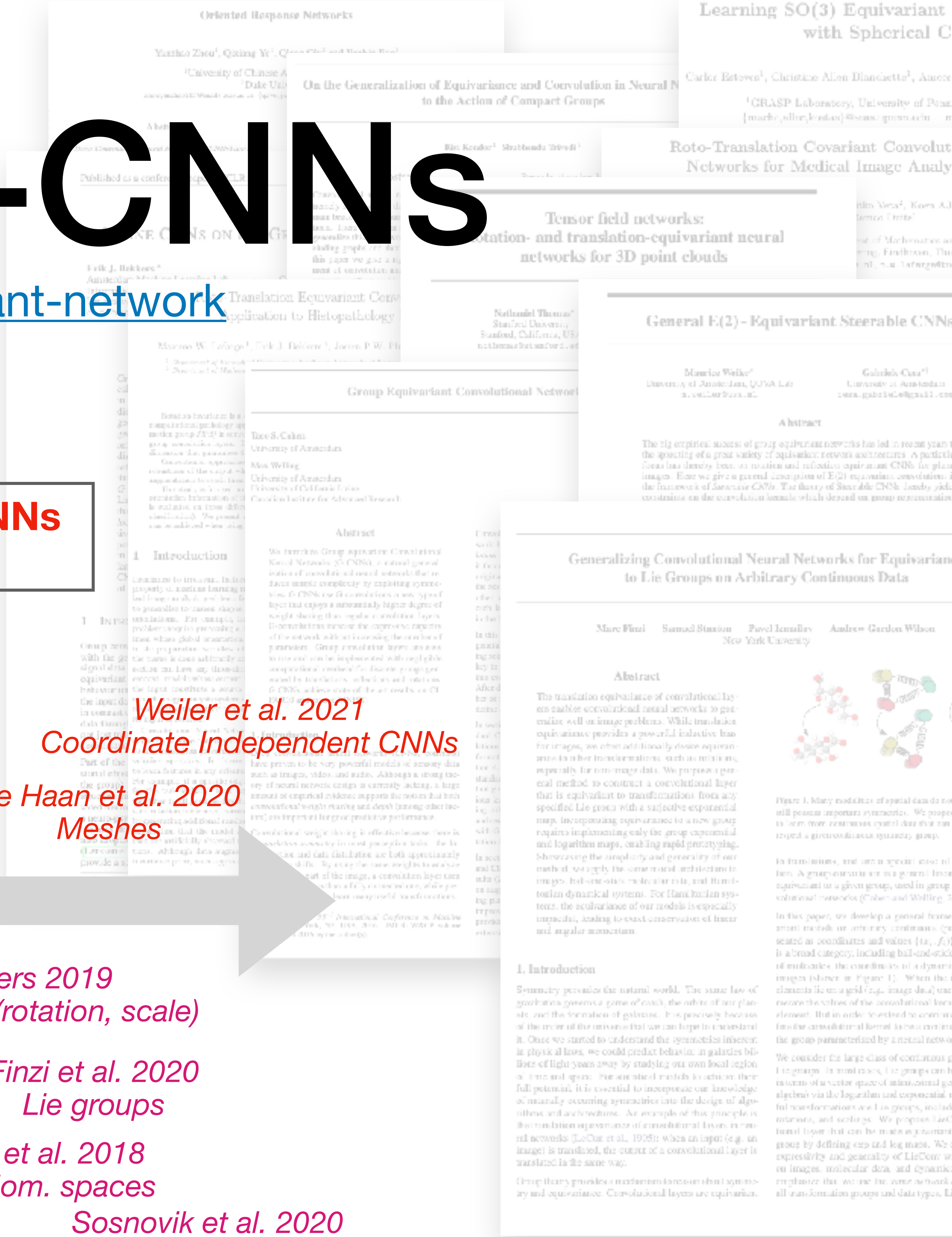
Continuous G-CNNs
(Lie groups)

Cesa-Lang-Weiler 2022
$G = \mathbb{R}^d \rtimes H$ *with* H *compact*

https://quva-lab.github.io/escnn/

A brief history of G-CNNs

https://github.com/Chen-Cai-OSU/awesome-equivariant-network



Discrete G-CNNs
(square/cube symmetries)

Gauge-equivariant CNNs
(manifolds)

Cohen-Welling 2016
p4m

Winkels-Cohen 2018
3D discrete roto-transl.

Hoogeboom et al. 2018
SE(2,6)

Dieleman et al. 2016
p4m

Worrall-Brostow 2018
3D discrete roto-transl.

Cohen et al. 2019
Icosahedron

Weiler et al. 2021
Coordinate Independent CNNs

de Haan et al. 2020
Meshes

NNs → CNNs → G-CNNs

LeCun et al. 1990

Mallat et al. 2013, 2015
SE(2)

Bekkers et al. 2014-2018
SE(2)

Weiler et al. 2017
SE(2)

Zhou et al. 2017
SE(2)

Worrall et al. 2017
SE(2)

Esteves 2017
SO(3)

Kondor 2018
SE(3)

Cohen et al. 2018
SE(3)

Kondor-Trivedi 2018
SE(3)

Weiler et al. 2018
SE(3)

Thomas et al. 2018
SE(3)

Weiler-Cesa 2019
SE(2)

Bekkers 2019
Lie groups (rotation, scale)

Finzi et al. 2020
Lie groups

Chakraborty et al. 2018
Riemannian Hom. spaces

Sosnovik et al. 2020
Scale-translation

Continuous rotation G-CNNs
(regular discretizations)

Continuous rotation G-CNNs
(steerable)

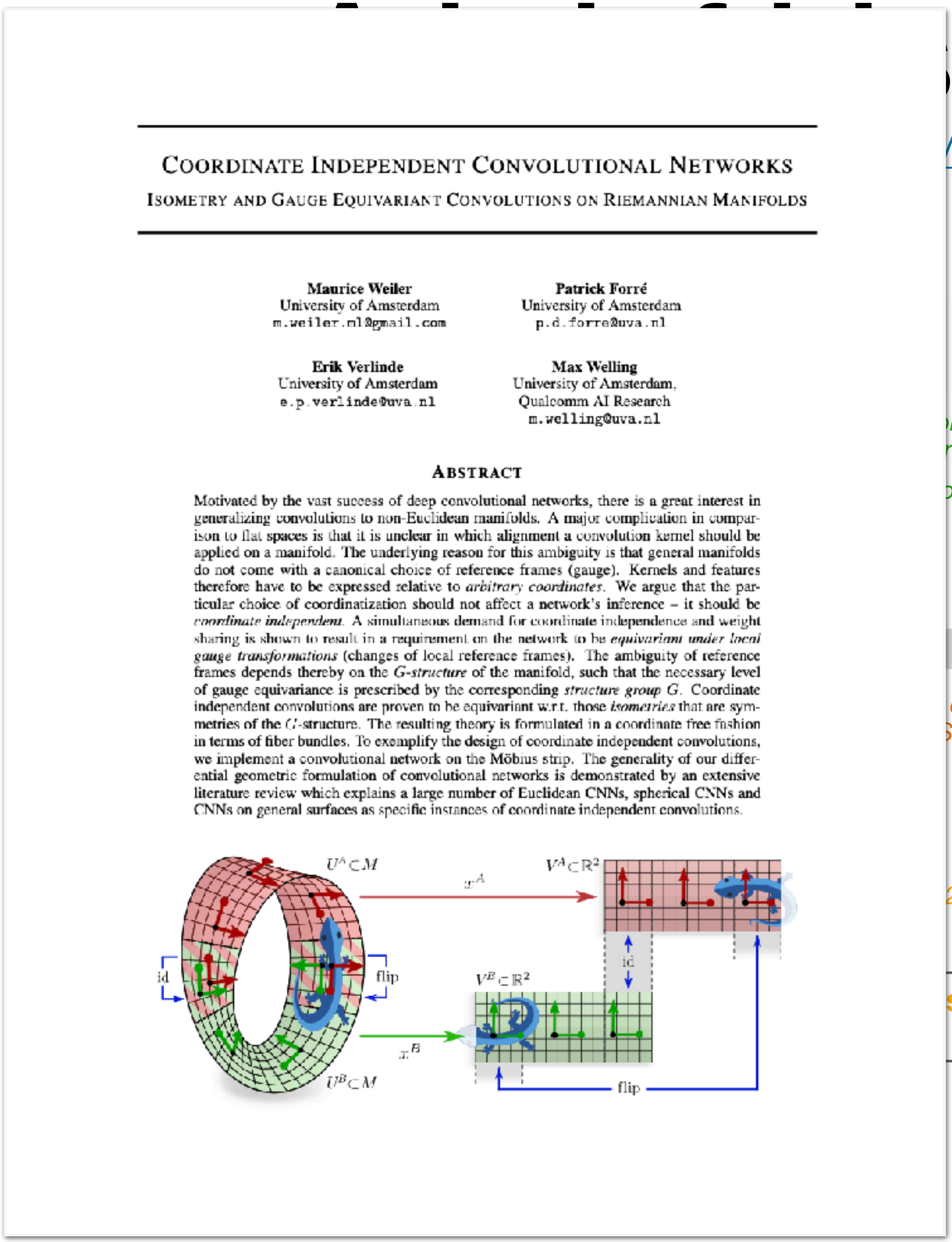
Continuous G-CNNs
(Lie groups)

Cesa-Lang-Weiler 2022
$G = \mathbb{R}^d \rtimes H$ *with* H *compact*

https://quva-lab.github.io/escnn/

...story ... G-CNNs

/Chen-Cai-OSU/a



COORDINATE INDEPENDENT CONVOLUTIONAL NETWORKS

ISOMETRY AND GAUGE EQUIVARIANT CONVOLUTIONS ON RIEMANNIAN MANIFOLDS

Maurice Weiler
University of Amsterdam
m.weiler.ml@gmail.com

Patrick Forré
University of Amsterdam
p.d.forre@uva.nl

Erik Verlinde
University of Amsterdam
e.p.verlinde@uva.nl

Max Welling
University of Amsterdam,
Qualcomm AI Research
m.welling@uva.nl

ABSTRACT

Motivated by the vast success of deep convolutional networks, there is a great interest in generalizing convolutions to non-Euclidean manifolds. A major complication in comparison to flat spaces is that it is unclear in which alignment a convolution kernel should be applied on a manifold. The underlying reason for this ambiguity is that general manifolds do not come with a canonical choice of reference frames (gauge). Kernels and features therefore have to be expressed relative to *arbitrary coordinates*. We argue that the particular choice of coordinatization should not affect a network's inference – it should be *coordinate independent*. A simultaneous demand for coordinate independence and weight sharing is shown to result in a requirement on the network to be *equivariant under local gauge transformations* (changes of local reference frames). The ambiguity of reference frames depends thereby on the *G-structure* of the manifold, such that the necessary level of gauge equivariance is prescribed by the corresponding *structure group* G. Coordinate independent convolutions are proven to be equivariant w.r.t. those *isometries* that are symmetries of the G-structure. The resulting theory is formulated in a coordinate free fashion in terms of fiber bundles. To exemplify the design of coordinate independent convolutions, we implement a convolutional network on the Möbius strip. The generality of our differential geometric formulation of convolutional networks is demonstrated by an extensive literature review which explains a large number of Euclidean CNNs, spherical CNNs and CNNs on general surfaces as specific instances of coordinate independent convolutions.

G

...hen 2018
roto-transl.

...ogeboom et al. 2018
SE(2,6)

Worrall-Brostow ...
3D discrete roto-t...

G-CNNs

... et al. 2017
SE(2)

Cohen et al...
SE(3)
Kon...

Esteves 2017
SO(3)

Kondor 2018
SE(3)

...2017

Thomas et a...
SE(3)

...s

Continuous
(st...

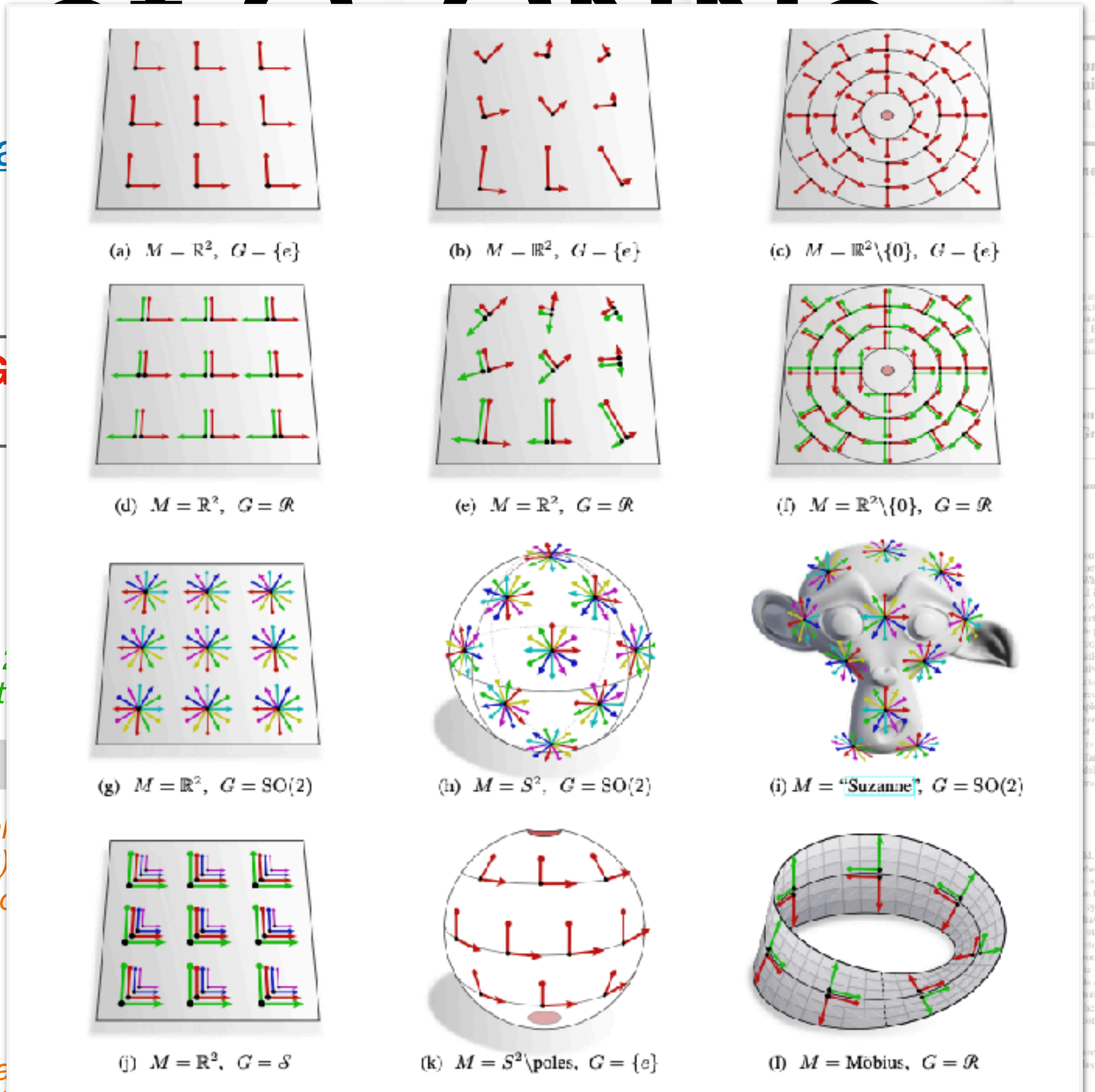
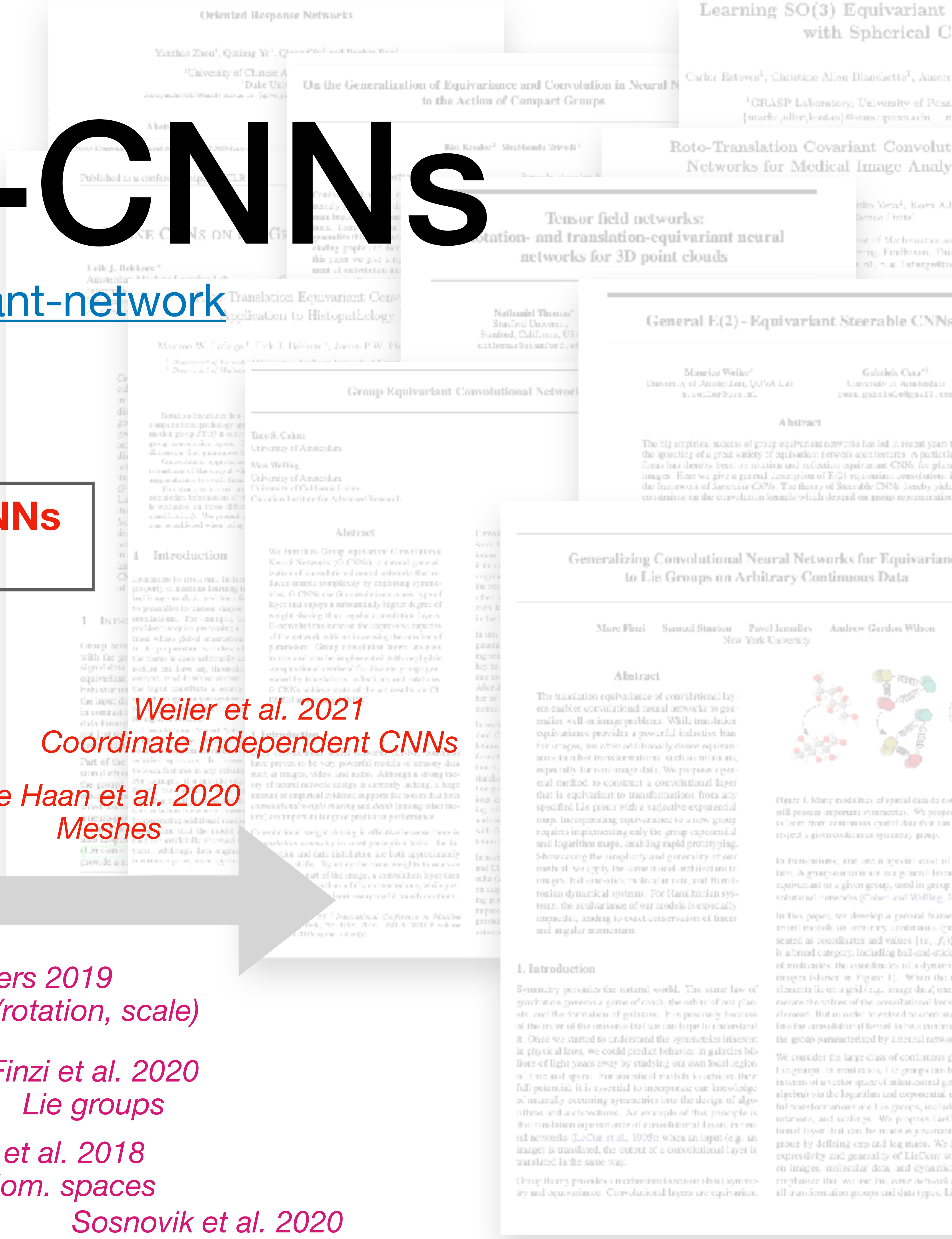


Figure 5: Exemplary G-structures GM for different structure groups G and on different manifolds M. The structure group G signals which values gauge transformations can take, and therefore how "big" the subset of distinguished frames at each point p is. Fig. 5a shows the canonical $\{e\}$-structure (frame field) of $\mathbb{R}^2$, which corresponds to conventional Euclidean CNNs. The G-structures in Figs. 5d, 5g and 5j are constructed by adding reflected ($G = \mathcal{R}$), rotated ($G = \mathrm{SO}(2)$) and scaled ($G = \mathcal{S}$) frames, respectively. The corresponding GM-convolutions are not only translation equivariant but equivariant under the action of affine groups $\mathrm{Aff}(G)$. G structures are usually not unique. Figs. 5b and 5e show alternative G structures on $\mathbb{R}^2$ (corresponding to an alternative metric relative to which their frames are orthonormal). They might not be practically relevant but demonstrate the flexibility of our framework. The $\{e\}$-structure in Fig. 5c corresponds to polar coordinates. As G-structures are required to be continuous, we removed the origin 0 where polar coordinates are singular. One can once again define an $\mathcal{R}$-structure by adding reflected frames as shown in Fig. 5f. These G structures model convolutions on $\mathbb{R}^2\backslash\{0\}$ which are $\mathrm{SO}(2)$ and $\mathrm{O}(2)$ equivariant but not translation equivariant. Fig. 5h shows the usual $\mathrm{SO}(2)$-structure on the embedded 2-sphere S^2, which is underlying $\mathrm{SO}(3)$-equivariant spherical CNNs. Another popular choice is the $\{e\}$-structure in Fig. 5k, which is induced by spherical coordinates. Note that this $\{e\}$-structure would be singular at the poles, which are therefore cut out. Continuous (i.e. non-singular) reductions of the structure group beyond $\mathrm{SO}(2)$ are on the sphere topologically obstructed. G steerable kernels with $G \geq \mathrm{SO}(2)$ are therefore strictly necessary for continuous convolutions on topological spheres like the mesh in Fig. 5i. Fig. 5l shows an $\mathcal{R}$-structure on the Möbius strip. As the Möbius strip is non-orientable, it does not admit a continuous reduction of the structure group beyond the reflection group $G = \mathcal{R}$.

https://qva-lab.github.io/escnn/

A brief history of G-CNNs

https://github.com/Chen-Cai-OSU/awesome-equivariant-network



NNs → CNNs → G-CNNs

LeCun et al. 1990

Discrete G-CNNs
(square/cube symmetries)

Cohen-Welling 2016
p4m

Winkels-Cohen 2018
3D discrete roto-transl.

Hoogeboom et al. 2018
SE(2,6)

Dieleman et al. 2016
p4m

Worrall-Brostow 2018
3D discrete roto-transl.

Gauge-equivariant CNNs
(manifolds)

Cohen et al. 2019
Icosahedron

Weiler et al. 2021
Coordinate Independent CNNs

de Haan et al. 2020
Meshes

Mallat et al. 2013, 2015
SE(2)

Bekkers et al. 2014-2018
SE(2)

Weiler et al. 2017
SE(2)

Zhou et al. 2017
SE(2)

Worrall et al. 2017
SE(2)

Esteves 2017
SO(3)

Kondor 2018
SE(3)

Thomas et al. 2018
SE(3)

Cohen et al. 2018
SE(3)

Kondor-Trivedi 2018
SE(3)

Weiler et al. 2018
SE(3)

Weiler-Cesa 2019
SE(2)

Bekkers 2019
Lie groups (rotation, scale)

Finzi et al. 2020
Lie groups

Chakraborty et al. 2018
Riemannian Hom. spaces

Sosnovik et al. 2020
Scale-translation

Continuous rotation G-CNNs
(regular discretizations)

Continuous rotation G-CNNs
(steerable)

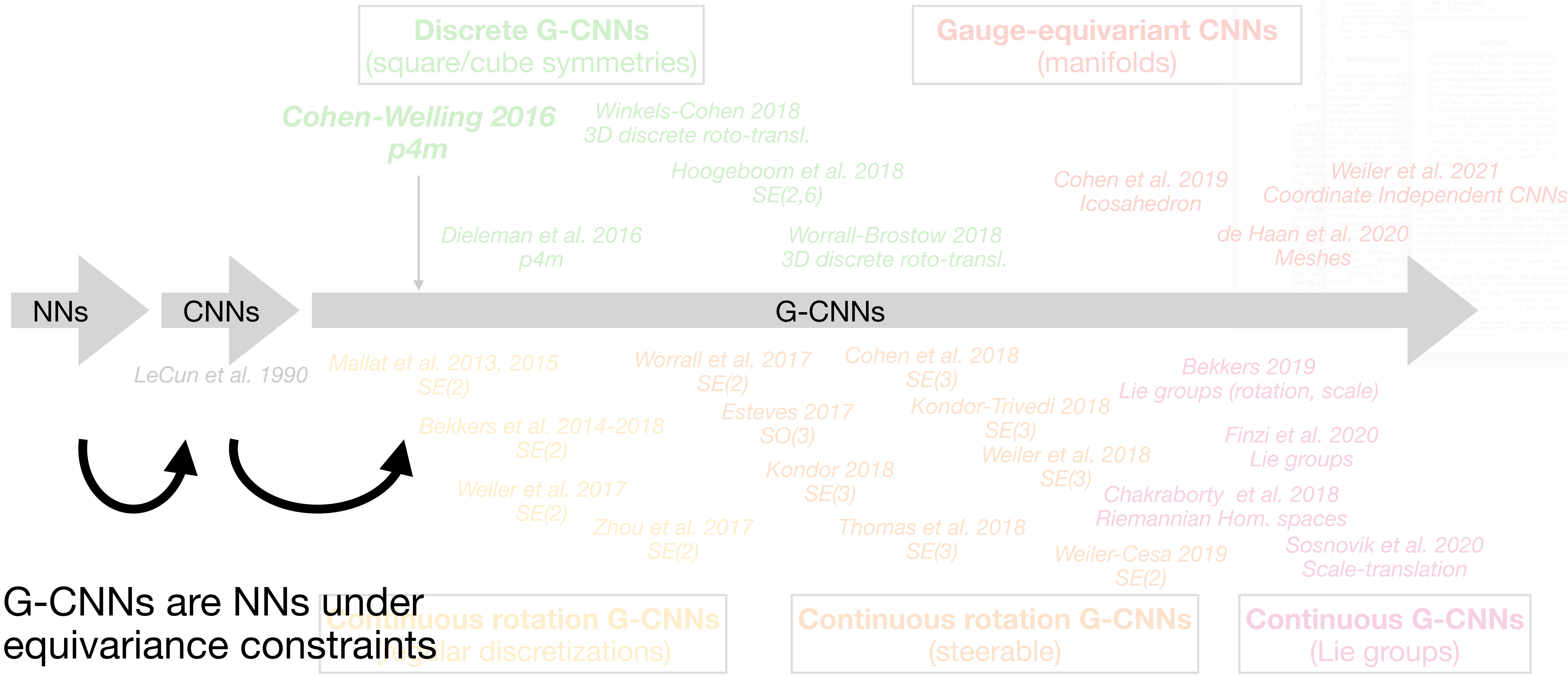
Continuous G-CNNs
(Lie groups)

Cesa-Lang-Weiler 2022
$G = \mathbb{R}^d \rtimes H$ *with* H *compact*

https://quva-lab.github.io/escnn/

A brief history of G-CNNs

https://github.com/Chen-Cai-OSU/awesome-equivariant-network



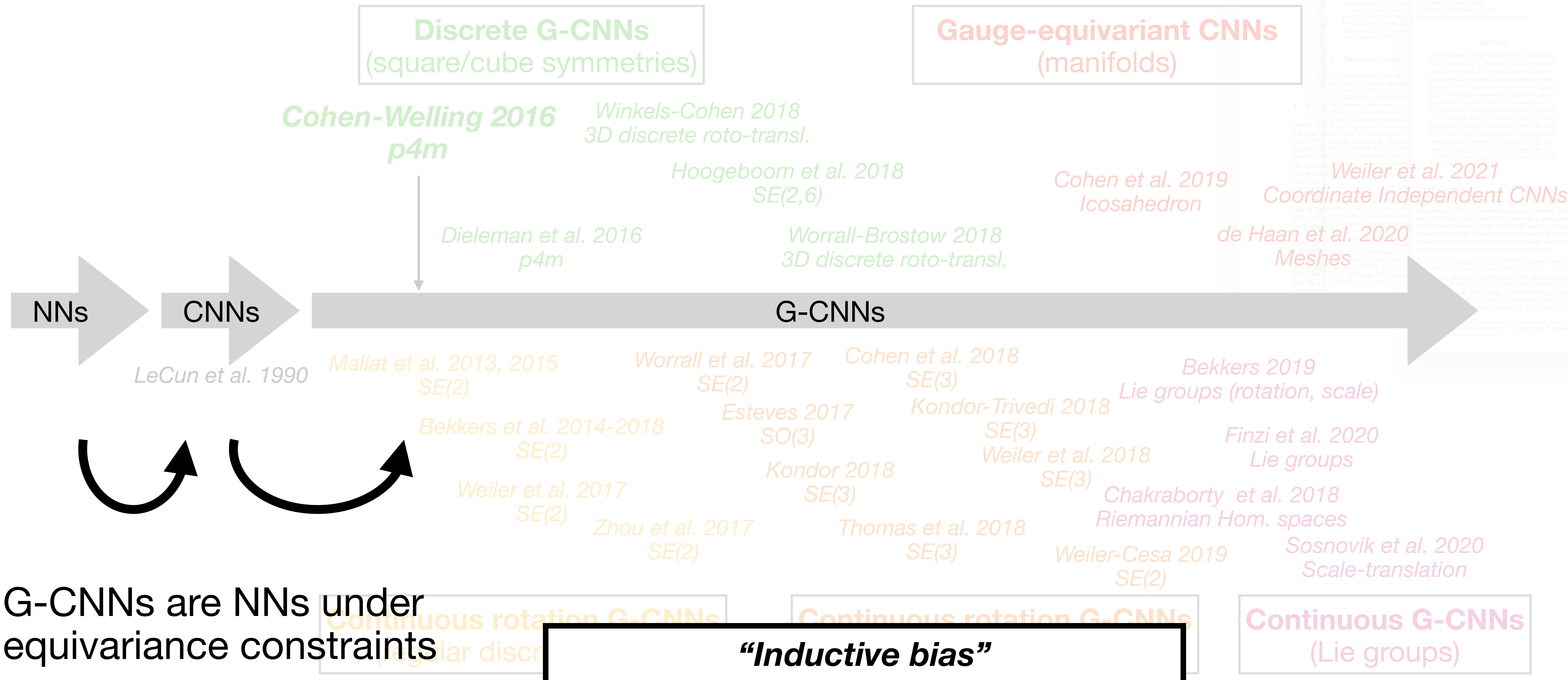
G-CNNs are NNs under equivariance constraints

Cesa-Lang-Weiler 2022
$G = \mathbb{R}^d \rtimes H$ with H compact

https://quva-lab.github.io/escnn/

A brief history of G-CNNs

https://github.com/Chen-Cai-OSU/awesome-equivariant-network



G-CNNs are NNs under equivariance constraints

"Inductive bias"

Reduce the search space for NNs

to only the sensible ones!

A brief history of G-CNNs

https://github.com/Chen-Cai-OSU/awesome-equivariant-network

Up next:
A linear map is equivariant
if and only if it is a group convolution

NNs → CNNs → G-CNNs

LeCun et al. 1990

Dieleman et al. 2016
p4m

Worrall-Brostow 2018
3D discrete roto-transl.

de Haan et al. 2020
Meshes

Mallat et al. 2013, 2015
SE(2)

Bekkers et al. 2014-2018
SE(2)

Weiler et al. 2017
SE(2)

Zhou et al. 2017
SE(2)

Worrall et al. 2017
SE(2)

Esteves 2017
SO(3)

Kondor 2018
SE(3)

Cohen et al. 2018
SE(3)

Kondor-Trivedi 2018
SE(3)

Weiler et al. 2018
SE(3)

Thomas et al. 2018
SE(3)

Weiler-Cesa 2019
SE(2)

Bekkers 2019
Lie groups (rotation, scale)

Finzi et al. 2020
Lie groups

Chakraborty et al. 2018
Riemannian Hom. spaces

Sosnovik et al. 2020
Scale-translation

Continuous G-CNNs
(Lie groups)

G-CNNs are NNs under
equivariance constraints

"Inductive bias"

Reduce the search space for NNs

to only the sensible ones!

https://quva-lab.github.io/escnn/