Group Equivariant Deep Learning

Lecture 3 - Equivariant graph neural networks

Lecture 3.6 - Regular (as opposed to steerable) equivariant graph NNs

Erik Bekkers, Amsterdam Machine Learning Lab, University of Amsterdam

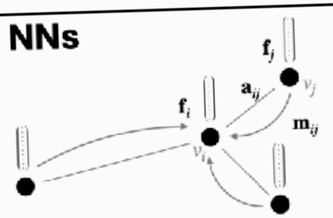
This mini-course serves as a module with the UvA Master AI course Deep Learning 2 https://uvadl2c.github.io/

Analysis of literature in terms of non-linear group convolutions

Lecture 2.2

Linear vs non-linear (group) convolutions

Message passing NNs



Compute messages: $\mathbf{m}_{ij} = \phi_m(\mathbf{f}_i, \mathbf{f}_j, \mathbf{a}_{ij})$

Aggregate and update: $\mathbf{f}'_i = \phi_f\left(\mathbf{f}_i, \sum_{j \in \mathcal{N}(i)} \mathbf{m}_{ij}\right)$

Classic point convolutions

(Lecture 1.7: regular g-convs on homogeneous spaces)

$$\mathbf{m}_{ij} = \mathbf{W}(\|\mathbf{x}_j - \mathbf{x}_i\|)\mathbf{f}_j$$

$$\mathbf{W}(g_i^{-1}g_j)\mathbf{f}_j$$

Linear convolution

Steerable G-CNNs

(Lecture 2: steerable g-convs)

$\mathcal{F}_H$

$$\mathbf{m}_{ij} = \mathbf{W}_{\hat{\mathbf{a}}_{ij}}(\|\mathbf{x}_j - \mathbf{x}_i\|)\hat{\mathbf{f}}_j$$

$$:= \hat{\mathbf{f}}_j \otimes_{cg}^{\mathbf{W}(\|\mathbf{x}_j - \mathbf{x}_i\|)} \hat{\mathbf{a}}_{ij}$$

Invariant Message **Passing NNs**

(Lecture 3)

$$\mathbf{m}_{ij} = \text{MLP}(\mathbf{f}_i, \mathbf{f}_j, \|\mathbf{x}_j - \mathbf{x}_i\|)$$

Non-linear "convolution"

Equivariant (Steerable) Message **Passing**

(Lecture 3)

$$\hat{\mathbf{m}}_{ij} = \widehat{\text{MLP}}\,(\hat{\mathbf{f}}_i, \hat{\mathbf{f}}_j, \mathbf{x}_j - \mathbf{x}_i)$$

With steerable MLP:

$$\widehat{\text{MLP}}_{\hat{\mathbf{a}}_{ij}}(\hat{\mathbf{f}}_i, \hat{\mathbf{f}}_j, \|\mathbf{x}_j - \mathbf{x}_i\|) := \sigma(\mathbf{W}^{(n)}_{\hat{\mathbf{a}}_{ij}}(\ldots(\sigma(\mathbf{W}^{(1)}_{\hat{\mathbf{a}}_{ij}}\hat{\mathbf{h}}_i))))$$

29

GEOMETRIC AND PHYSICAL QUANTITIES IMPROVE E(3) EQUIVARIANT MESSAGE PASSING

Johannes Brandstetter*
University of Amsterdam
Johannes Kepler University Linz
brandstetter@ml.jku.at

Rob Hesselink*
University of Amsterdam
r.d.hesselink@uva.nl

Elise van der Pol
UvA-Bosch DeltaLab
University of Amsterdam
e.e.vanderpol@uva.nl

Erik J Bekkers
University of Amsterdam
e.j.bekkers@uva.nl

Max Welling
UvA-Bosch DeltaLab
University of Amsterdam
m.welling@uva.nl

ABSTRACT

Including covariant information, such as position, force, velocity or spin is important in many tasks in computational physics and chemistry. We introduce Steerable E(3) Equivariant Graph Neural Networks (SEGNNs) that generalise equivariant graph networks, such that node and edge attributes are not restricted to invariant scalars, but can contain covariant information, such as vectors or tensors. This model, composed of steerable MLPs, is able to incorporate geometric and physical information in both the message and update functions. Through the definition of steerable node attributes, the MLPs provide a new class of activation functions for general use with steerable feature fields. We discuss ours and related work through the lens of *equivariant non-linear convolutions*, which further allows us to pin-point the successful components of SEGNNs: *non-linear* message aggregation improves upon classic *linear* (steerable) point convolutions; *steerable messages* improve upon recent equivariant graph networks that send invariant messages. We demonstrate the effectiveness of our method on several tasks in computational physics and chemistry and provide extensive ablation studies.

1 INTRODUCTION

The success of Convolutional Neural Networks (CNNs) (LeCun et al., 1998; 2015; Schmidhuber, 2015; Krizhevsky et al., 2012) is a key factor for the rise of deep learning, attributed to their capability of exploiting translation symmetries, thereby introducing a strong inductive bias. Recent work has shown that designing CNNs to exploit additional symmetries via group convolutions has even further increased their performance (Cohen & Welling, 2016; 2017; Worrall et al., 2017; Cohen et. al., 2018; Kondor & Trivedi, 2018; Weiler et al., 2018; Bekkers et al., 2018; Bekkers, 2019; Weiler & Cesa, 2019). Graph neural networks (GNNs) and CNNs are closely related to each other via their aggregation of local information. More precisely, CNNs can be formulated as message passing layers (Gilmer et al., 2017) based on a sum aggregation of messages that are obtained by relative position-dependent *linear* transformations of neighbouring node features. The power of message passing layers is, however, that node features are transformed and propagated in a highly *non-linear* manner. Equivariant GNNs have been proposed before as either PointConv-type (Wu et al., 2019; Kristof et al., 2017) implementations of steerable (Thomas et al., 2018; Anderson et al., 2019; Fuchs et al., 2020) or regular group convolutions (Finzi et al., 2020). The most important component in these methods are the convolution layers. Although powerful, such layers only (pseudo[1]) linearly transform the graphs and non-linearity is only obtained via point-wise activations.

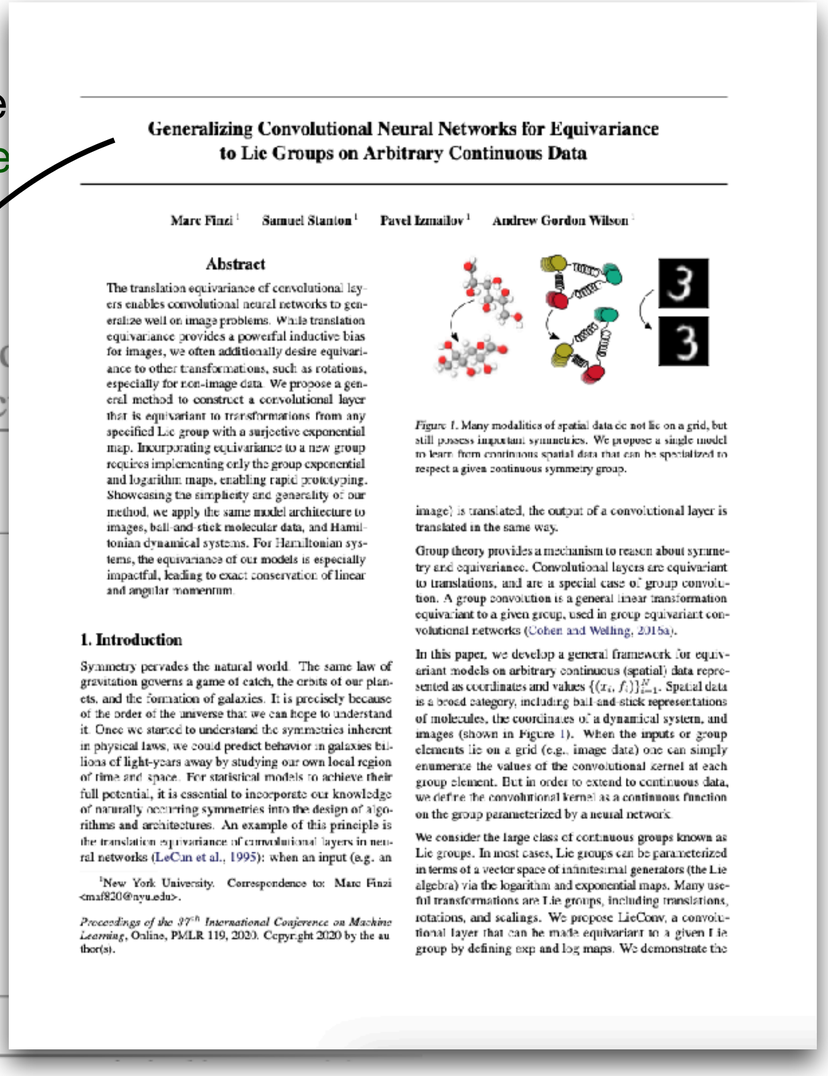
[1]Methods such as SE(3)-transformers (Fuchs et al., 2020) and Cormorant (Anderson et al., 2019) include an input-dependent attention component that augments the convolutions.

Linear vs Non-linear & Regular vs Steerable

Recall lecture 1.7:
"Any **equivariant linear layer** betwee
feature maps on homogeneous space
is a **group convolution**"

Table 2: Performance comparison
Error (MAE) between model predic

			Task	α	$\Delta\varepsilon$	$\varepsilon_{\text{HOMO}}$
			Units	bohr^3	meV	meV
non-linear		no geometry	NMP	.092	69	43
	regular	$\mathbb{R}^3$	SchNet *	.235	63	41
pseudo-linear	steerable	$SE(3)$	Cormorant	.085	61	34
	steerable	$SE(3)$	L1Net	.088	68	46
	regular	G	LieConv	.084	49	30
	steerable	$SE(3)$	TFN	.223	58	40
pseudo-linear	steerable	$SE(3)$	SE(3)-Tr.	.142	53	35
non-linear	regular	$\mathbb{R}^3 \times S^2 \times \mathbb{R}^+$	DimeNet++ *	**.043**	**32**	24
non-linear	regular	$\mathbb{R}^3 \times S^2 \times \mathbb{R}^+$	SphereNet *	.046	**32**	**23**
non-linear	reguleerable?	$SE(3)$	PaiNN *	.045	45	27
non-linear	regular	$\mathbb{R}^3$	EGNN	.071	48	29
non-linear	steerable	$SE(3)$	SEGNN (Ours)	.060	42	24



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

Marc Finzi [1] Samuel Stanton [1] Pavel Izmailov [1] Andrew Gordon Wilson [1]

Abstract

The translation equivariance of convolutional layers enables convolutional neural networks to generalize well on image problems. While translation equivariance provides a powerful inductive bias for images, we often additionally desire equivariance to other transformations, such as rotations, especially for non-image data. We propose a general method to construct a convolutional layer that is equivariant to transformations from any specified Lie group with a surjective exponential map. Incorporating equivariance to a new group requires implementing only the group exponential and logarithm maps, enabling rapid prototyping. Showcasing the simplicity and generality of our method, we apply the same model architecture to images, ball-and-stick molecular data, and Hamiltonian dynamical systems. For Hamiltonian systems, the equivariance of our models is especially impactful, leading to exact conservation of linear and angular momentum.

Figure 1. Many modalities of spatial data do not lie on a grid, but still possess important symmetries. We propose a single model to learn from continuous spatial data that can be specialized to respect a given continuous symmetry group.

1. Introduction

Symmetry pervades the natural world. The same law of gravitation governs a game of catch, the orbits of our planets, and the formation of galaxies. It is precisely because of the order of the universe that we can hope to understand it. Once we started to understand the symmetries inherent in physical laws, we could predict behavior in galaxies billions of light-years away by studying our own local region of time and space. For statistical models to achieve their full potential, it is essential to incorporate our knowledge of naturally occurring symmetries into the design of algorithms and architectures. An example of this principle is the translation equivariance of convolutional layers in neural networks (LeCun et al., 1995): when an input (e.g. an image) is translated, the output of a convolutional layer is translated in the same way.

Group theory provides a mechanism to reason about symmetry and equivariance. Convolutional layers are equivariant to translations, and are a special case of group convolution. A group convolution is a general linear transformation equivariant to a given group, used in group equivariant convolutional networks (Cohen and Welling, 2016a).

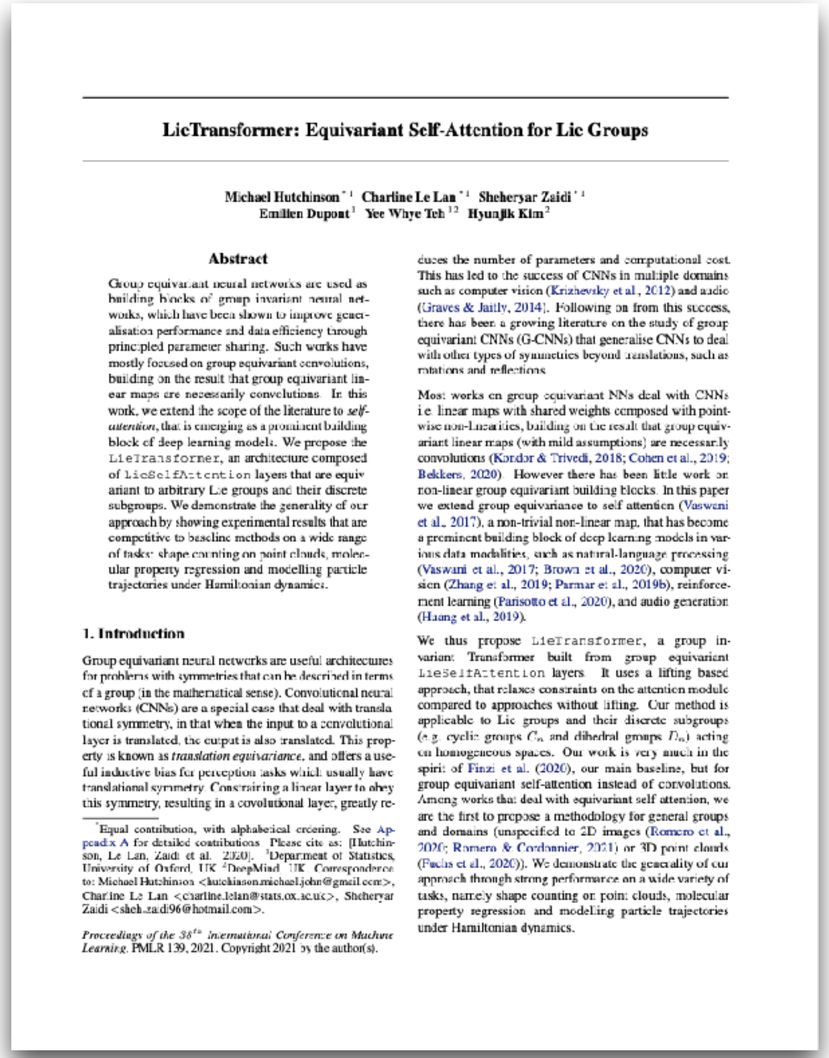
In this paper, we develop a general framework for equivariant models on arbitrary continuous (spatial) data represented as coordinates and values $\{(x_i, f_i)\}_{i=1}^M$. Spatial data is a broad category, including ball-and-stick representations of molecules, the coordinates of a dynamical system, and images (shown in Figure 1). When the inputs or group elements lie on a grid (e.g., image data) one can simply enumerate the values of the convolutional kernel at each group element. But in order to extend to continuous data, we define the convolutional kernel as a continuous function on the group parameterized by a neural network.

We consider the large class of continuous groups known as Lie groups. In most cases, Lie groups can be parameterized in terms of a vector space of infinitesimal generators (the Lie algebra) via the logarithm and exponential maps. Many useful transformations are Lie groups, including translations, rotations, and scalings. We propose LieConv, a convolutional layer that can be made equivariant to a given Lie group by defining exp and log maps. We demonstrate the

[1]New York University. Correspondence to: Marc Finzi <maf820@nyu.edu>.

Proceedings of the 37th International Conference on Machine Learning, Online, PMLR 119, 2020. Copyright 2020 by the author(s).

Linear vs Non-linear & Regular vs Steerable



LieTransformer: Equivariant Self-Attention for Lie Groups

Michael Hutchinson [*1] Charline Le Lan [*1] Sheheryar Zaidi [*1]
Emilien Dupont [1] Yee Whye Teh [1 2] Hyunjik Kim [2]

Abstract

Group equivariant neural networks are used as building blocks of group invariant neural networks, which have been shown to improve generalisation performance and data efficiency through principled parameter sharing. Such works have mostly focused on group equivariant convolutions, building on the result that group equivariant linear maps are necessarily convolutions. In this work, we extend the scope of the literature to *self-attention*, that is emerging as a prominent building block of deep learning models. We propose the `LieTransformer`, an architecture composed of `LieSelfAttention` layers that are equivariant to arbitrary Lie groups and their discrete subgroups. We demonstrate the generality of our approach by showing experimental results that are competitive to baseline methods on a wide range of tasks: shape counting on point clouds, molecular property regression and modelling particle trajectories under Hamiltonian dynamics.

1. Introduction

Group equivariant neural networks are useful architectures for problems with symmetries that can be described in terms of a group (in the mathematical sense). Convolutional neural networks (CNNs) are a special case that deal with translational symmetry, in that when the input to a convolutional layer is translated, the output is also translated. This property is known as *translation equivariance*, and offers a useful inductive bias for perception tasks which usually have translational symmetry. Constraining a linear layer to obey this symmetry, resulting in a convolutional layer, greatly reduces the number of parameters and computational cost. This has led to the success of CNNs in multiple domains such as computer vision (Krizhevsky et al., 2012) and audio (Graves & Jaitly, 2014). Following on from this success, there has been a growing literature on the study of group equivariant CNNs (G-CNNs) that generalise CNNs to deal with other types of symmetries beyond translations, such as rotations and reflections.

Most works on group equivariant NNs deal with CNNs i.e. linear maps with shared weights composed with pointwise non-linearities, building on the result that group equivariant linear maps (with mild assumptions) are necessarily convolutions (Kondor & Trivedi, 2018; Cohen et al., 2019; Bekkers, 2020). However there has been little work on non-linear group equivariant building blocks. In this paper we extend group equivariance to self attention (Vaswani et al., 2017), a non-trivial non-linear map, that has become a prominent building block of deep learning models in various data modalities, such as natural-language processing (Vaswani et al., 2017; Brown et al., 2020), computer vision (Zhang et al., 2019; Parmar et al., 2019b), reinforcement learning (Parisotto et al., 2020), and audio generation (Huang et al., 2019).

We thus propose `LieTransformer`, a group invariant Transformer built from group equivariant `LieSelfAttention` layers. It uses a lifting based approach, that relaxes constraints on the attention module compared to approaches without lifting. Our method is applicable to Lie groups and their discrete subgroups (e.g. cyclic groups C_n and dihedral groups D_n) acting on homogeneous spaces. Our work is very much in the spirit of Finzi et al. (2020), our main baseline, but for group equivariant self-attention instead of convolutions. Among works that deal with equivariant self attention, we are the first to propose a methodology for general groups and domains (unspecified to 2D images (Romero et al., 2020; Romero & Cordonnier, 2021) or 3D point clouds (Fuchs et al., 2020)). We demonstrate the generality of our approach through strong performance on a wide variety of tasks, namely: shape counting on point clouds, molecular property regression and modelling particle trajectories under Hamiltonian dynamics.

[*]Equal contribution, with alphabetical ordering. See Appendix A for detailed contributions. Please cite as: [Hutchinson, Le Lan, Zaidi et al. 2020]. [1]Department of Statistics, University of Oxford, UK [2]DeepMind, UK. Correspondence to: Michael Hutchinson <michaelhutchinson@gmail.com>, Charline Le Lan <charline.lelan@stats.ox.ac.uk>, Sheheryar Zaidi <sheh.zaidi96@hotmail.com>.

Proceedings of the 38th International Conference on Machine Learning, PMLR 139, 2021.

Linear vs Non-linear & Regular vs Steerable

LieTransformer: Equivariant Self-Attention for Lie Groups

Michael Hutchinson [*1] Charline Le Lan [*1] Sheheryar Zaidi [*1]
Emilien Dupont [1] Yee Whye Teh [1 2] Hyunjik Kim [2]

Abstract

Group equivariant neural networks are used as building blocks of group invariant neural networks, which have been shown to improve generalisation performance and data efficiency through principled parameter sharing. Such works have mostly focused on group equivariant convolutions, building on the result that group equivariant linear maps are necessarily convolutions. In this work, we extend the scope of the literature to *self-attention*, that is emerging as a prominent building block of deep learning models. We propose the `LieTransformer`, an architecture composed of `LieSelfAttention` layers that are equivariant to arbitrary Lie groups and their discrete subgroups. We demonstrate the generality of our approach by showing experimental results that are competitive to baseline methods on a wide range of tasks: shape counting on point clouds, molecular property regression and modelling particle trajectories under Hamiltonian dynamics.

1. Introduction

Group equivariant neural networks are useful architectures for problems with symmetries that can be described in terms of a group (in the mathematical sense). Convolutional neural networks (CNNs) are a special case that deal with translational symmetry, in that when the input to a convolutional layer is translated, the output is also translated. This property is known as *translation equivariance*, and offers a useful inductive bias for perception tasks which usually have translational symmetry. Constraining a linear layer to obey this symmetry, resulting in a convolutional layer, greatly reduces the number of parameters and computational cost. This has led to the success of CNNs in multiple domains such as computer vision (Krizhevsky et al., 2012) and audio (Graves & Jaitly, 2014). Following on from this success, there has been a growing literature on the study of group equivariant CNNs (G-CNNs) that generalise CNNs to deal with other types of symmetries beyond translations, such as rotations and reflections.

Most works on group equivariant NNs deal with CNNs i.e. linear maps with shared weights composed with pointwise non-linearities, building on the result that group equivariant linear maps (with mild assumptions) are necessarily convolutions (Kondor & Trivedi, 2018; Cohen et al., 2019; Bekkers, 2020). However there has been little work on non-linear group equivariant building blocks. In this paper we extend group equivariance to self attention (Vaswani et al., 2017), a non-trivial non-linear map, that has become a prominent building block of deep learning models in various data modalities, such as natural-language processing (Vaswani et al., 2017; Brown et al., 2020), computer vision (Zhang et al., 2019; Parmar et al., 2019b), reinforcement learning (Parisotto et al., 2020), and audio generation (Huang et al., 2019).

We thus propose `LieTransformer`, a group invariant Transformer built from group equivariant `LieSelfAttention` layers. It uses a lifting based approach, that relaxes constraints on the attention module compared to approaches without lifting. Our method is applicable to Lie groups and their discrete subgroups (e.g. cyclic groups C_n and dihedral groups D_n) acting on homogeneous spaces. Our work is very much in the spirit of Finzi et al. (2020), our main baseline, but for group equivariant self-attention instead of convolutions. Among works that deal with equivariant self attention, we are the first to propose a methodology for general groups and domains (unspecified to 2D images (Romero et al., 2020; Romero & Cordonnier, 2021) or 3D point clouds (Fuchs et al., 2020)). We demonstrate the generality of our approach through strong performance on a wide variety of tasks, namely shape counting on point clouds, molecular property regression and modelling particle trajectories under Hamiltonian dynamics.

[*]Equal contribution, with alphabetical ordering. See Appendix A for detailed contributions. Please cite as: {Hutchinson, Le Lan, Zaidi et al. 2020}. [1]Department of Statistics, University of Oxford, UK [2]DeepMind, UK. Correspondence to: Michael Hutchinson <michaelhutchinson.michael.john@gmail.com>, Charline Le Lan <charline.lelan@stats.ox.ac.uk>, Sheheryar Zaidi <sheh.zaidi96@hotmail.com>.

Proceedings of the 38th International Conference on Machine Learning, PMLR 139, 2021.

Attentive Group Equivariant Convolutional Networks

David W. Romero [1] Erik J. Bekkers [2] Jakub M. Tomczak [1] Mark Hoogendoorn [1]

Abstract

Although group convolutional networks are able to learn powerful representations based on symmetry patterns, they lack explicit means to learn meaningful relationships among them (e.g., relative positions and poses). In this paper, we present *attentive group equivariant convolutions*, a generalization of the group convolution, in which attention is applied during the course of convolution to accentuate meaningful symmetry combinations and suppress non-plausible, misleading ones. We indicate that prior work on visual attention can be described as special cases of our proposed framework and show empirically that our *attentive group equivariant convolutional networks* consistently outperform conventional group convolutional networks on benchmark image datasets. Simultaneously, we provide interpretability to the learned concepts through the visualization of equivariant attention maps.

1. Introduction

Convolutional Neural Networks (CNNs) (LeCun et al., 1989) have shown impressive performance in a wide variety of domains. The developments of CNNs as well as of many other machine learning approaches have been fueled by intuitions and insights into the composition and *modus operandi* of multiple biological systems (Wertheimer, 1938; Biederman, 1987; Dalahaut & Kutz, 2019; Blake & Lee, 2005; Zhaoping, 2014; Delahunt & Kutz, 2019). Though CNNs have achieved remarkable performance increases on several benchmark problems, their training efficiency as well as generalization capabilities are still open for improvement. One concept being exploited for this purpose is that of *equivariance*, again drawing inspiration from human beings.

Humans are able to identify familiar objects despite modifications in location, size, viewpoint, lighting conditions and background (Bruce & Humphreys, 1994). In addition, we do not just recognize them but are able to describe in detail the type and amount of modification applied to them as well (von Helmholtz, 1868; Cassirer, 1944; Schmidt et al., 2016). Equivariance is strongly related to the idea of *symmetricity*. As these modifications do not modify the essence of the underlying object, they should be treated (and learned) as a single concept. Recently, several approaches have embraced these ideas to preserve symmetries including translations (LeCun et al., 1989), planar rotations (Dieleman et al., 2016; Marcos et al., 2017; Worrall et al., 2017; Weiler et al., 2018b; Li et al., 2018; Cheng et al., 2018; Hoogeboom et al., 2018; Bekkers et al., 2018; Veeling et al., 2018; Lenssen et al., 2018; Smets et al., 2020), spherical rotations (Cohen et al., 2018; Worrall & Brostow, 2018; Weiler et al., 2018a; Thomas et al., 2018; Cohen et al., 2019b), scaling (Marcos et al., 2018; Worrall & Welling, 2019; Sosnovik et al., 2020) and general symmetry groups (Cohen & Welling, 2016a; Kondor & Trivedi, 2018; Weiler & Cesa, 2019; Cohen et al., 2019a; Bekkers, 2020; Romero & Hoogendoorn, 2020; Venkataraman et al., 2020).

Figure 1. Meaningful relationships among object symmetries. Though every figure is composed by the same elements, only the outermost examples resemble faces. The relative positions, orientations and scales of elements in the innermost examples do not match any meaningful face composition and hence, should not be labelled as such. Built upon Fig. 1 from Schwarzer (2000).

While group convolutional networks are able to learn powerful representations based on symmetry patterns, they lack any explicit means to learn meaningful relationships among them, e.g., relative positions, orientations and scales (Fig. 1). In this paper, we draw inspiration from another promising development in the machine learning domain driven by neuroscience and psychology (e.g., Pashler (2016)), *attention*, to learn such relationships. The notion of attention is related to the idea that not all components of an input signal are *per se* equally relevant for a particular task. As a consequence,

[1]Vrije Universiteit Amsterdam, [2]University of Amsterdam, The Netherlands. Correspondence to: David W. Romero <d.w.romeroguzman@vu.nl>.

Proceedings of the 37th International Conference on Machine Learning, Online, PMLR 119, 2020.

GROUP EQUIVARIANT STAND-ALONE SELF-ATTENTION FOR VISION

David W. Romero
Vrije Universiteit Amsterdam
d.w.romeroguzman@vu.nl

Jean-Baptiste Cordonnier
École Polytechnique Fédérale de Lausanne (EPFL)
jean-baptiste.cordonnier@epfl.ch

ABSTRACT

We provide a general self-attention formulation to impose group equivariance to arbitrary symmetry groups. This is achieved by defining positional encodings that are invariant to the action of the group considered. Since the group acts on the positional encoding directly, group equivariant self attention networks (`GSA-Nets`) are steerable by nature. Our experiments on vision benchmarks demonstrate consistent improvements of `GSA-Nets` over non-equivariant self-attention networks.

1 INTRODUCTION

Recent advances in Natural Language Processing have been largely attributed to the rise of the *Transformer* (Vaswani et al., 2017). Its key difference with previous methods, e.g., recurrent neural networks, convolutional neural networks (CNNs), is its ability to query information from all the input words simultaneously. This is achieved via the *self-attention operation* (Bahdanau et al., 2015; Cheng et al., 2016), which computes the similarity between representations of words in the sequence in the form of *attention scores*. Next, the representation of each word is updated based on the words with the highest attention scores. Inspired by the capacity of transformers to learn meaningful inter-word dependencies, researchers have started applying self-attention in vision tasks. It was first adopted into CNNs by channel-wise attention (Hu et al., 2018) and non-local spatial modeling (Wang et al., 2018). More recently, it has been proposed to replace CNNs with self-attention networks either partially (Bello et al., 2019) or entirely (Ramachandran et al., 2019). Contrary to discrete convolutional kernels, weights in self-attention are not tied to particular positions (Fig. A.1), yet self-attention layers are able to express any convolutional layer (Cordonnier et al., 2020). This flexibility allows leveraging long-range dependencies under a fixed parameter budget.

An arguable orthogonal advancement to deep learning architectures is the incorporation of symmetries into the model itself. The seminal work by Cohen & Welling (2016) provides a recipe to extend the *translation equivariance* of CNNs to other symmetry groups to improve generalization and sample-efficiency further (see §2). *Translation equivariance* is key to the success of CNNs. It describes the property that if a pattern is translated, its numerical descriptors are also translated, but not modified.

In this work, we introduce *group self-attention*, a self-attention formulation that grants equivariance to arbitrary symmetry groups. This is achieved by defining positional encodings invariant to the action of the group considered. In addition to generalization and sample-efficiency improvements provided by group equivariance, group equivariant self-attention networks (`GSA-Nets`) bring important benefits over group convolutional architectures: (*i*) *Parameter efficiency:* contrary to conventional discrete group convolutional kernels, where weights are tied to particular positions of neighborhoods or the group, group equivariant self-attention leverages long-range dependencies on group functions under a fixed parameter budget, yet it is able to express any group convolutional kernel. This allows for very expressive networks with low parameter count. (*ii*) *Steerability:* since the group acts directly on the positional encoding, `GSA-Nets` are *steerable* (Weiler et al., 2018b) by nature. This allows us to go beyond group discretizations that live in the grid without introducing interpolation artifacts.

Contributions:

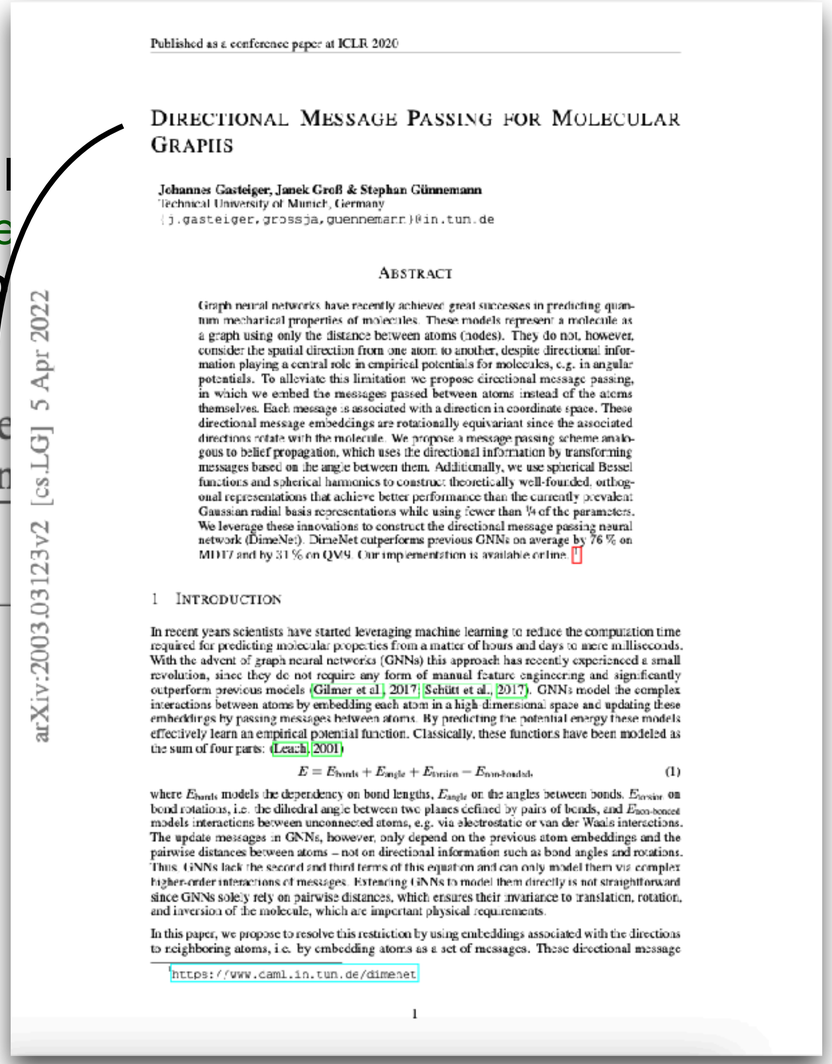
- We provide an extensive analysis on the equivariance properties of self-attention (§4).
- We provide a general formulation to impose group equivariance to self-attention (§5).
- We provide instances of self-attentive architectures equivariant to several symmetry groups (§6).
- Our results demonstrate consistent improvements of `GSA-Nets` over non-equivariant ones (§6).

Linear vs Non-linear & Regular vs Steerable

Recall lecture
"Any **equivariant linear**
feature maps on homoge
is a **group convo**

Table 2: Performanc
Error (MAE) betwee

			Task	α				
			Units	bohr^3				
non-linear		no geometry	NMP	.092				
	regular	$\mathbb{R}^3$	SchNet *	.235				
pseudo-linear	steerable	$SE(3)$	Cormorant	.085				
	steerable	$SE(3)$	L1Net	.088				
	regular	G	LieConv	.084				
	steerable	$SE(3)$	TFN	.223				
pseudo-linear	steerable	$SE(3)$	SE(3)-Tr.	.142				
non-linear	regular	$\mathbb{R}^3 \times S^2 \times \mathbb{R}^+$	DimeNet++ *	**.043**				
non-linear	regular	$\mathbb{R}^3 \times S^2 \times \mathbb{R}^+$	SphereNet *	.046				
non-linear	reguleerable?	$SE(3)$	PaiNN *	.045				
non-linear	regular	$\mathbb{R}^3$	EGNN	.071				
non-linear	steerable	$SE(3)$	SEGNN (Ours)	.060	42	24	21	.023	.031



Published as a conference paper at ICLR 2020

DIRECTIONAL MESSAGE PASSING FOR MOLECULAR GRAPHS

Johannes Gasteiger, Janek Groß & Stephan Günnemann
Technical University of Munich, Germany
{j.gasteiger,grossja,guennemann}@in.tum.de

ABSTRACT

Graph neural networks have recently achieved great successes in predicting quantum mechanical properties of molecules. These models represent a molecule as a graph using only the distance between atoms (nodes). They do not, however, consider the spatial direction from one atom to another, despite directional information playing a central role in empirical potentials for molecules, e.g. in angular potentials. To alleviate this limitation we propose directional message passing, in which we embed the messages passed between atoms instead of the atoms themselves. Each message is associated with a direction in coordinate space. These directional message embeddings are rotationally equivariant since the associated directions rotate with the molecule. We propose a message passing scheme analogous to belief propagation, which uses the directional information by transforming messages based on the angle between them. Additionally, we use spherical Bessel functions and spherical harmonics to construct theoretically well-founded, orthogonal representations that achieve better performance than the currently prevalent Gaussian radial basis representations while using fewer than ¼ of the parameters. We leverage these innovations to construct the directional message passing neural network (DimeNet). DimeNet outperforms previous GNNs on average by 76 % on MD17 and by 31 % on QM9. Our implementation is available online.[1]

1 INTRODUCTION

In recent years scientists have started leveraging machine learning to reduce the computation time required for predicting molecular properties from a matter of hours and days to mere milliseconds. With the advent of graph neural networks (GNNs) this approach has recently experienced a small revolution, since they do not require any form of manual feature engineering and significantly outperform previous models (Gilmer et al., 2017; Schütt et al., 2017). GNNs model the complex interactions between atoms by embedding each atom in a high dimensional space and updating these embeddings by passing messages between atoms. By predicting the potential energy these models effectively learn an empirical potential function. Classically, these functions have been modeled as the sum of four parts: (Leach, 2001)

$$E = E_{\text{bonds}} + E_{\text{angle}} + E_{\text{torsion}} - E_{\text{non-bonded}} \quad (1)$$

where E_{bonds} models the dependency on bond lengths, E_{angle} on the angles between bonds, E_{torsion} on bond rotations, i.e. the dihedral angle between two planes defined by pairs of bonds, and $E_{\text{non-bonded}}$ models interactions between unconnected atoms, e.g. via electrostatic or van der Waals interactions. The update messages in GNNs, however, only depend on the previous atom embeddings and the pairwise distances between atoms – not on directional information such as bond angles and rotations. Thus, GNNs lack the second and third terms of this equation and can only model them via complex higher-order interactions of messages. Extending GNNs to model them directly is not straightforward since GNNs solely rely on pairwise distances, which ensures their invariance to translation, rotation, and inversion of the molecule, which are important physical requirements.

In this paper, we propose to resolve this restriction by using embeddings associated with the directions to neighboring atoms, i.e. by embedding atoms as a set of messages. These directional message

[1] https://www.caml.in.tum.de/dimenet

1

Linear vs Non-linear & Regular vs Steerable

Recall lecture
"Any **equivariant linear**
feature maps on homoge
is a **group convo**

Table 2: Performanc
Error (MAE) betwee

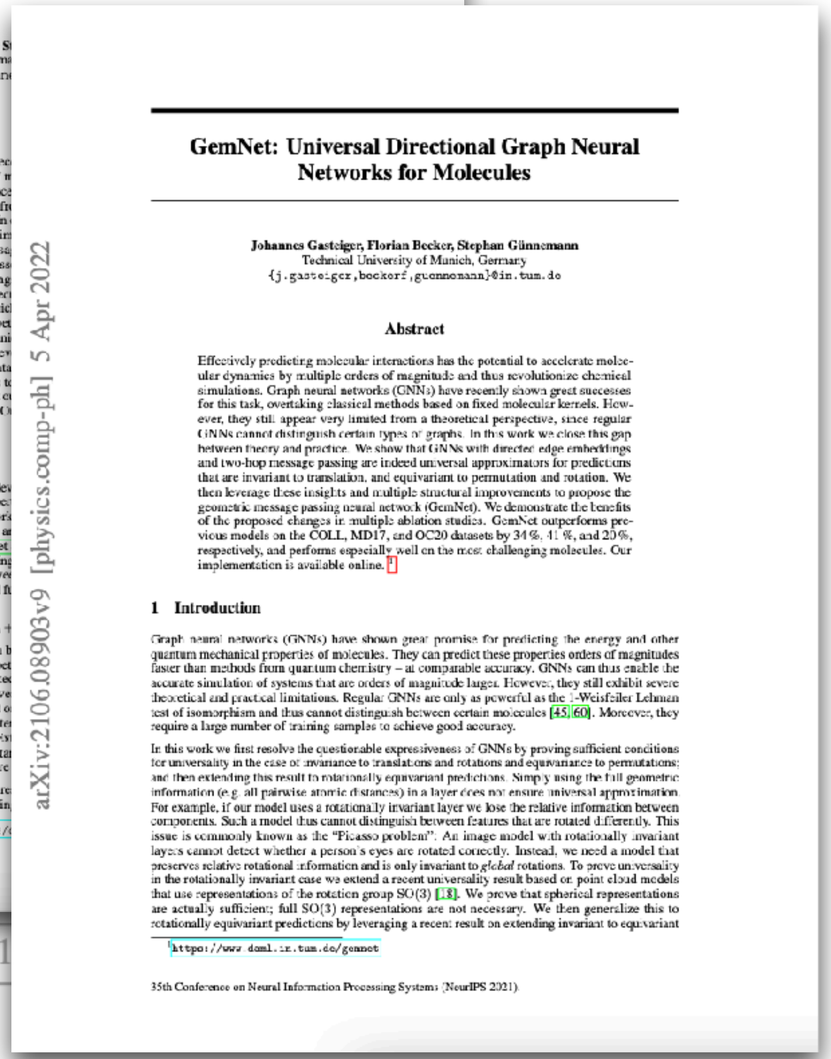
			Task	α			
			Units	bohr^3			
non-linear		no geometry	NMP	.092			
	regular	$\mathbb{R}^3$	SchNet *	.235			
pseudo-linear	steerable	$SE(3)$	Cormorant	.085			
	steerable	$SE(3)$	L1Net	.088			
	regular	G	LieConv	.084			
	steerable	$SE(3)$	TFN	.223			
pseudo-linear	steerable	$SE(3)$	SE(3)-Tr.	.142			
non-linear	regular	$\mathbb{R}^3 \times S^2 \times \mathbb{R}^+$	DimeNet++ *	**.043**			
non-linear	regular	$\mathbb{R}^3 \times S^2 \times \mathbb{R}^+$	SphereNet *	.046			
non-linear	reguleerable?	$SE(3)$	PaiNN *	.045			
non-linear	regular	$\mathbb{R}^3$	EGNN	.071			
non-linear	steerable	$SE(3)$	SEGNN (Ours)	.060	42	24	21


Published as a conference paper at ICLR 2020

DIRECTIONAL MESSAGE PASSING FOR MOLECULAR GRAPHS

Johannes Gasteiger, Janek Groß & S
Technical University of Munich, Germa
{j.gasteiger,grossja,guenn

arXiv:2003.03123v2 [cs.LG] 5 Apr 2022


GemNet: Universal Directional Graph Neural Networks for Molecules

Johannes Gasteiger, Florian Becker, Stephan Günnemann
Technical University of Munich, Germany
{j.gasteiger,beckerf,guennemann}@in.tum.de

Abstract

Effectively predicting molecular interactions has the potential to accelerate molecular dynamics by multiple orders of magnitude and thus revolutionize chemical simulations. Graph neural networks (GNNs) have recently shown great successes for this task, overtaking classical methods based on fixed molecular kernels. However, they still appear very limited from a theoretical perspective, since regular GNNs cannot distinguish certain types of graphs. In this work we close this gap between theory and practice. We show that GNNs with directed edge embeddings and two-hop message passing are indeed universal approximators for predictions that are invariant to translation, and equivariant to permutation and rotation. We then leverage these insights and multiple structural improvements to propose the geometric message passing neural network (GemNet). We demonstrate the benefits of the proposed changes in multiple ablation studies. GemNet outperforms previous models on the COLL, MD17, and OC20 datasets by 34 %, 41 %, and 20 %, respectively, and performs especially well on the most challenging molecules. Our implementation is available online.[1]

1 Introduction

Graph neural networks (GNNs) have shown great promise for predicting the energy and other quantum mechanical properties of molecules. They can predict these properties orders of magnitudes faster than methods from quantum chemistry – at comparable accuracy. GNNs can thus enable the accurate simulation of systems that are orders of magnitude larger. However, they still exhibit severe theoretical and practical limitations. Regular GNNs are only as powerful as the 1-Weisfeiler Lehman test of isomorphism and thus cannot distinguish between certain molecules [45, 60]. Moreover, they require a large number of training samples to achieve good accuracy.

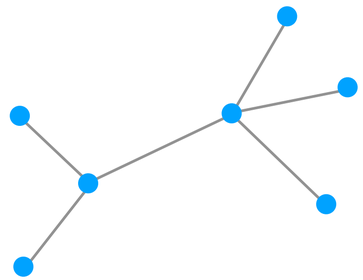
In this work we first resolve the questionable expressiveness of GNNs by proving sufficient conditions for universality in the case of invariance to translations and rotations and equivariance to permutations; and then extending this result to rotationally equivariant predictions. Simply using the full geometric information (e.g. all pairwise atomic distances) in a layer does not ensure universal approximation. For example, if our model uses a rotationally invariant layer we lose the relative information between components. Such a model thus cannot distinguish between features that are rotated differently. This issue is commonly known as the "Picasso problem": An image model with rotationally invariant layers cannot detect whether a person's eyes are rotated correctly. Instead, we need a model that preserves relative rotational information and is only invariant to *global* rotations. To prove universality in the rotationally invariant case we extend a recent universality result based on point cloud models that use representations of the rotation group SO(3) [18]. We prove that spherical representations are actually sufficient; full SO(3) representations are not necessary. We then generalize this to rotationally equivariant predictions by leveraging a recent result on extending invariant to equivariant

[1] https://www.daml.in.tum.de/gemnet

35th Conference on Neural Information Processing Systems (NeurIPS 2021).

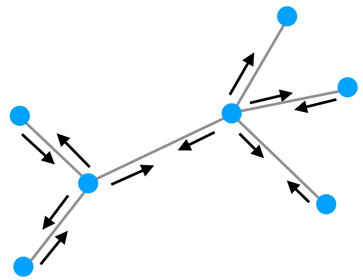
arXiv:2106.08903v9 [physics.comp-ph] 5 Apr 2022

Discrete regular group convolutions on $\mathbb{R}^3 \times S^2$



Atomic Point cloud
$\mathbb{R}^3$

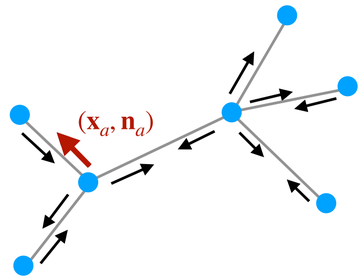
Discrete regular group convolutions on $\mathbb{R}^3 \times S^2$



Atomic Point cloud / Edge cloud

$\mathbb{R}^3$ $\mathbb{R}^3 \times S^2$

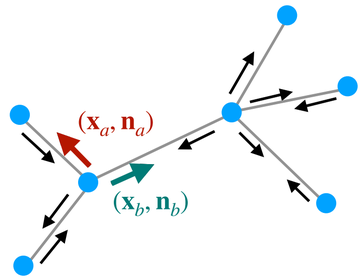
Discrete regular group convolutions on $\mathbb{R}^3 \times S^2$



Atomic Point cloud / Edge cloud

$\mathbb{R}^3$ $\mathbb{R}^3 \times S^2$

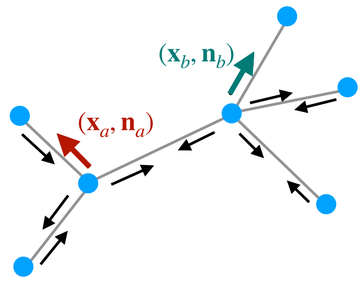
Discrete regular group convolutions on $\mathbb{R}^3 \times S^2$



Atomic Point cloud / Edge cloud

$\mathbb{R}^3$ $\mathbb{R}^3 \times S^2$

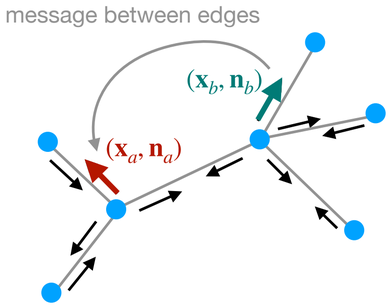
Discrete regular group convolutions on $\mathbb{R}^3 \times S^2$



Atomic Point cloud / Edge cloud

$\mathbb{R}^3$ $\mathbb{R}^3 \times S^2$

Discrete regular group convolutions on $\mathbb{R}^3 \times S^2$



Atomic Point cloud / Edge cloud

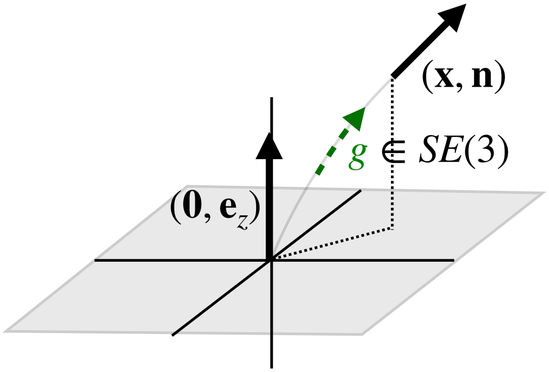
$\mathbb{R}^3$ $\mathbb{R}^3 \times S^2$

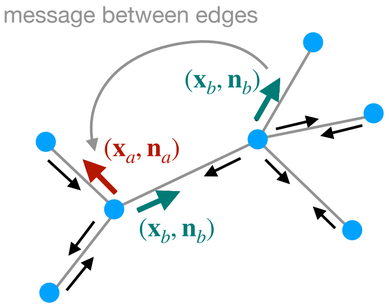
Discrete regular group convolutions on $\mathbb{R}^3 \times S^2$

The *transitive* action of $SE(3)$ on homogeneous space $\mathbb{R}^3 \times S^2 = SE(3)/SO(2)$

$$g^{-1} \cdot (\mathbf{x}', \mathbf{n}') = (\mathbf{R}^{-1}(\mathbf{x}' - \mathbf{x}), \mathbf{R}^{-1}\mathbf{n}') \in \mathbb{R}^3 \times S^2$$

Transitivity: $\quad \forall_{(\mathbf{x},\mathbf{n}) \in \mathbb{R}^3 \times S^2} \; \exists_{g_{(\mathbf{x},\mathbf{n})} \in SE(3)} : \; (\mathbf{x}, \mathbf{n}) = g_{(\mathbf{x},\mathbf{n})} \cdot (\mathbf{0}, \mathbf{e}_z)$





Atomic Point cloud / Edge cloud

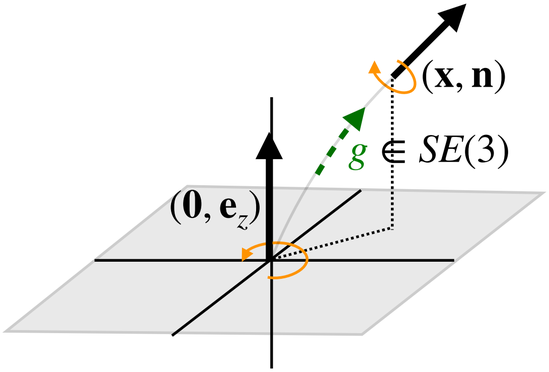
$\mathbb{R}^3$ $\quad$ $\mathbb{R}^3 \times S^2$

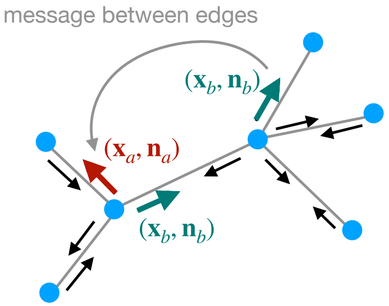
Discrete regular group convolutions on $\mathbb{R}^3 \times S^2$

The *transitive* action of $SE(3)$ on homogeneous space $\mathbb{R}^3 \times S^2 = SE(3)/SO(2)$

$$g^{-1} \cdot (\mathbf{x}', \mathbf{n}') = (\mathbf{R}^{-1}(\mathbf{x}' - \mathbf{x}), \mathbf{R}^{-1}\mathbf{n}') \in \mathbb{R}^3 \times S^2$$

Transitivity: $\quad \forall_{(\mathbf{x},\mathbf{n}) \in \mathbb{R}^3 \times S^2} \; \exists_{g_{(\mathbf{x},\mathbf{n})} \in SE(3)} : \; (\mathbf{x}, \mathbf{n}) = g_{(\mathbf{x},\mathbf{n})} \cdot (\mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{0}, \mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{e}_z)$





Atomic Point cloud / Edge cloud

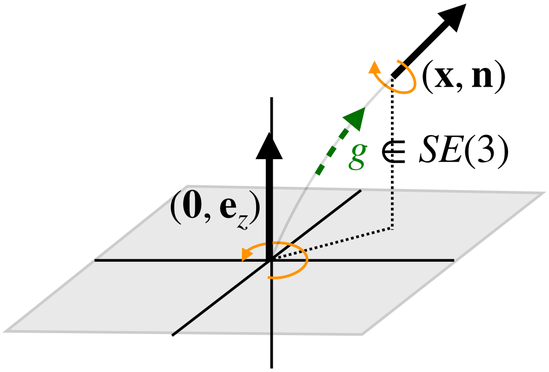
$\mathbb{R}^3$ $\quad$ $\mathbb{R}^3 \times S^2$

Discrete regular group convolutions on $\mathbb{R}^3 \times S^2$

The *transitive* action of $SE(3)$ on homogeneous space $\mathbb{R}^3 \times S^2 = SE(3)/SO(2)$

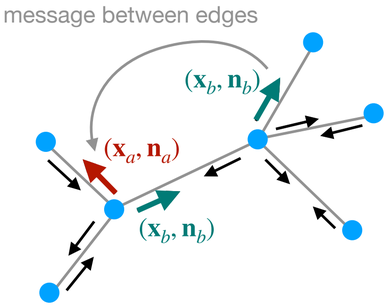
$$g^{-1} \cdot (\mathbf{x}', \mathbf{n}') = (\mathbf{R}^{-1}(\mathbf{x}' - \mathbf{x}), \mathbf{R}^{-1}\mathbf{n}') \in \mathbb{R}^3 \times S^2$$

Transitivity: $\forall_{(\mathbf{x},\mathbf{n}) \in \mathbb{R}^3 \times S^2} \exists_{g_{(\mathbf{x},\mathbf{n})} \in SE(3)} : \; (\mathbf{x}, \mathbf{n}) = g_{(\mathbf{x},\mathbf{n})} \cdot (\mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{0}, \mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{e}_z)$



Then $SE(3)$ equivariant convolution on homogeneous space $\mathbb{R}^3 \times S^2 = SE(3)/SO(2)$ are given by

$$f^{out}(\mathbf{x}_a, \mathbf{n}_a) = \int_{\mathbb{R}^d} \int_{S^2} k(g^{-1}_{(\mathbf{x}_a,\mathbf{n}_a)} \cdot (\mathbf{x}_b, \mathbf{n}_b)) f(\mathbf{x}_b, \mathbf{n}_b) \mathrm{d}\mathbf{x}_b \mathrm{d}\mathbf{n}_b$$



Atomic Point cloud / Edge cloud

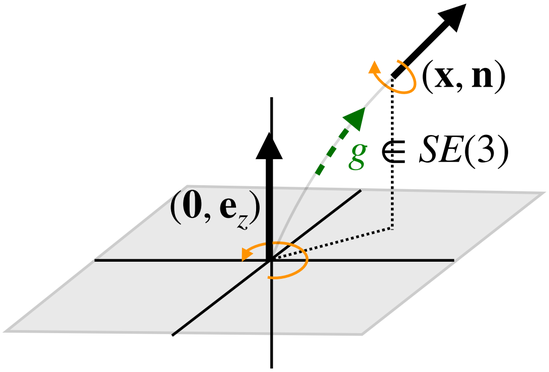
$\mathbb{R}^3$ $\mathbb{R}^3 \times S^2$

Discrete regular group convolutions on $\mathbb{R}^3 \times S^2$

The *transitive* action of $SE(3)$ on homogeneous space $\mathbb{R}^3 \times S^2 = SE(3)/SO(2)$

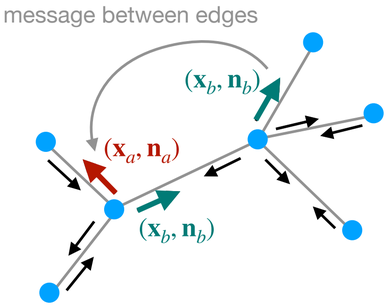
$$g^{-1} \cdot (\mathbf{x}', \mathbf{n}') = (\mathbf{R}^{-1}(\mathbf{x}' - \mathbf{x}), \mathbf{R}^{-1}\mathbf{n}') \in \mathbb{R}^3 \times S^2$$

Transitivity: $\quad \forall_{(\mathbf{x},\mathbf{n}) \in \mathbb{R}^3 \times S^2} \; \exists_{g_{(\mathbf{x},\mathbf{n})} \in SE(3)} : \; (\mathbf{x}, \mathbf{n}) = g_{(\mathbf{x},\mathbf{n})} \cdot (\mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{0}, \mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{e}_z)$



Then $SE(3)$ equivariant convolution on homogeneous space $\mathbb{R}^3 \times S^2 = SE(3)/SO(2)$ are given by

$$f^{out}(\mathbf{x}_a, \mathbf{n}_a) = \int_{\mathbb{R}^d} \int_{S^2} k(\mathbf{R}_{\mathbf{n}_a}^{-1}(\mathbf{x}_b - \mathbf{x}_a), \mathbf{R}_{\mathbf{n}_a}^{-1}\mathbf{n}_b) f(\mathbf{x}_b, \mathbf{n}_b) \mathrm{d}\mathbf{x}_b \mathrm{d}\mathbf{n}_b$$



Atomic Point cloud / Edge cloud

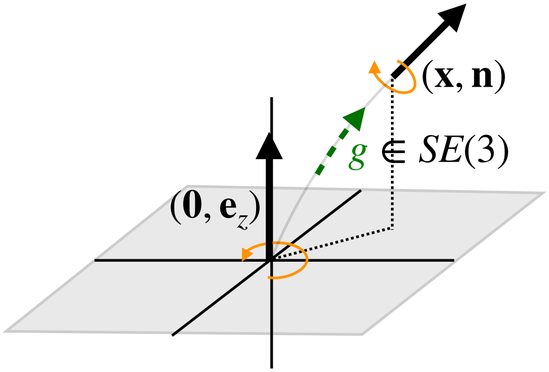
$\mathbb{R}^3$ $\quad$ $\mathbb{R}^3 \times S^2$

Discrete regular group convolutions on $\mathbb{R}^3 \times S^2$

The *transitive* action of $SE(3)$ on homogeneous space $\mathbb{R}^3 \times S^2 = SE(3)/SO(2)$

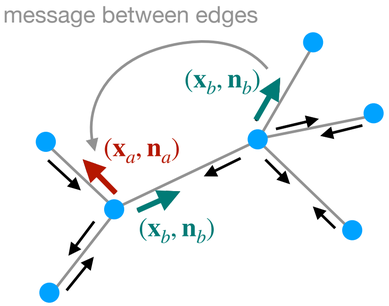
$$g^{-1} \cdot (\mathbf{x}', \mathbf{n}') = (\mathbf{R}^{-1}(\mathbf{x}' - \mathbf{x}), \mathbf{R}^{-1}\mathbf{n}') \in \mathbb{R}^3 \times S^2$$

Transitivity: $\forall_{(\mathbf{x},\mathbf{n}) \in \mathbb{R}^3 \times S^2} \; \exists_{g_{(\mathbf{x},\mathbf{n})} \in SE(3)} : \; (\mathbf{x}, \mathbf{n}) = g_{(\mathbf{x},\mathbf{n})} \cdot (\mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{0}, \mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{e}_z)$



Then $SE(3)$ equivariant convolution on homogeneous space $\mathbb{R}^3 \times S^2 = SE(3)/SO(2)$ are given by

$$f^{out}(\mathbf{x}_a, \mathbf{n}_a) = \sum_{\mathbf{x}_b \in \mathcal{N}_a} \int_{S^2} k(\mathbf{R}_{\mathbf{n}_a}^{-1}(\mathbf{x}_b - \mathbf{x}_a), \mathbf{R}_{\mathbf{n}_a}^{-1}\mathbf{n}_b) f(\mathbf{x}_b, \mathbf{n}_b) \mathrm{d}\mathbf{n}_b$$



Atomic Point cloud / Edge cloud

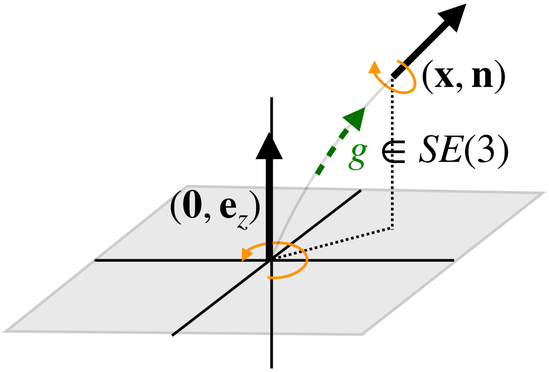
$\mathbb{R}^3$ $\mathbb{R}^3 \times S^2$

Discrete regular group convolutions on $\mathbb{R}^3 \times S^2$

The *transitive* action of $SE(3)$ on homogeneous space $\mathbb{R}^3 \times S^2 = SE(3)/SO(2)$

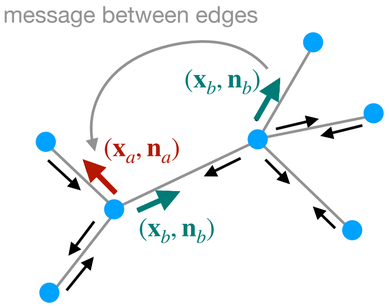
$$g^{-1} \cdot (\mathbf{x}', \mathbf{n}') = (\mathbf{R}^{-1}(\mathbf{x}' - \mathbf{x}), \mathbf{R}^{-1}\mathbf{n}') \in \mathbb{R}^3 \times S^2$$

Transitivity: $\forall_{(\mathbf{x},\mathbf{n}) \in \mathbb{R}^3 \times S^2} \; \exists_{g_{(\mathbf{x},\mathbf{n})} \in SE(3)} : \; (\mathbf{x}, \mathbf{n}) = g_{(\mathbf{x},\mathbf{n})} \cdot (\mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{0}, \mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{e}_z)$



Then $SE(3)$ equivariant convolution on homogeneous space $\mathbb{R}^3 \times S^2 = SE(3)/SO(2)$ are given by

$$f^{out}(\mathbf{x}_a, \mathbf{n}_a) = \sum_{\mathbf{x}_b \in \mathcal{N}_a} \sum_{\mathbf{n}_b \in \mathcal{R}_b} k(\mathbf{R}_{\mathbf{n}_a}^{-1}(\mathbf{x}_b - \mathbf{x}_a), \mathbf{R}_{\mathbf{n}_a}^{-1}\mathbf{n}_b) f(\mathbf{x}_b, \mathbf{n}_b)$$



Atomic Point cloud / Edge cloud

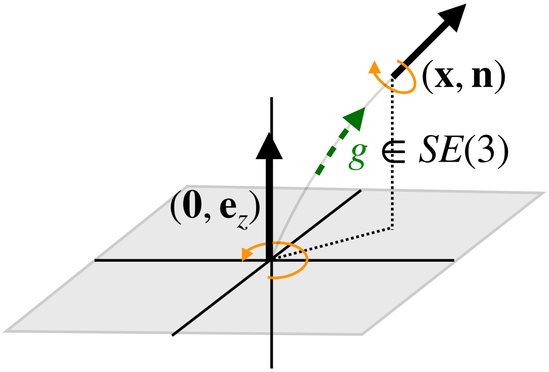
$\mathbb{R}^3$ $\mathbb{R}^3 \times S^2$

Discrete regular group convolutions on $\mathbb{R}^3 \times S^2$

The *transitive* action of $SE(3)$ on homogeneous space $\mathbb{R}^3 \times S^2 = SE(3)/SO(2)$

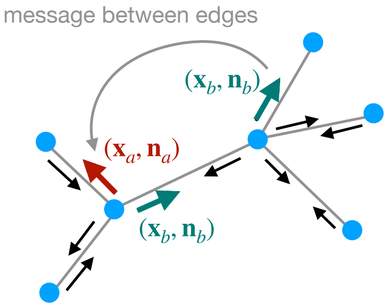
$$g^{-1} \cdot (\mathbf{x}', \mathbf{n}') = (\mathbf{R}^{-1}(\mathbf{x}' - \mathbf{x}), \mathbf{R}^{-1}\mathbf{n}') \in \mathbb{R}^3 \times S^2$$

Transitivity: $\forall_{(\mathbf{x},\mathbf{n}) \in \mathbb{R}^3 \times S^2} \; \exists_{g_{(\mathbf{x},\mathbf{n})} \in SE(3)} : \; (\mathbf{x}, \mathbf{n}) = g_{(\mathbf{x},\mathbf{n})} \cdot (\mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{0}, \mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{e}_z)$



Then $SE(3)$ equivariant convolution on homogeneous space $\mathbb{R}^3 \times S^2 = SE(3)/SO(2)$ are given by

$$f^{out}(\mathbf{x}_a, \mathbf{n}_a) = \sum_{\mathbf{x}_b \in \mathcal{N}_a} \sum_{\mathbf{n}_b \in \mathcal{R}_b} k(\mathbf{R}_{\mathbf{n}_a}^{-1}\mathbf{x}_{ba}, \mathbf{R}_{\mathbf{n}_a}^{-1}\mathbf{n}_b) f(\mathbf{x}_b, \mathbf{n}_b)$$



Atomic Point cloud / Edge cloud

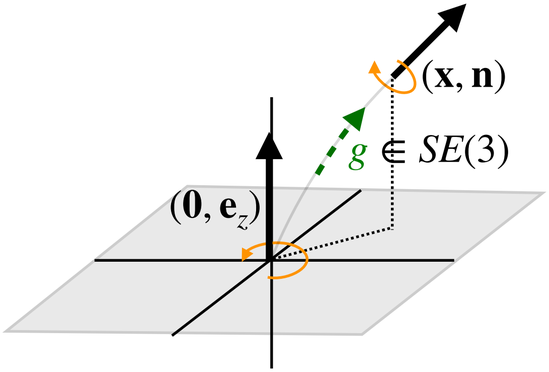
$\mathbb{R}^3$ $\mathbb{R}^3 \times S^2$

Discrete regular group convolutions on $\mathbb{R}^3 \times S^2$

The *transitive* action of $SE(3)$ on homogeneous space $\mathbb{R}^3 \times S^2 = SE(3)/SO(2)$

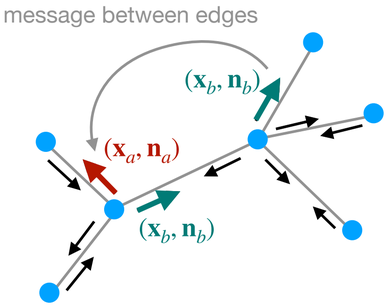
$$g^{-1} \cdot (\mathbf{x}', \mathbf{n}') = (\mathbf{R}^{-1}(\mathbf{x}' - \mathbf{x}), \mathbf{R}^{-1}\mathbf{n}') \in \mathbb{R}^3 \times S^2$$

Transitivity: $\quad \forall_{(\mathbf{x},\mathbf{n}) \in \mathbb{R}^3 \times S^2} \; \exists_{g_{(\mathbf{x},\mathbf{n})} \in SE(3)} : \; (\mathbf{x}, \mathbf{n}) = g_{(\mathbf{x},\mathbf{n})} \cdot (\mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{0}, \mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{e}_z)$



Then $SE(3)$ equivariant convolution on homogeneous space $\mathbb{R}^3 \times S^2 = SE(3)/SO(2)$ are given by

$$f^{out}(\mathbf{x}_a, \mathbf{n}_a) = \sum_{\mathbf{x}_b \in \mathcal{N}_a} \sum_{\mathbf{n}_b \in \mathcal{R}_b} k(\mathbf{R}_{\mathbf{n}_a}^{-1}\mathbf{x}_{ba}, \mathbf{R}_{\mathbf{n}_a}^{-1}\mathbf{n}_b) f(\mathbf{x}_b, \mathbf{n}_b)$$



Atomic Point cloud / Edge cloud

$\mathbb{R}^3$ $\quad$ $\mathbb{R}^3 \times S^2$

With kernel constraint

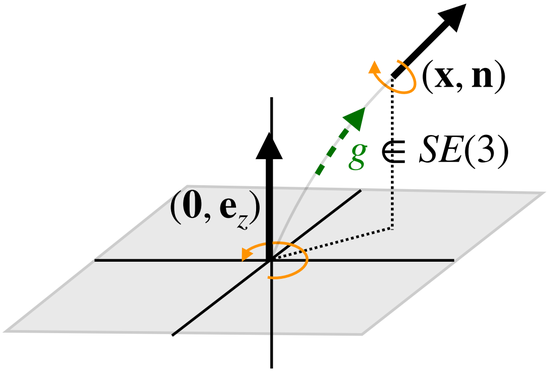
$$\forall_{\alpha \in [0, 2\pi)} : \quad k(\mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{x}, \mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{n}) = k(\mathbf{x}, \mathbf{n})$$

Discrete regular group convolutions on $\mathbb{R}^3 \times S^2$

The *transitive* action of $SE(3)$ on homogeneous space $\mathbb{R}^3 \times S^2 = SE(3)/SO(2)$

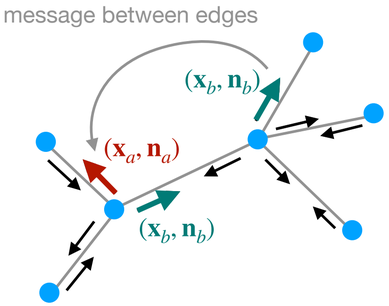
$$g^{-1} \cdot (\mathbf{x}', \mathbf{n}') = (\mathbf{R}^{-1}(\mathbf{x}' - \mathbf{x}), \mathbf{R}^{-1}\mathbf{n}') \in \mathbb{R}^3 \times S^2$$

Transitivity: $\quad \forall_{(\mathbf{x},\mathbf{n}) \in \mathbb{R}^3 \times S^2} \; \exists_{g_{(\mathbf{x},\mathbf{n})} \in SE(3)} : \; (\mathbf{x}, \mathbf{n}) = g_{(\mathbf{x},\mathbf{n})} \cdot (\mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{0}, \mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{e}_z)$



Then $SE(3)$ equivariant convolution on homogeneous space $\mathbb{R}^3 \times S^2 = SE(3)/SO(2)$ are given by

$$f^{out}(\mathbf{x}_a, \mathbf{n}_a) = \sum_{\mathbf{x}_b \in \mathcal{N}_a} \sum_{\mathbf{n}_b \in \mathcal{R}_b} k(\mathbf{R}_{\mathbf{n}_a}^{-1}\mathbf{x}_{ba}, \mathbf{R}_{\mathbf{n}_a}^{-1}\mathbf{n}_b) f(\mathbf{x}_b, \mathbf{n}_b)$$



Atomic Point cloud / Edge cloud
$\mathbb{R}^3$ $\quad$ $\mathbb{R}^3 \times S^2$

With kernel constraint which implies “conditioning” on invariant attributes:

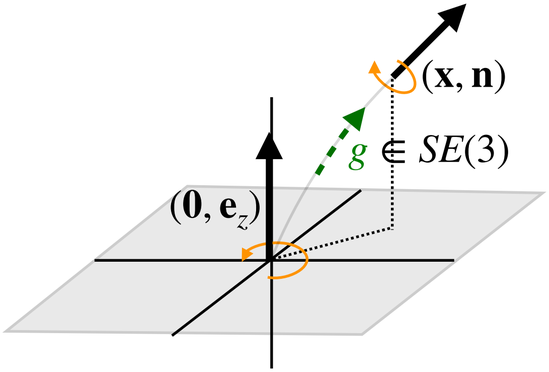
$$\forall_{\alpha \in [0,2\pi)} : \quad k(\mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{x}, \mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{n}) = k(\mathbf{x}, \mathbf{n}) \; = k(\underbrace{\|\mathbf{x}\|, \measuredangle \mathbf{x}\mathbf{e}_z}_{\text{in } \mathbf{x}}, \underbrace{\measuredangle \mathbf{n}\mathbf{e}_z}_{\text{in } \mathbf{n}})$$

Discrete regular group convolutions on $\mathbb{R}^3 \times S^2$

The *transitive* action of $SE(3)$ on homogeneous space $\mathbb{R}^3 \times S^2 = SE(3)/SO(2)$

$$g^{-1} \cdot (\mathbf{x}', \mathbf{n}') = (\mathbf{R}^{-1}(\mathbf{x}' - \mathbf{x}), \mathbf{R}^{-1}\mathbf{n}') \in \mathbb{R}^3 \times S^2$$

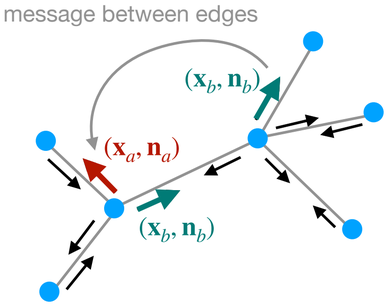
Transitivity: $\quad \forall_{(\mathbf{x},\mathbf{n}) \in \mathbb{R}^3 \times S^2} \; \exists_{g_{(\mathbf{x},\mathbf{n})} \in SE(3)} : \; (\mathbf{x}, \mathbf{n}) = g_{(\mathbf{x},\mathbf{n})} \cdot (\mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{0}, \mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{e}_z)$



Then $SE(3)$ equivariant convolution on homogeneous space $\mathbb{R}^3 \times S^2 = SE(3)/SO(2)$ are given by

$$f^{out}(\mathbf{x}_a, \mathbf{n}_a) = \sum_{\mathbf{x}_b \in \mathcal{N}_a} \sum_{\mathbf{n}_b \in \mathcal{R}_b} k(\mathbf{R}_{\mathbf{n}_a}^{-1}\mathbf{x}_{ba}, \mathbf{R}_{\mathbf{n}_a}^{-1}\mathbf{n}_b) f(\mathbf{x}_b, \mathbf{n}_b)$$

$$= \sum_{\mathbf{x}_b \in \mathcal{N}_a} \sum_{\mathbf{n}_b \in \mathcal{R}_b} k(\|\mathbf{R}_{\mathbf{n}_a}^{-1}\mathbf{x}_{ba}\| \; , \; \measuredangle(\mathbf{R}_{\mathbf{n}_a}^{-1}\mathbf{x}_{ba})\mathbf{e}_z \; , \; \measuredangle(\mathbf{R}_{\mathbf{n}_a}^{-1}\mathbf{n}_b)\mathbf{e}_z) \, f(\mathbf{x}_b, \mathbf{n}_b)$$



Atomic Point cloud / Edge cloud

$\mathbb{R}^3$ $\quad$ $\mathbb{R}^3 \times S^2$

With kernel constraint which implies “conditioning” on invariant attributes:

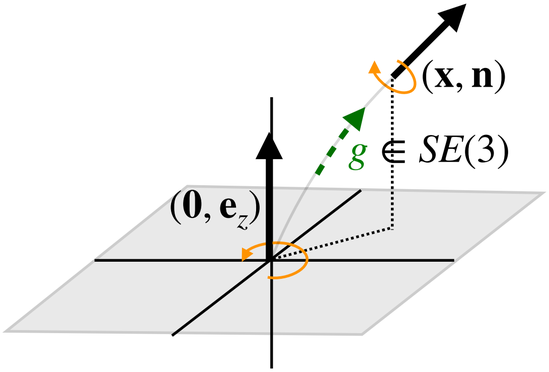
$$\forall_{\alpha \in [0,2\pi)} : \quad k(\mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{x}, \mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{n}) = k(\mathbf{x}, \mathbf{n}) \; = k(\underbrace{\|\mathbf{x}\|, \measuredangle\mathbf{x}\mathbf{e}_z}_{\text{in } \mathbf{x}}, \underbrace{\measuredangle\mathbf{n}\mathbf{e}_z}_{\text{in } \mathbf{n}})$$

Discrete regular group convolutions on $\mathbb{R}^3 \times S^2$

The *transitive* action of $SE(3)$ on homogeneous space $\mathbb{R}^3 \times S^2 = SE(3)/SO(2)$

$$g^{-1} \cdot (\mathbf{x}', \mathbf{n}') = (\mathbf{R}^{-1}(\mathbf{x}' - \mathbf{x}), \mathbf{R}^{-1}\mathbf{n}') \in \mathbb{R}^3 \times S^2$$

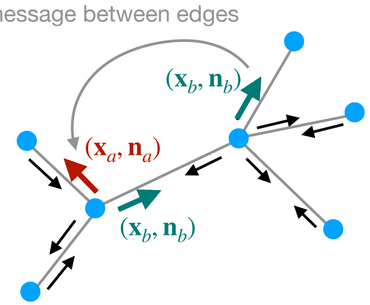
Transitivity: $\quad \forall_{(\mathbf{x},\mathbf{n}) \in \mathbb{R}^3 \times S^2} \; \exists_{g_{(\mathbf{x},\mathbf{n})} \in SE(3)} : \; (\mathbf{x}, \mathbf{n}) = g_{(\mathbf{x},\mathbf{n})} \cdot (\mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{0}, \mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{e}_z)$



Then $SE(3)$ equivariant convolution on homogeneous space $\mathbb{R}^3 \times S^2 = SE(3)/SO(2)$ are given by

$$f^{out}(\mathbf{x}_a, \mathbf{n}_a) = \sum_{\mathbf{x}_b \in \mathcal{N}_a} \sum_{\mathbf{n}_b \in \mathcal{R}_b} k(\mathbf{R}_{\mathbf{n}_a}^{-1}\mathbf{x}_{ba}, \mathbf{R}_{\mathbf{n}_a}^{-1}\mathbf{n}_b) f(\mathbf{x}_b, \mathbf{n}_b)$$

$$= \sum_{\mathbf{x}_b \in \mathcal{N}_a} \sum_{\mathbf{n}_b \in \mathcal{R}_b} k(\|\mathbf{x}_{ba}\| \; , \; \measuredangle(\mathbf{R}_{\mathbf{n}_a}^{-1}\mathbf{x}_{ba})\mathbf{e}_z \; , \; \measuredangle(\mathbf{R}_{\mathbf{n}_a}^{-1}\mathbf{n}_b)\mathbf{e}_z) \, f(\mathbf{x}_b, \mathbf{n}_b)$$



Atomic Point cloud / Edge cloud

$\mathbb{R}^3$ $\quad$ $\mathbb{R}^3 \times S^2$

With kernel constraint which implies “conditioning” on invariant attributes:

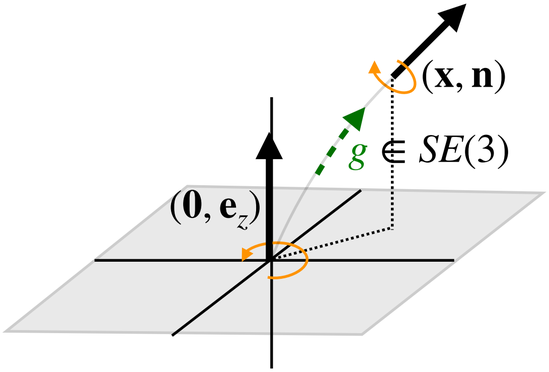
$$\forall_{\alpha \in [0,2\pi)} : \quad k(\mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{x}, \mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{n}) = k(\mathbf{x}, \mathbf{n}) \; = k(\underbrace{\|\mathbf{x}\|, \measuredangle\mathbf{x}\mathbf{e}_z}_{\text{in } \mathbf{x}}, \underbrace{\measuredangle\mathbf{n}\mathbf{e}_z}_{\text{in } \mathbf{n}})$$

Discrete regular group convolutions on $\mathbb{R}^3 \times S^2$

The *transitive* action of $SE(3)$ on homogeneous space $\mathbb{R}^3 \times S^2 = SE(3)/SO(2)$

$$g^{-1} \cdot (\mathbf{x}', \mathbf{n}') = (\mathbf{R}^{-1}(\mathbf{x}' - \mathbf{x}), \mathbf{R}^{-1}\mathbf{n}') \in \mathbb{R}^3 \times S^2$$

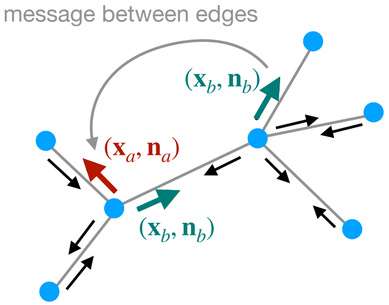
Transitivity: $\quad \forall_{(\mathbf{x},\mathbf{n}) \in \mathbb{R}^3 \times S^2} \; \exists_{g_{(\mathbf{x},\mathbf{n})} \in SE(3)} : \; (\mathbf{x}, \mathbf{n}) = g_{(\mathbf{x},\mathbf{n})} \cdot (\mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{0}, \mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{e}_z)$



Then $SE(3)$ equivariant convolution on homogeneous space $\mathbb{R}^3 \times S^2 = SE(3)/SO(2)$ are given by

$$f^{out}(\mathbf{x}_a, \mathbf{n}_a) = \sum_{\mathbf{x}_b \in \mathcal{N}_a} \sum_{\mathbf{n}_b \in \mathcal{R}_b} k(\mathbf{R}_{\mathbf{n}_a}^{-1}\mathbf{x}_{ba}, \mathbf{R}_{\mathbf{n}_a}^{-1}\mathbf{n}_b) f(\mathbf{x}_b, \mathbf{n}_b)$$

$$= \sum_{\mathbf{x}_b \in \mathcal{N}_a} \sum_{\mathbf{n}_b \in \mathcal{R}_b} k(\|\mathbf{x}_{ba}\| \; , \; \measuredangle(\mathbf{R}_{\mathbf{n}_a}^{-1}\mathbf{x}_{ba})\mathbf{e}_z \; , \; \measuredangle(\mathbf{R}_{\mathbf{n}_a}^{-1}\mathbf{n}_b)\mathbf{e}_z) \, f(\mathbf{x}_b, \mathbf{n}_b)$$



Atomic Point cloud / Edge cloud
$\mathbb{R}^3$ $\mathbb{R}^3 \times S^2$

With kernel constraint which implies "conditioning" on invariant attributes:

$$\forall_{\alpha \in [0,2\pi)} : \quad k(\mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{x}, \mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{n}) = k(\mathbf{x}, \mathbf{n}) \; = k(\underbrace{\|\mathbf{x}\|, \measuredangle\mathbf{x}\mathbf{e}_z}_{\text{in } \mathbf{x}}, \underbrace{\measuredangle\mathbf{n}\mathbf{e}_z}_{\text{in } \mathbf{n}})$$

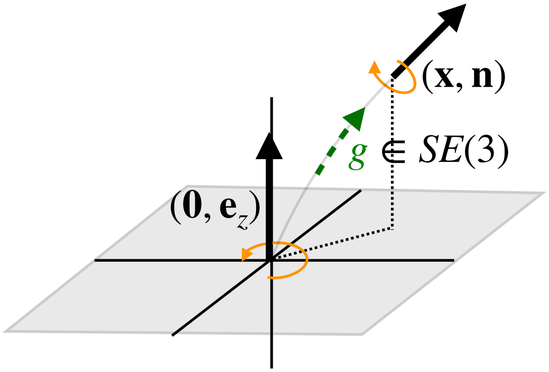
$$\begin{aligned}\measuredangle(\mathbf{R}_{\mathbf{n}_a}^{-1}\mathbf{x}_{ba})\mathbf{e}_z &= \arccos(\mathbf{R}_{\mathbf{n}_a}^{-1}\hat{\mathbf{x}}_{ba})^T\mathbf{e}_z \\ &= \arccos \hat{\mathbf{x}}_{ba}^T(\mathbf{R}_{\mathbf{n}_a}\mathbf{e}_z) \\ &= \arccos \hat{\mathbf{x}}_{ba}^T\mathbf{n}_a \\ &= \measuredangle\mathbf{x}_{ba}\mathbf{n}_a\end{aligned}$$

Discrete regular group convolutions on $\mathbb{R}^3 \times S^2$

The *transitive* action of $SE(3)$ on homogeneous space $\mathbb{R}^3 \times S^2 = SE(3)/SO(2)$

$$g^{-1} \cdot (\mathbf{x}', \mathbf{n}') = (\mathbf{R}^{-1}(\mathbf{x}' - \mathbf{x}), \mathbf{R}^{-1}\mathbf{n}') \in \mathbb{R}^3 \times S^2$$

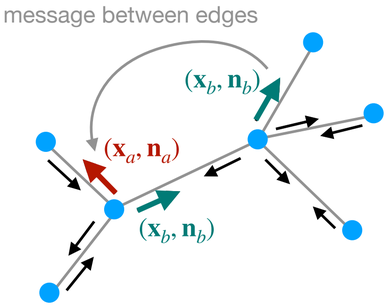
Transitivity: $\quad \forall_{(\mathbf{x},\mathbf{n}) \in \mathbb{R}^3 \times S^2} \; \exists_{g_{(\mathbf{x},\mathbf{n})} \in SE(3)} : \; (\mathbf{x}, \mathbf{n}) = g_{(\mathbf{x},\mathbf{n})} \cdot (\mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{0}, \mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{e}_z)$



Then $SE(3)$ equivariant convolution on homogeneous space $\mathbb{R}^3 \times S^2 = SE(3)/SO(2)$ are given by

$$f^{out}(\mathbf{x}_a, \mathbf{n}_a) = \sum_{\mathbf{x}_b \in \mathcal{N}_a} \sum_{\mathbf{n}_b \in \mathcal{R}_b} k(\mathbf{R}_{\mathbf{n}_a}^{-1}\mathbf{x}_{ba}, \mathbf{R}_{\mathbf{n}_a}^{-1}\mathbf{n}_b) f(\mathbf{x}_b, \mathbf{n}_b)$$

$$= \sum_{\mathbf{x}_b \in \mathcal{N}_a} \sum_{\mathbf{n}_b \in \mathcal{R}_b} k(\|\mathbf{x}_{ba}\| \; , \; \measuredangle \mathbf{n}_a \mathbf{x}_{ba} \; , \; \measuredangle \mathbf{n}_a \mathbf{n}_b) \, f(\mathbf{x}_b, \mathbf{n}_b)$$



Atomic Point cloud / Edge cloud

$\mathbb{R}^3$ $\quad$ $\mathbb{R}^3 \times S^2$

With kernel constraint which implies “conditioning” on invariant attributes:

$$\forall_{\alpha \in [0,2\pi)} : \quad k(\mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{x}, \mathbf{R}_{\alpha,\mathbf{e}_z}\mathbf{n}) = k(\mathbf{x}, \mathbf{n}) \; = k(\underbrace{\|\mathbf{x}\|, \measuredangle \mathbf{x}\mathbf{e}_z}_{\text{in } \mathbf{x}}, \underbrace{\measuredangle \mathbf{n}\mathbf{e}_z}_{\text{in } \mathbf{n}})$$

Linear vs Non-linear & Regular vs Steerable

Recall lecture
"Any **equivariant linear**
feature maps on homoge
is a **group convo**

Table 2: Performanc
Error (MAE) betwee

			Task	α			
			Units	bohr^3			
non-linear		no geometry	NMP	.092			
	regular	$\mathbb{R}^3$	SchNet *	.235			
pseudo-linear	steerable	$SE(3)$	Cormorant	.085			
	steerable	$SE(3)$	L1Net	.088			
	regular	G	LieConv	.084			
	steerable	$SE(3)$	TFN	.223			
pseudo-linear	steerable	$SE(3)$	SE(3)-Tr.	.142			
non-linear	regular	$\mathbb{R}^3 \times S^2 \times \mathbb{R}^+$	DimeNet++ *	**.043**			
non-linear	regular	$\mathbb{R}^3 \times S^2 \times \mathbb{R}^+$	SphereNet *	.046			
non-linear	reguleerable?	$SE(3)$	PaiNN *	.045			
non-linear	regular	$\mathbb{R}^3$	EGNN	.071			
non-linear	steerable	$SE(3)$	SEGNN (Ours)	.060	42	24	21


Published as a conference paper at ICLR 2020

DIRECTIONAL MESSAGE PASSING FOR MOLECULAR GRAPHS

Johannes Gasteiger, Janek Groß & S
Technical University of Munich, Germa
{j.gasteiger,grossja,gianr

Graph neural networks have rec
tum mechanical properties of m
a graph using only the distance
consider the spatial direction fr
mation playing a central role in
potentials. To alleviate this lim
in which we embed the messa
themselves. Each message is ass
directional message embedding
directions rotate with the molec
gous to belief propagation, whic
messages based on the angle bet
functions and spherical harmoni
onal representations that achiev
Gaussian radial basis representa
We leverage these innovations t
network (DimeNet). DimeNet o
MD17 and by 31 % on QM9. O

1 INTRODUCTION

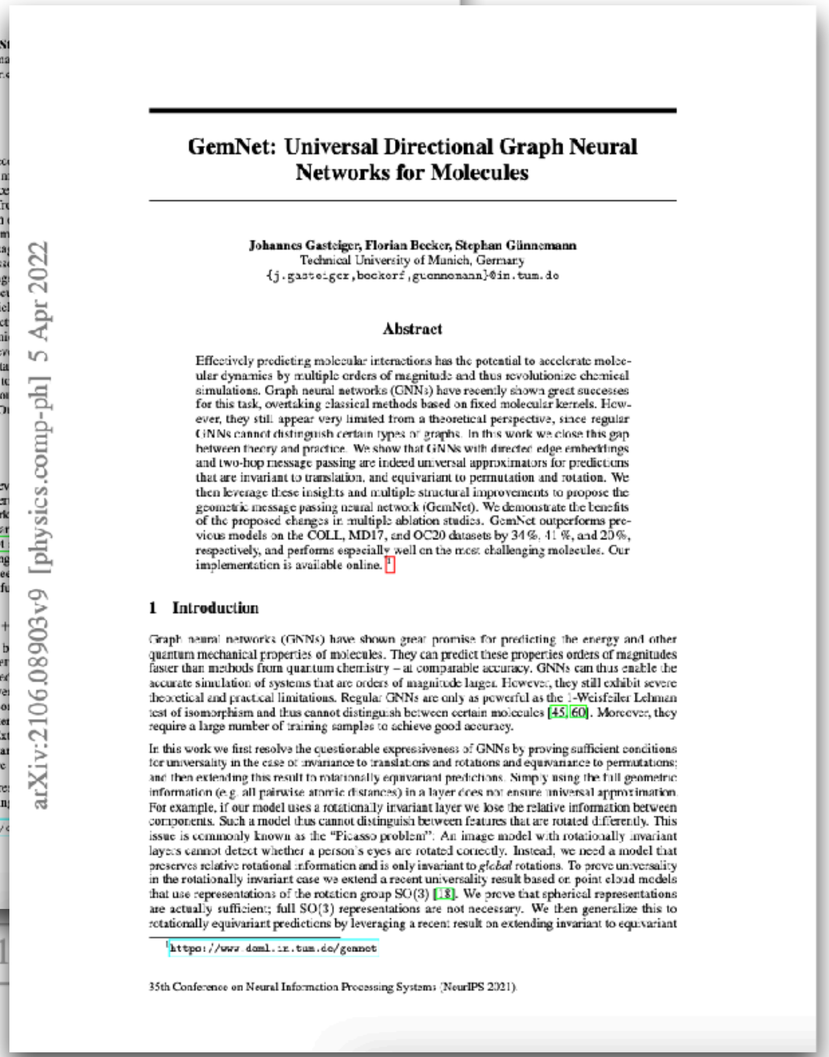
In recent years scientists have started lev
required for predicting molecular proper
With the advent of graph neural network
revolution, since they do not require a
outperform previous models (Gilmer et
interactions between atoms by embedding
embeddings by passing messages betwe
effectively learn an empirical potential fu
the sum of four parts: (Leach, 2001)

$$E = E_{\text{bonds}} +$$

where E_{bonds} models the dependency on b
bond rotations, i.e. the dihedral angle bet
models interactions between unconnected
The update messages in GNNs, however
pairwise distances between atoms – not o
Thus, GNNs lack the second and third te
higher order interactions of messages. Ex
since GNNs solely rely on pairwise distan
and inversion of the molecule, which are

In this paper, we propose to resolve this re
to neighboring atoms, i.e. by embeddin

[1] https://www.daml.in.tum.de/c



GemNet: Universal Directional Graph Neural Networks for Molecules

Johannes Gasteiger, Florian Becker, Stephan Günnemann
Technical University of Munich, Germany
{j.gasteiger,becker,guennemann}@in.tum.de

Abstract

Effectively predicting molecular interactions has the potential to accelerate molecular dynamics by multiple orders of magnitude and thus revolutionize chemical simulations. Graph neural networks (GNNs) have recently shown great successes for this task, overtaking classical methods based on fixed molecular kernels. However, they still appear very limited from a theoretical perspective, since regular GNNs cannot distinguish certain types of graphs. In this work we close this gap between theory and practice. We show that GNNs with directed edge embeddings and two-hop message passing are indeed universal approximators for predictions that are invariant to translation, and equivariant to permutation and rotation. We then leverage these insights and multiple structural improvements to propose the geometric message passing neural network (GemNet). We demonstrate the benefits of the proposed changes in multiple ablation studies. GemNet outperforms previous models on the COLL, MD17, and OC20 datasets by 34 %, 41 %, and 20 %, respectively, and performs especially well on the most challenging molecules. Our implementation is available online.[1]

1 Introduction

Graph neural networks (GNNs) have shown great promise for predicting the energy and other quantum mechanical properties of molecules. They can predict these properties orders of magnitudes faster than methods from quantum chemistry – at comparable accuracy. GNNs can thus enable the accurate simulation of systems that are orders of magnitude larger. However, they still exhibit severe theoretical and practical limitations. Regular GNNs are only as powerful as the 1-Weisfeiler Lehman test of isomorphism and thus cannot distinguish between certain molecules [45, 60]. Moreover, they require a large number of training samples to achieve good accuracy.

In this work we first resolve the questionable expressiveness of GNNs by proving sufficient conditions for universality in the case of invariance to translations and rotations and equivariance to permutations; and then extending this result to rotationally equivariant predictions. Simply using the full geometric information (e.g. all pairwise atomic distances) in a layer does not ensure universal approximation. For example, if our model uses a rotationally invariant layer we lose the relative information between components. Such a model thus cannot distinguish between features that are rotated differently. This issue is commonly known as the "Picasso problem": An image model with rotationally invariant layers cannot detect whether a person's eyes are rotated correctly. Instead, we need a model that preserves relative rotational information and is only invariant to *global* rotations. To prove universality in the rotationally invariant case we extend a recent universality result based on point cloud models that use representations of the rotation group SO(3) [18]. We prove that spherical representations are actually sufficient; full SO(3) representations are not necessary. We then generalize this to rotationally equivariant predictions by leveraging a recent result on extending invariant to equivariant

[1] https://www.daml.in.tum.de/gemnet

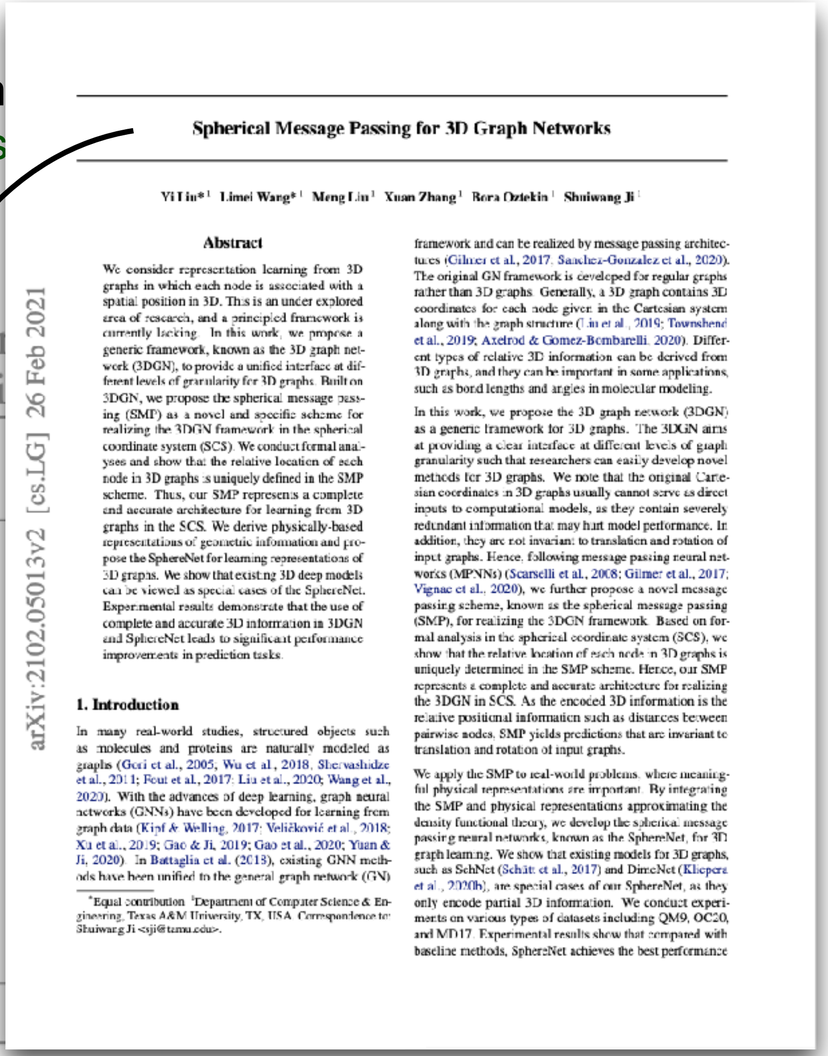
35th Conference on Neural Information Processing Systems (NeurIPS 2021).

Linear vs Non-linear & Regular vs Steerable

Recall lecture 1.7:
"Any **equivariant linear layer** between feature maps on homogeneous spaces is a **group convolution**"

			Task	α	$\Delta\varepsilon$	$\varepsilon_{\text{HOMO}}$
			Units	bohr^3	meV	meV
non-linear		no geometry	NMP	.092	69	43
	regular	$\mathbb{R}^3$	SchNet *	.235	63	41
pseudo-linear	steerable	$SE(3)$	Cormorant	.085	61	34
	steerable	$SE(3)$	L1Net	.088	68	46
	regular	G	LieConv	.084	49	30
	steerable	$SE(3)$	TFN	.223	58	40
pseudo-linear	steerable	$SE(3)$	SE(3)-Tr.	.142	53	35
non-linear	regular	$\mathbb{R}^3 \times S^2 \times \mathbb{R}^+$	DimeNet++ *	**.043**	**32**	24
non-linear	regular	$\mathbb{R}^3 \times S^2 \times \mathbb{R}^+$	SphereNet *	.046	**32**	**23**
non-linear	reguleerable?	$SE(3)$	PaiNN *	.045	45	27
non-linear	regular	$\mathbb{R}^3$	EGNN	.071	48	29
non-linear	steerable	$SE(3)$	SEGNN (Ours)	.060	42	24

Table 2: Performance comparison o
Error (MAE) between model predicti



arXiv:2102.05013v2 [cs.LG] 26 Feb 2021

Spherical Message Passing for 3D Graph Networks

Yi Liu*[1] Limei Wang*[1] Meng Liu[1] Xuan Zhang[1] Bora Oztekin[1] Shuiwang Ji[1]

Abstract

We consider representation learning from 3D graphs in which each node is associated with a spatial position in 3D. This is an under explored area of research, and a principled framework is currently lacking. In this work, we propose a generic framework, known as the 3D graph network (3DGN), to provide a unified interface at different levels of granularity for 3D graphs. Built on 3DGN, we propose the spherical message passing (SMP) as a novel and specific scheme for realizing the 3DGN framework in the spherical coordinate system (SCS). We conduct formal analyses and show that the relative location of each node in 3D graphs is uniquely defined in the SMP scheme. Thus, our SMP represents a complete and accurate architecture for learning from 3D graphs in the SCS. We derive physically-based representations of geometric information and propose the SphereNet for learning representations of 3D graphs. We show that existing 3D deep models can be viewed as special cases of the SphereNet. Experimental results demonstrate that the use of complete and accurate 3D information in 3DGN and SphereNet leads to significant performance improvements in prediction tasks.

1. Introduction

In many real-world studies, structured objects such as molecules and proteins are naturally modeled as graphs (Gori et al., 2005; Wu et al., 2018; Shervashidze et al., 2011; Fout et al., 2017; Liu et al., 2020; Wang et al., 2020). With the advances of deep learning, graph neural networks (GNNs) have been developed for learning from graph data (Kipf & Welling, 2017; Veličković et al., 2018; Xu et al., 2019; Gao & Ji, 2019; Gao et al., 2020; Yuan & Ji, 2020). In Battaglia et al. (2018), existing GNN methods have been unified to the general graph network (GN) framework and can be realized by message passing architectures (Gilmer et al., 2017; Sanchez-Gonzalez et al., 2020). The original GN framework is developed for regular graphs rather than 3D graphs. Generally, a 3D graph contains 3D coordinates for each node given in the Cartesian system along with the graph structure (Liu et al., 2019; Townshend et al., 2019; Axelrod & Gomez-Bombarelli, 2020). Different types of relative 3D information can be derived from 3D graphs, and they can be important in some applications, such as bond lengths and angles in molecular modeling.

In this work, we propose the 3D graph network (3DGN) as a generic framework for 3D graphs. The 3DGN aims at providing a clear interface at different levels of graph granularity such that researchers can easily develop novel methods for 3D graphs. We note that the original Cartesian coordinates in 3D graphs usually cannot serve as direct inputs to computational models, as they contain severely redundant information that may hurt model performance. In addition, they are not invariant to translation and rotation of input graphs. Hence, following message passing neural networks (MPNNs) (Scarselli et al., 2008; Gilmer et al., 2017; Vignac et al., 2020), we further propose a novel message passing scheme, known as the spherical message passing (SMP), for realizing the 3DGN framework. Based on formal analysis in the spherical coordinate system (SCS), we show that the relative location of each node in 3D graphs is uniquely determined in the SMP scheme. Hence, our SMP represents a complete and accurate architecture for realizing the 3DGN in SCS. As the encoded 3D information is the relative positional information such as distances between pairwise nodes, SMP yields predictions that are invariant to translation and rotation of input graphs.

We apply the SMP to real-world problems, where meaningful physical representations are important. By integrating the SMP and physical representations approximating the density functional theory, we develop the spherical message passing neural networks, known as the SphereNet, for 3D graph learning. We show that existing models for 3D graphs, such as SchNet (Schütt et al., 2017) and DimeNet (Klicpera et al., 2020b), are special cases of our SphereNet, as they only encode partial 3D information. We conduct experiments on various types of datasets including QM9, OC20, and MD17. Experimental results show that compared with baseline methods, SphereNet achieves the best performance

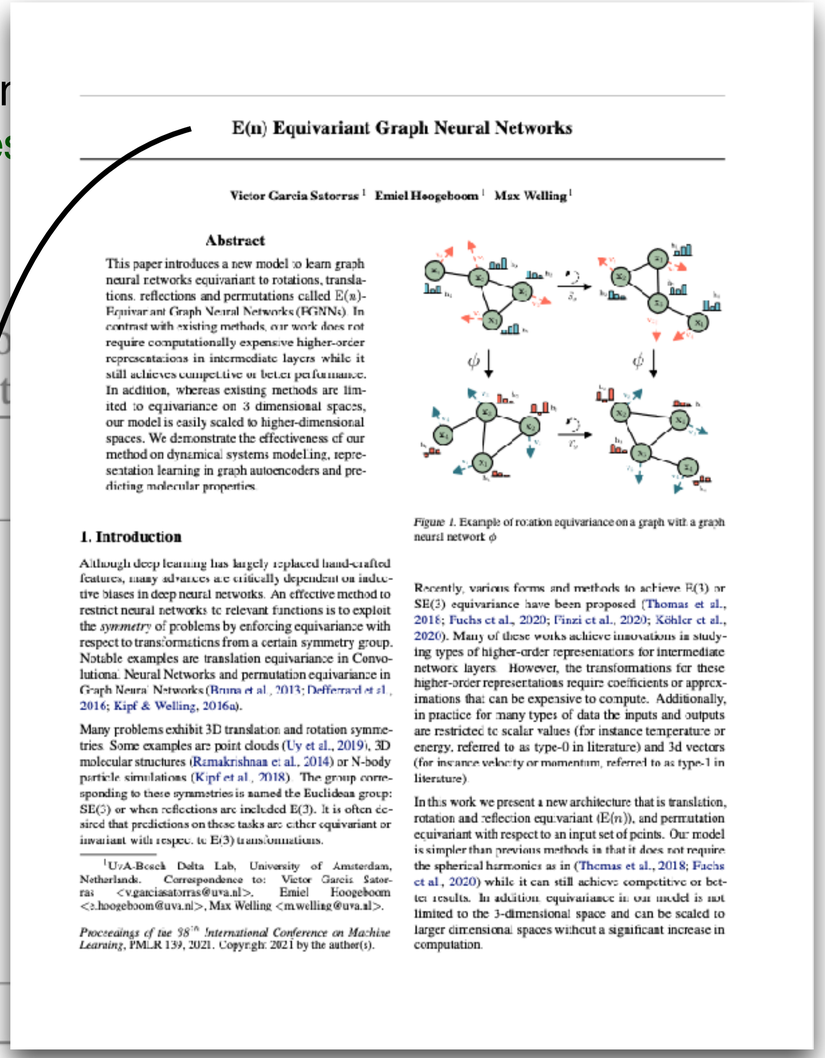
*Equal contribution [1]Department of Computer Science & Engineering, Texas A&M University, TX, USA. Correspondence to: Shuiwang Ji <sji@tamu.edu>.

Linear vs Non-linear & Regular vs Steerable

Recall lecture 1.7:
"Any **equivariant linear layer** between feature maps on homogeneous spaces is a **group convolution**"

			Task	α	$\Delta\varepsilon$	$\varepsilon_{\text{HOMO}}$
			Units	bohr^3	meV	meV
non-linear		no geometry	NMP	.092	69	43
	regular	$\mathbb{R}^3$	SchNet *	.235	63	41
pseudo-linear	steerable	$SE(3)$	Cormorant	.085	61	34
	steerable	$SE(3)$	L1Net	.088	68	46
	regular	G	LieConv	.084	49	30
	steerable	$SE(3)$	TFN	.223	58	40
pseudo-linear	steerable	$SE(3)$	SE(3)-Tr.	.142	53	35
non-linear	regular	$\mathbb{R}^3 \times S^2 \times \mathbb{R}^+$	DimeNet++ *	**.043**	**32**	24
non-linear	regular	$\mathbb{R}^3 \times S^2 \times \mathbb{R}^+$	SphereNet *	.046	**32**	**23**
non-linear	reguleerable?	$SE(3)$	PaiNN *	.045	45	27
non-linear	regular	$\mathbb{R}^3$	EGNN	.071	48	29
non-linear	steerable	$SE(3)$	SEGNN (Ours)	.060	42	24

Table 2: Performance comparison
Error (MAE) between model predict



E(n) Equivariant Graph Neural Networks

Victor Garcia Satorras [1] Emiel Hoogeboom [1] Max Welling [1]

Abstract

This paper introduces a new model to learn graph neural networks equivariant to rotations, translations, reflections and permutations called E(n)-Equivariant Graph Neural Networks (EGNNs). In contrast with existing methods, our work does not require computationally expensive higher-order representations in intermediate layers while it still achieves competitive or better performance. In addition, whereas existing methods are limited to equivariance on 3 dimensional spaces, our model is easily scaled to higher-dimensional spaces. We demonstrate the effectiveness of our method on dynamical systems modelling, representation learning in graph autoencoders and predicting molecular properties.

Figure 1. Example of rotation equivariance on a graph with a graph neural network ϕ

1. Introduction

Although deep learning has largely replaced hand-crafted features, many advances are critically dependent on inductive biases in deep neural networks. An effective method to restrict neural networks to relevant functions is to exploit the *symmetry* of problems by enforcing equivariance with respect to transformations from a certain symmetry group. Notable examples are translation equivariance in Convolutional Neural Networks and permutation equivariance in Graph Neural Networks (Bruna et al., 2013; Defferrard et al., 2016; Kipf & Welling, 2016a).

Many problems exhibit 3D translation and rotation symmetries. Some examples are point clouds (Uy et al., 2019), 3D molecular structures (Ramakrishnan et al., 2014) or N-body particle simulations (Kipf et al., 2018). The group corresponding to these symmetries is named the Euclidean group: SE(3) or when reflections are included E(3). It is often desired that predictions on these tasks are either equivariant or invariant with respect to E(3) transformations.

Recently, various forms and methods to achieve E(3) or SE(3) equivariance have been proposed (Thomas et al., 2018; Fuchs et al., 2020; Finzi et al., 2020; Köhler et al., 2020). Many of these works achieve innovations in studying types of higher-order representations for intermediate network layers. However, the transformations for these higher-order representations require coefficients or approximations that can be expensive to compute. Additionally, in practice for many types of data the inputs and outputs are restricted to scalar values (for instance temperature or energy, referred to as type-0 in literature) and 3d vectors (for instance velocity or momentum, referred to as type-1 in literature).

In this work we present a new architecture that is translation, rotation and reflection equivariant (E(n)), and permutation equivariant with respect to an input set of points. Our model is simpler than previous methods in that it does not require the spherical harmonics as in (Thomas et al., 2018; Fuchs et al., 2020) while it can still achieve competitive or better results. In addition equivariance in our model is not limited to the 3-dimensional space and can be scaled to larger dimensional spaces without a significant increase in computation.

[1]UvA-Bosch Delta Lab, University of Amsterdam, Netherlands. Correspondence to: Victor Garcia Satorras <v.garciasatorras@uva.nl>, Emiel Hoogeboom <e.hoogeboom@uva.nl>, Max Welling <m.welling@uva.nl>.

Proceedings of the 38th International Conference on Machine Learning, PMLR 139, 2021.

Linear vs Non-linear & Regular vs Steerable

Recall lecture 1.7:
"Any **equivariant linear layer** between
feature maps on homogeneous spaces
is a **group convolution**"

Table 2: Performance comparison on the QM9 dataset. N
Error (MAE) between model predictions and ground truth

			Task	α	$\Delta\varepsilon$	$\varepsilon_{\text{HOMO}}$	$\varepsilon_{\text{LUMO}}$	μ	C_ν
			Units	bohr^3	meV	meV	meV	D	cal/mol
non-linear		no geometry	NMP	.092	69	43	38	.030	.040
	regular	$\mathbb{R}^3$	SchNet *	.235	63	41	34	.033	.033
pseudo-linear	steerable	$SE(3)$	Cormorant	.085	61	34	38	.038	.026
	steerable	$SE(3)$	L1Net	.088	68	46	35	.043	.031
	regular	G	LieConv	.084	49	30	25	.032	.038
	steerable	$SE(3)$	TFN	.223	58	40	38	.064	.101
pseudo-linear	steerable	$SE(3)$	SE(3)-Tr.	.142	53	35	33	.051	.054
non-linear	regular	$\mathbb{R}^3 \times S^2 \times \mathbb{R}^+$	DimeNet++ *	**.043**	**32**	24	19	.029	.023
non-linear	regular	$\mathbb{R}^3 \times S^2 \times \mathbb{R}^+$	SphereNet *	.046	**32**	**23**	**18**	.026	**.021**
non-linear	reguleerable?	$SE(3)$	PaiNN *	.045	45	27	20	**.012**	.024
non-linear	regular	$\mathbb{R}^3$	EGNN	.071	48	29	25	.029	.031
non-linear	steerable	$SE(3)$	SEGNN (Ours)	.060	42	24	21	.023	.031

Table 2: Comparison on QM9.