E(2) - Equivariant Steerable CNNs

Gabriele Cesa

September 10, 2020

UNIVERSITY OF AMSTERDAM

MSc ARTIFICIAL INTELLIGENCE

MASTER THESIS

E(2) - Equivariant Steerable CNNs

A GENERAL SOLUTION AND IMPLEMENTATION OF EQUIVARIANCE TO PLANAR ISOMETRIES

---

by

GABRIELE CESA

11887524

September 10, 2020

36 EC

January - August 2019

Supervisor:
MSc MAURICE WEILER

Assessors:
Prof. MAX WELLING

Prof. ERIK J. BEKKERS

INFORMATICS INSTITUTE

QUVA Lab, AMLAB

Gabriele Cesa

E(2) - Equivariant Steerable CNNs

A general solution and implementation of equivariance to planar isometries

January - August 2019

Assessors: Max Welling, Erik J. Bekkers

Supervisor: Maurice Weiler

University of Amsterdam

QUVA Lab, AMLAB

Informatics Institute

Science Park 904

1098 XH and Amsterdam

Abstract

In the latest years, equivariant neural networks have received increasing attention in the deep learning community, as the recent growth in the family of equivariant architectures in the literature demonstrates. Since images are among the most common types of data, this trend is especially notable in the planar - 2-dimensional - setting where rotations and reflections, beyond translations, can be considered. In this thesis, we provide a unified description of $\mathrm{E}(2)$-equivariant Convolutional Neural Networks using the framework of *Steerable CNNs*. In this framework, the feature spaces of CNNs are associated with well-defined transformation laws, defined by group representations. Using some important results in Group Representation Theory, this translates into constraints on the convolution kernels which map between different feature spaces. By reducing these constraints to constraints under irreducible representations of a group, we solve these constraints for arbitrary representations of that group. Therefore, we provide general solutions to the irreducible kernel space constraints for all subgroups of the Euclidean group $\mathrm{E}(2)$. These solutions allow us to not only re-implement a wide range of previously proposed models but also to design new ones and perform a systematic evaluation of them. Finally, by replacing conventional convolution with $\mathrm{E}(2)$-steerable convolution in some of the most popular CNN architectures, we achieve significant improvements on CIFAR-10, CIFAR-100 and STL-10.

Acknowledgement

First of all, I would like to thank Maurice Weiler, my supervisor, for his guidance and for investing so much on me since the very beginning. I have learned a lot from him in the last two years, and I truly enjoyed collaborating with him.

I would also like to thank Max Welling and Erik J. Bekkers for agreeing to be my examiners. I am thankful to all my colleagues at Qualcomm for giving me the time and the opportunity to finish my thesis.

I also want to thank all the special people I have met during my Master's, with whom I shared beautiful moments and overcame many challenges. In particular, a special mention goes to Davide Belli, who accompanied me through this journey since our Bachelor, and to Gabriele Bani and Andrii Skliar, for all the support, the patience and for never making a day at home boring.

My last acknowledgments are dedicated to all my friends and my family in Italy, who, despite the distance, always felt close. Finally, I am especially thankful to my parents and my brother, who have always encouraged and supported me.

Contents

1 Introduction

> *About ten years ago, some computer scientists came by and said they heard we have some really cool problems. They showed that the problems are NP-complete and went away.*
>
> **— Joe Felsenstein**

1.1 Motivation and Problem Statement

In recent years, imposing equivariance to the action of symmetry groups was shown to be a powerful inductive bias in the design of neural network architectures. The equivariance of a network's layers guarantees a desired transformation behavior of convolutional features under corresponding transformations of their inputs, thereby achieving improved generalization capabilities and sample complexities compared to a conventional design. Because of their high practical relevance, a large number of rotation- and reflection- equivariant architectures for planar images have been suggested in the literature. Nevertheless, no systematic study, which reproduces and compares all these works, has been performed yet.

Steerable CNNs, initially introduced in [13] and later extended in [44, 10, 9, 11], represent a first step toward this goal by defining a very general notion of equivariant convolutions on homogeneous spaces. In particular, $\mathrm{E}(2)$-steerable CNNs describe rotation- and reflection-equivariant convolutions on the image plane $\mathbb{R}^2$. The feature spaces of steerable CNNs are interpreted as spaces of *feature fields*, i.e., features associated with a specific transformation law that defines their *field type*. A transformation of the model's input results in another in each feature field of the network according to its own law. Mathematically, the transformations considered form a *group* and a feature field's law is determined by a *group representation*. In order to guarantee the specified transformation laws of feature spaces, each layer in the network needs to be compatible with the type of its own input and output spaces. In the particular case of convolution layers, the kernels are subject to a linear constraint, which depends on the group representations of the spaces. Previous works, including but not limited to [13, 44], have already solved this constraint for specific groups and representations. However, no general solution strategy has been

proposed before. In this work, we present a general method to automatically solve the kernel space constraint defined by any pair of representations by reducing it to much simpler constraints under single *irreducible* representations.

In particular, we solve the kernel constraint defined by arbitrary representations of the orthogonal group $\mathrm{O}(2)$ and its subgroups. Our method enables us to re-interpret a broad family of pre-existing equivariant models - including regular GCNNs [12, 45, 22, 2, 18, 27], classical Steerable CNNs [13], Harmonic Networks [48], gated Harmonic Networks [44], Vector Field Networks [30], (roto-translational) Scattering Transforms [39, 41, 5, 40, 33] - and design entirely new ones in the same general framework. Besides, we can create heterogeneous architectures by combining field types previously used in different networks.

Moreover, we introduce the group restriction operation, which enables one to adjust each layer's equivariance to the symmetries existing at the scale of its field of view. This construction can be helpful, for instance, when working with natural images, which have a typical global orientation but where low-level, local patterns often exhibit rotational symmetry. Therefore, group restriction allows equivariant networks to leverage the emerging symmetries in the data at smaller scales.

Although the theory presented can describe any equivariant steerable CNN, it does not favor any specific choice of group representations or non-linearities. For this reason, we perform an extensive benchmark study, comparing different combinations of equivariance groups, representations and non-linear layers. We experiment on MNIST 12k, rotated MNIST $\mathrm{SO}(2)$ and reflected and rotated MNIST $\mathrm{O}(2)$ to examine the effect of different symmetries in the data. Consequently, we validate our equivariant convolution as a drop-in replacement for the conventional convolution layer on CIFAR10, CIFAR100 and STL-10, and we find significant improvements over the non-equivariant baselines.

Apart from the obvious image processing applications, the methods and the framework we described are relevant for a more general class of problems. Indeed, the strategy we proposed to solve the kernel constraints can generally be used to solve the constraints required by steerable CNNs on homogeneous spaces [10, 9] or by gauge equivariant CNNs on Riemannian manifolds [11]. In particular, when considering signals defined over a 2-dimensional manifold equipped with a subgroup $H \leq \mathrm{O}(2)$ as a structure group, a gauge equivariant CNN enforces precisely the same constraints studied in this work. Thus, our solutions and our findings can be directly applied to e.g. spherical CNNs [8, 11, 26, 19, 34, 23] and many geometric deep learning architectures [35, 32, 4, 3].

1.2 Outline

Chapter 2 introduces all mathematical concepts required to understand the theory of steerable CNNs as well as the notation used throughout this thesis. It starts with a brief introduction to *Group Theory* in Sec. 2.1 and proceeds with more advanced results from *Group Representation Theory* and *Character Theory* in Sec. 2.3 and Sec. 2.6 respectively. Finally, it includes a short overview of the groups here considered together with their representation theory in Sec. 2.7.

Chapter 3 focuses on Steerable CNNs. It first presents the concept of *group convolution* and the more familiar *Group-Convolutional Neural Networks* (GCNNs) in Sec. 3.1. Then, the theory of Euclidean steerable CNNs as described in [44] is briefly reviewed in Sec. 3.2 and Sec. 3.3, where the concepts of feature fields and *steerable convolution* are explained. Sec. 3.4 and Sec. 3.5 present our general strategy to solve the kernel constraints associated to arbitrary representations by decomposing them into their irreducible components. Most of the related works can be seen as specific choices of groups, representations and non-linearities in the steerable CNNs framework. For this reason, we choose to address related works in Sec. 3.6. Sec 3.7 describes the *group restriction* operation as a means to enforce an adaptive level of equivariance in the model and exploit local symmetries in non globally symmetric data.

The implementation details are discussed in Chapter 4. In particular, we describe how steerable convolution can be efficiently implemented in Sec. 4.1. We publish our code as Python library based on *PyTorch* at `https://github.com/QUVA-Lab/e2cnn`. In Sec. 4.4, we give a short overview of the library *e2cnn*.

Finally, Chapter 5 includes an experimental analysis of steerable CNNs. We first benchmark a broad range of equivariant models on different MNIST variations in Sec. 5.1. We then replace conventional convolution with steerable convolution in a popular CNNs architecture and compare the relative improvement in performance on CIFAR10 and CIFAR100 in Sec. 5.5 and on STL-10 in Sec. 5.6.

Mathematical Preliminaries

2

> *In mathematics you don't understand things.*
> *You just get used to them.*
>
> **— John von Neumann**

Here, we introduce the main definitions and concepts from *Group Theory* and *Group Representation Theory*, which are needed to understand the framework equivariant neural networks are built on. All concepts will be accompanied by a number of examples to clarify their meanings and introduce the specific instances we will use in the next chapters.

2.1 Group Theory

Definition 1: Group

A group is a pair $(G, \cdot)$ containing a set G and a binary operation

$$\cdot : G \times G \to G, \ (h, g) \mapsto h \cdot g$$

satisfying the following **group axioms**:

- Associativity: $\forall a, b, c \in G \quad a \cdot (b \cdot c) = (a \cdot b) \cdot c$
- Identity: $\exists e \in G : \forall g \in G \quad g \cdot e = e \cdot g = g$
- Inverse: $\forall g \in G \ \exists g^{-1} \in G : \quad g \cdot g^{-1} = g^{-1} \cdot g = e$

The binary operation $\cdot$ is called the *group law*. It can be proven that the inverse elements g^{-1} of an element g and the identity element e are *unique*.

In order to reduce the notation, it is common to write hg for $h \cdot g$ and to refer to the whole group with G when this is not ambiguous. We can also use the power notation

$$g^n = \underbrace{g \cdot g \cdot g \cdot \ldots \cdot g}_{n \text{ times}}$$

to abbreviate the combination of the element g with itself n times.

Example 1: Real numbers

The set $\mathbb{R}$ together with the binary operation $+$ forms the group $(\mathbb{R}, +)$ of the real numbers. Indeed, the sum is associative and has identity element 0. Moreover, each element x has inverse $-x$ such that $x + (-x) = (-x) + x = 0$.

Example 2: General Linear Group

The set $\mathrm{GL}(\mathbb{R}^n)$ of all invertible real $n \times n$ matrices together with the usual matrix multiplication is a group. The matrix multiplication is associative and the matrix I_n (the $n \times n$ identity matrix) is the identity element. By definition, every matrix in $\mathrm{GL}(\mathbb{R}^n)$ has an inverse.

Definition 2: Order of a Group

The **order** of a group G is the *cardinality* of its set and it is indicated by $|G|$.

Definition 3: Finite Group

A *finite group* is a group with a finite number of elements.

Note that the definition of a group in Def. 1 does not require the operation to be *commutative*. Groups with this property form a special family:

Definition 4: Abelian Group

An **abelian** group $(G, \cdot)$ is a group whose group law additionally satisfies the **commutativity axiom**:

- Commutativity: $\forall a, b \in G \quad a \cdot b = b \cdot a$

Example 3: $\mathbb{Z}/n\mathbb{Z}$

The set $\{0, 1, 2, \ldots, n-2, n-1\}$ together with the sum modulo n

$$+ : (a, b) \mapsto a + b \mod n$$

forms the group $\mathbb{Z}/n\mathbb{Z}$ of integers modulo n. The group has order $n \in \mathbb{N}^+$ and, therefore, it is a *finite* group. Moreover, the addition is *commutative*, hence this group is *abelian*.

Definition 5: Group Homomorphism

Given two groups $(G, \cdot)$ and $(H, *)$, a map $f : G \to H$ is a **group homomorphism** from G to H if

$$\forall a, b \in G \quad f(a \cdot b) = f(a) * f(b) \ .$$

It follows that the function f necessarily maps the identity of G to the identity of H, i.e. $f(e_G) = e_H$. A group homomorphism preserves also the inverse of each elements, i.e. $\forall g \in G \quad f(g^{-1}) = f(g)^{-1}$.

It is worth mentioning two special cases of group homomorphism:

Definition 6: Group Isomorphism

A *group homomorphism* f from G to H is a **group isomorphism** if it is *bijective* (*surjective* and *injective*), i.e. if and only if:

$$\forall h \in H, \quad \exists!\ g \in G : \quad f(g) = h \ .$$

Definition 7: Group Automorphism

A *group homomorphism* f from G to G itself is called a group *endomorphism*.

A *group endomorphism* which is also *bijective* (i.e. an *isomorphism*) is called a **group automorphism**.

Given two groups H and G, if there exists an isomorphisms between them, then the two groups are said *isomorphic*.

Example 4: Cyclic Group

The set of all the complex n-th roots of the unity $\{e^{ik\frac{2\pi}{n}} \mid 0 \leq k < n\}$ forms a group under multiplication. This group is *isomorphic* to the group $\mathbb{Z}/n\mathbb{Z}$ seen in the previous example. Indeed, we can define an homomorphism f between them as

$$f : e^{ik\frac{2\pi}{n}} \mapsto k \ .$$

One can verify this is indeed an isomorphism.

This group is also called **Cyclic Group** of order n, often indicated as C_n. More abstractly, this group can be defined as

$$\mathrm{C}_N = \{e = g^0, g, g^2, \ldots, g^{n-1} | g^i = g^j \iff i \equiv j \mod N\} \ .$$

Note that any element of the group can be identified by a power of the *generating* element g. This group will appear often in the rest of this work.

Groups are especially useful to describe the *symmetries* of a space. This is indeed the case in this work; here, in particular, we are interested in describing the symmetries of signals and functions defined over the plane. The symmetries of a space are mathematically described as the *action* of a group on it.

Definition 8: Group Action and G-Space

Given a group G, a (left) G**-space** X is a set X equipped with a **group action** $G \times X \to X, \ (g, x) \mapsto g.x$, i.e. a map satisfying the following axioms:

- identity: $\forall x \in X \quad e.x = x$
- compatibility: $\forall a, b \in G \ \ \forall x \in X \quad a.(b.x) = (ab).x$

In this case, G is said to *act* on X.

For any group $(G, \cdot)$, its *group law* $\cdot : G \times G \to G$ trivially defines a *group action* of the group on itself ($X = G$).

Example 5

Consider the two-dimensional real space (the *euclidean plane*) $X = \mathbb{R}^2$.
We can define an action of the group $(\mathbb{R}, +)$ on this space as:

$$\forall t \in (\mathbb{R}, +), \forall (x, y) \in X = \mathbb{R}^2 \quad t, (x, y) \mapsto t.(x, y) = (x + t, y)$$

where the group elements act by translating horizontally the points in the plane.

Example 6: Orthogonal Group

One of the main groups we are interested in is the *special orthogonal group* $\mathrm{SO}(2)$ which contains all the planar rotations. The action of a rotation $r_\theta \in \mathrm{SO}(2)$ by an angle θ can be defined as:

$$\forall x \in X = \mathbb{R}^2 \quad r_\theta, x \mapsto r_\theta.x = \psi(\theta)x$$

where $\psi(\theta)$ is the rotation matrix

$$\psi(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}$$

while $\psi(\theta)x$ is the usual matrix-vector product. The group $\mathrm{SO}(2)$ is also the group of all 2×2 *orthogonal* real matrices with positive determinat

$$\mathrm{SO}(2) = \{O \in \mathbb{R}^{2\times 2} \,|\, O^T O = \mathrm{id}_{2\times 2} \text{ and } \det(O) = 1\} .$$

Another group we will consider often is the *orthogonal group* $\mathrm{O}(2)$ which contains all the planar rotations and reflections. The action of a rotation $r_\theta \in \mathrm{O}(2)$ is defined as before for $\mathrm{SO}(2)$. Instead, a reflection f reflects the points around the x-axis by inverting the sign of the first coordinate of a point:

$$\forall x \in X = \mathbb{R}^2 \quad f, x \mapsto f.x = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} x$$

The group $\mathrm{O}(2)$ is also the group of all 2×2 *orthogonal* real matrices

$$\mathrm{O}(2) = \{O \in \mathbb{R}^{2\times 2} \,|\, O^T O = \mathrm{id}_{2\times 2}\} .$$

Note that $\det(O) = \pm 1$ for any $O \in \mathrm{O}(2)$.

For any specific element $x \in X$, one can ask where it is mapped by the action of the group G.

Definition 9: Transitive Group Action

Given a group G with an action on a G-space X, if this action can move any element of X to any of its other elements, i.e.

$$\forall x, y \in X,\ \exists g \in G: \quad y = g.x$$

this action is said to be *transitive*.

Example 7: Translation Group

Consider again the two-dimensional real space (the *euclidean plane*) $X = \mathbb{R}^2$. The action of the group $(\mathbb{R}, +)$ described in Example 5 only translates horizontally the points in the plane. It follows that there is no element of $(\mathbb{R}, +)$ which map a point (x_1, y_1) to another point (x_2, y_2) with $y_1 \neq y_2$. This action is, therefore, not transitive.
On the other hand, we can consider the translation group $(\mathbb{R}^2, +)$ with the following action:

$$\forall t \in (\mathbb{R}^2, +), \forall (x, y) \in X = \mathbb{R}^2 \quad t, (x, y) \mapsto t.(x, y) = (x + t_1, y + t_2)\ .$$

For any pair of points (x_1, y_1) and (x_2, y_2), there is always a translation $t = (x_2 - x_1, y_2 - y_1) \in (\mathbb{R}^2, +)$ which maps the first to the second. The action of $(\mathbb{R}^2, +)$ is, therefore, *transitive* over the space $\mathbb{R}^2$.

Generally, however, an element $x \in X$ can be mapped to only a subset of X. This subset is called the *orbit* of G through x and it is indicated as:

$$G.x = \{g.x \mid g \in G\} \subseteq X\ .$$

2.2 Quotient Groups, Cosets and Group Products

In this section we introduce some useful concepts which will be used later both to describe steerable feature fields and to derive statistics on spaces containing symmetries.

Definition 10: Subgroup

Given a group $(G, \cdot)$, a non-empty subset $H \subseteq G$ is a **subgroup** of G if it forms a group $(H, \cdot)$ under the same group law, restricted to H.
For H to be a subgroup of G, it is necessary and sufficient that the restricted group law and the inverse are *closed* in H, i.e.

- $\forall a, b \in H \quad a \cdot b \in H$
- $\forall h \in H \quad h^{-1} \in H$

This is usually denoted as $H \leq G$.

Any subgroup needs to include the identity element e. Moreover, any group has at leas the *trivial group* and the group itself as subgroups.

Example 8

We have already introduced the group of continuous planar rotations $\mathrm{SO}(2)$ and the cyclic group C_N. Any cyclic group C_N is a finite subgroup of $\mathrm{SO}(2)$ and can be interpreted as the group of N rotations by angles which are integer multiples of $\frac{2\pi}{N}$.
Indeed, in a previous example, we have seen that the cyclic group C_N is *isomorphic* to the group of the N-th roots of the unity under multiplication which, in turn, can be interpreted as rotations in the complex plane $\mathbb{C}$.
The elements of the group C_N can be identified as elements of $\mathrm{SO}(2)$ through the following *inclusion* map:

$$\mathrm{C}_N \to \mathrm{SO}(2), \; g^k \mapsto r_{k\frac{2\pi}{N}} \; .$$

Let G be a group and $H < G$ a subgroup of G.

Definition 11: Cosets

A **left coset** of H in G is $gH = \{gh \mid h \in H\}$ for an element $g \in G$.
Similarly, a **right coset** of H in G is $Hg = \{hg \mid h \in H\}$ for an element $g \in G$.

Intuitively, the left (or right) coset of an element g is the set of all elements of G reachable through the right (or left) action of elements $h \in H < G$. Therefore, a coset contains the *orbit* of H through an element of G.

Cosets form a *partitioning* of the group G, i.e. they are disjoint and their union is equal to the whole group G. It can be shown that all cosets have cardinality equal to the order of H. Indeed, the cosets of H in G define *equivalence classes* over the

elements of G. Additionally, the coset of H through the identity element e is equal to H itself, i.e. $eH = H$, and it is the only coset which is also a group (because the identity only belongs to this coset).

Definition 12: Index

The **index** of H in G is the number of left (or right) *cosets* of H in G. More precisely, it is the cardinality of the set $\{gH \mid g \in G\}$. The index of H in G is denoted $|G : H|$.

In the case both G and H are finite groups, it can be shown (*Lagrange's theorem*) that

$$|G : H| = \frac{|G|}{|H|} > 0 \in \mathbb{N} \ .$$

Example 9

Given a cyclic group C_N of order N, for any positive integer M such that $M|N$ ("M divides N"), i.e. $\exists p \in \mathbb{N} \ : N = pM$, the cyclic group C_M of order M is a subgroup of C_N.
Indeed, the subset $\{g^0, g^p, g^{2p}, \ldots, g^{(M-1)p}\}$ of C_N is closed under multiplication and inverse and it is isomorphic to C_M. A left coset of C_M in C_N through an element $g^k \in \mathrm{C}_N$ looks like

$$g^k \, \mathrm{C}_M = \{g^k g^{tp} = g^{k+tp} \mid 0 \leq t < M\} \ .$$

Therefore, each coset has size M. Note also that if $j \equiv k \mod M$, it holds that $g^k \, \mathrm{C}_M = g^j \, \mathrm{C}_M$. The property of belonging to the same coset defines an *equivalence relation* where each coset is an *equivalence class*. One can verify that the elements $e, g, g^2, \ldots, g^{p-1}$ define different cosets and that these are all the existing cosets. It follows that $|\, \mathrm{C}_N : \mathrm{C}_M \,| = p$ and, therefore, that $|\, \mathrm{C}_N : \mathrm{C}_M \,| = \frac{N}{M}$.

Definition 13: Quotient Space

A **quotient space** (or **cosets space**) is the space of all *left* (or *right*) *cosets* of H in G.
Precisely, the **left quotient space** is denoted as $G/H = \{gH \mid g \in G\}$, while the **right quotient space** is denoted as $H\backslash G = \{Hg \mid g \in G\}$.

Given a *quotient space*, it is natural to define a projection (called **canonical projection**)

$$p : G \to G/H, \ g \mapsto p(g) = gH$$

that maps an element $g \in G$ to its own coset.

Definition 14: Section

A **section** of the quotient space G/H is a map $s : G/H \to G$ such that $s \circ p = id_{G/H}$, i.e.

$$\forall gH \in G/H \;\; p(s(gH)) = gH.$$

In other words, a *section* maps each coset to an element in that coset. Such element can be thought as a **representative** of that coset. Note also that $gH = s(gH)H$. This also enables us to identify any coset gH by its *representative* $s(gH)$.

Note that we can always define an *action* of G on quotient spaces. Indeed, consider a left *quotient space* G/H; we can define the left action of an element $g \in G$ on an element $g'H \in G/H$ as:

$$g(g'H) = (gg')H$$

Because of the properties of groups, this action is *transitive* (Def 9), i.e. any coset can be reached by any other coset with some element $g \in G$. One can also verify that this action is independent from the element g' used to identify the coset $g'H$. Another interesting property of these spaces is that they are always *homogeneous spaces*.

Theorem 1: Homogeneous Space

An **homogeneous space** is a G-*space* with a *transitive* action of G.
Any *homogeneous space* is isomorphic to some *quotient space* G/H with the transitive action of G over it.

Example 10: Sphere

The two dimensional sphere S^2 is isomorphic to the *quotient space* $\mathrm{SO}(3)/\mathrm{SO}(2)$. Fixing an origin $o \in \mathrm{S}^2$, any point $p \in \mathrm{S}^2$ can be reached with a 3D rotation $r_p \in \mathrm{SO}(3)$. A rotation around the axis along the origin $o \in \mathrm{S}^2$ is an element $r_\theta \in \mathrm{SO}(2)$ and does not move the origin o. Therefore, any rotation $r_\theta \in \mathrm{SO}(2)$ around the origin o followed by a rotation $r_p \in \mathrm{SO}(3)$ will move the origin o to the same point $p \in \mathrm{S}^2$. Indeed, any point $p \in \mathrm{S}^2$ in the sphere can be identified with a coset $\{r_p r_\theta \mid r_\theta \in \mathrm{SO}(2)\} \in \mathrm{SO}(3)/\mathrm{SO}(2)$.

A special case occurs when the subgroup H has the following property:

Definition 15: Normal Subgroup

Consider a group G and a subgroup $H < G$. If

$$\forall g \in G \;\; gH = Hg$$

then H is a **normal subgroup** of G. In this case, we write $H \triangleleft G$.

It follows that if $H \triangleleft G$, then

$$G/H \cong H\backslash G$$

This enables us to endow a group structure on the quotient space by identifying the coset eH with the identity and defining the product $*$ between two cosets:

$$\forall gH, g'H \in G/H \;\; (gH) * (g'H) = gHg'H = gg'HH = (gg')H \in G/H$$

Again, it can be shown that this product does not depend on the elements g and g' considered. In other words, any element of gH maps any element of $g'H$ to some element in $gg'H$. One can verify this operation satisfies the *group axioms* in Def. 1.

Definition 16: Quotient Group

If $H \triangleleft G$, then the quotient space G/H is itself a group (**quotient group**)

Example 11: Quotient Group

Consider the group of planar translations and rotations $\mathrm{SE}(2)$. An element of $t_{\boldsymbol{v}} r_\theta \in \mathrm{SE}(2)$ is a rotation $r_\theta \in SO(2)$ by an angle θ followed by a translation $t_{\boldsymbol{v}} \in (\mathbb{R}^2, +)$ by the vector $\boldsymbol{v} \in \mathbb{R}^2$. The group $\mathrm{SO}(2)$ of rotations is a subgroup of $\mathrm{SE}(2)$ while $(\mathbb{R}^2, +)$ is a *normal* subgroup of $\mathrm{SE}(2)$.
Let's first look at the left cosets of $(\mathbb{R}^2, +)$ in $\mathrm{SE}(2)$, i.e. the quotient space:

$$\begin{aligned}\mathrm{SE}(2)/(\mathbb{R}^2,+) &= \{\, t_{\boldsymbol{v}} r_\theta (\mathbb{R}^2,+) = r_\theta t_{\psi(-\theta)\boldsymbol{v}} (\mathbb{R}^2,+) = r_\theta (\mathbb{R}^2,+) \mid t_{\boldsymbol{v}} r_\theta \in \mathrm{SE}(2) \,\} \\ &= \{\, r_\theta (\mathbb{R}^2,+) \mid r_\theta \in \mathrm{SO}(2) \,\} \,.\end{aligned}$$

We notice its elements can be identified with elements of $\mathrm{SO}(2)$ by the map

$$f : \mathrm{SE}(2)/(\mathbb{R}^2,+) \to \mathrm{SO}(2), \quad r_\theta(\mathbb{R}^2,+) \mapsto s(r_\theta(\mathbb{R}^2,+)) = r_\theta \,.$$

Given a coset $r_\alpha(\mathbb{R}^2,+) \in \mathrm{SE}(2)/(\mathbb{R}^2,+)$, we can define an action on another coset $r_\beta(\mathbb{R}^2,+)$ by looking at the action of one of its elements $r_\alpha t_{\boldsymbol{v}}$:

$$(r_\alpha t_{\boldsymbol{v}})\; r_\beta(\mathbb{R}^2,+) = r_{\alpha+\beta} t_{\psi(-\beta)\boldsymbol{v}}\; (\mathbb{R}^2,+) = r_{\alpha+\beta}\; (\mathbb{R}^2,+) \,.$$

The result only depends on r_α but not on the translation $t_{\boldsymbol{v}}$; therefore it is the same for any element in $r_\alpha(\mathbb{R}^2,+)$ chosen. This enables us to define a group action on the quotient space $\mathrm{SE}(2)/(\mathbb{R}^2,+)$ (one can verify its invertibility and associativity). We can also recognize the similarity of this action with the *group law* of $\mathrm{SO}(2)$. Indeed, the quotient $\mathrm{SE}(2)/(\mathbb{R}^2,+)$ is *isomorphic* to the group $\mathrm{SO}(2)$. We can verify the map f is an *isomorphism*. By construction, f is bijective. We now show it is also a group homomorphis:

$$\begin{aligned} f\left(r_\alpha(\mathbb{R}^2,+)\; r_\beta(\mathbb{R}^2,+)\right) &= f\left(r_\alpha\; r_\beta\; (\{t_{\psi(-\beta)\boldsymbol{v}} \mid \boldsymbol{v} \in \mathbb{R}^2\},+)\;\; (\mathbb{R}^2,+)\right) \\ &= f\left(r_\alpha\; r_\beta\; (\mathbb{R}^2,+)\right) \\ &= r_\alpha\; r_\beta \end{aligned}$$

One may ask whether the quotient $\mathrm{SE}(2)/\,\mathrm{SO}(2)$ has the same property.

$$\mathrm{SE}(2)/\,\mathrm{SO}(2) = \{\, t_{\boldsymbol{v}} r_\theta\, \mathrm{SO}(2) = t_{\boldsymbol{v}}\, \mathrm{SO}(2) \mid t_{\boldsymbol{v}} r_\theta \in \mathrm{SE}(2) \,\}$$

Here, we can identify the elements of $\mathrm{SE}(2)/\,\mathrm{SO}(2)$ with the elements of $\mathbb{R}^2$:

$$f : \mathrm{SE}(2)/\,\mathrm{SO}(2) \to \mathbb{R}^2, \quad t_{\boldsymbol{v}}\, \mathrm{SO}(2) \mapsto s(t_{\boldsymbol{v}}\, \mathrm{SO}(2)) = \boldsymbol{v} \,.$$

Note that here we did not write $(\mathbb{R}^2,+)$ but we referred to $\mathbb{R}^2$ only as a set. Indeed, a coset $t_{\boldsymbol{v}}\, \mathrm{SO}(2)$ does not act as a translation $t_{\boldsymbol{v}} \in (\mathbb{R}^2,+)$ on the other cosets. Different elements of the same coset $t_{\boldsymbol{v}}\, \mathrm{SO}(2)$ apply the same translation by $\boldsymbol{v}$ but also rotate the input by different angles, mapping to different cosets. Therefore, $\mathrm{SE}(2)/\,\mathrm{SO}(2)$ does not have a group structure.

So far, given a group we have described its subgroups and how they appear inside the group. Now, given some smaller groups we show how they can be combined to build new larger groups.

Definition 17: Direct Product

Given two groups $(K, *)$ and $(H, +)$, the **direct product** group $(K \times H, \cdot)$ is defined as the *Cartesian product* $K \times H$ of the sets K and H together with the following group law:

$$(k_1, h_1) \cdot (k_2, h_2) = (k_1 * k_2, h_1 + h_2) \, .$$

The direct product between H and K is usually denoted as $K \times H$.
One can easily verify that this construction satisfies the group axioms in Def 1.
This definition can be easily generalized to the direct product of more than two groups.

Given a direct product $K \times H$, the subsets $\{(e_K, h) | h \in H\}$ and $\{(k, e_h) | k \in K\}$ form *normal subgroups* and are isomorphic to H and K, respectively. Any element $(k, h) \in K \times H$ can be uniquely decomposed as the product of an element of K and an element of H, e.g. $(k, h) = (e_K, h) \cdot (k, e_H) = (k, e_H) \cdot (e_K, h)$. Note also that the elements of K commute with the elements H.

Example 12

Any element of the group $(\mathbb{R}^2, +)$ of translations over the real plane can be decomposed into a vertical and a horizontal translation. The group $(\mathbb{R}^2, +)$ is indeed isomorphic to the *direct product* $(\mathbb{R}, +) \times (\mathbb{R}, +)$ of two copies of the group $(\mathbb{R}, +)$ of translations along a line.

The *semi-direct product* is a generalization of the direct product. While the direct product factorizes a group in the product of two normal subgroups whose elements commute with each other, in a semi-direct product only one of the subgroups needs to be normal.

Definition 18: Semi-Direct Product

Given two groups $(N, *)$ and $(H, +)$ and an action $\phi : H \times N \to N$ of H on N, the **semi-direct product** group $N \rtimes_\phi H$ is defined as the *Cartesian product* $N \times H$ equipped with the following binary operation:

$$(n_1, h_1) \cdot (n_2, h_2) = (n_1 * \phi(h_1, n_2), h_1 + h_2) \ .$$

Note that the resulting group depends on the map ϕ and that different maps lead to different groups.
Like in a direct product, any element of a semi-direct product can be *uniquely identified* by a pair of elements of the two subgroups.

The group N is a normal subgroup of the semi-direct product group, but H is not necessarily normal. Moreover, when ϕ is the identity map on N for any $h \in H$, i.e. $\forall\ h \in H,\ n \in N,\ \phi(h, n) = n$, we obtain the previous *direct product*.

Example 13: Special Euclidean group $\mathrm{SE}(2)$

The group $\mathrm{SE}(2)$ is an example of *semi-direct product*. In Ex. 11, we have seen that $\mathrm{SO}(2)$ is a subgroup of $\mathrm{SE}(2)$ while $(\mathbb{R}^2, +)$ is a normal subgroup. Any element of $\mathrm{SE}(2)$ can be identified by a pair $(t_{\boldsymbol{v}}, r_\theta) = t_{\boldsymbol{v}} r_\theta$ with $t_{\boldsymbol{v}} \in (\mathbb{R}^2, +)$ and $r_\theta \in \mathrm{SO}(2)$. The product of two elements is:

$$\begin{aligned}
(t_{\boldsymbol{v}_1},\ r_{\theta_1}) \cdot (t_{\boldsymbol{v}_2},\ r_{\theta_2}) &= t_{\boldsymbol{v}_1} r_{\theta_1} t_{\boldsymbol{v}_2} r_{\theta_2} \\
&= t_{\boldsymbol{v}_1} t_{\psi(\theta_1)\boldsymbol{v}_2} r_{\theta_1} r_{\theta_2} \\
&= (t_{\boldsymbol{v}_1} t_{\psi(\theta_1)\boldsymbol{v}_2},\ r_{\theta_1} r_{\theta_2}) \\
&= (t_{\boldsymbol{v}_1 + \psi(\theta_1)\boldsymbol{v}_2},\ r_{\theta_1 + \theta_2})
\end{aligned}$$

We can identify the action

$$\phi : (\mathbb{R}^2, +) \times \mathrm{SO}(2) \to (\mathbb{R}^2, +), (t_{\boldsymbol{v}_2}, r_{\theta_1}) \mapsto t_{\psi(\theta_1)\boldsymbol{v}_2}$$

Therefore:

$$\mathrm{SE}(2) = (\mathbb{R}^2, +) \rtimes_\phi \mathrm{SO}(2)$$

2.3 Group Representation Theory

In the context of deep learning, data and features are represented as numerical vectors. For this reason, we are particularly interested in G-spaces that are vector spaces and the group actions on them. Therefore, in this section, we will focus on a particular type of group actions, *linear group representations*, which model

abstract algebraic group elements via their action on some vector space, that is, by representing them as linear transformations (matrices) on that space. Group representations are studied in *Representation theory* and form the backbone of Steerable CNNs since they describe the transformation laws of feature spaces. A useful resource that covers most of the representation theory for finite groups is [38].

Definition 19: Linear Group Representation

A **linear group representation** ρ of a group G on a vector space (representation space) V is a *group homomorphism* from G to the general linear group $\mathrm{GL}(V)$, i.e. it is a map

$$\rho : G \to \mathrm{GL}(V) \quad \text{such that} \quad \rho(g_1 g_2) = \rho(g_1)\rho(g_2) \quad \forall g_1, g_2 \in G\,.$$

Recall that, for $V = \mathbb{R}^n$, $\mathrm{GL}(\mathbb{R}^n)$ is the group of all real invertible $n \times n$ matrices, see Example 2.

The requirement to be a homomorphism, i.e. to satisfy $\rho(g_1 g_2) = \rho(g_1)\rho(g_2)$, ensures the compatibility of the matrix multiplication $\rho(g_1)\rho(g_2)$ with the group composition $g_1 g_2$ which is necessary for a well defined group action. We want to emphasize that group representations do not need to model the group *faithfully* (they are homomorphisms but not necessarily isomorphisms).

Example 14: Trivial representation

A simple example is the *trivial representation* $\rho : G \to \mathrm{GL}(\mathbb{R})$ which maps any group element to the identity, i.e. $\forall g \in G \quad \rho(g) = 1$.

Example 15: Rotations matrices

The 2-dimensional rotation matrices

$$\psi : \mathrm{SO}(2) \to \mathrm{GL}(\mathbb{R}^2),\ r_\theta \mapsto \psi(r_\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}$$

are an example of a representation of the group $\mathrm{SO}(2)$ (the group of all planar rotations).

Definition 20: Equivalent representations

Two representations ρ and ρ' on a vector space V are called **equivalent** (or **isomorphic**) iff they are related by a change of basis $Q \in \mathrm{GL}(V)$, i.e.

$$\forall g \in G, \quad \rho'(g) = Q\rho(g)Q^{-1} \, .$$

Equivalent representations behave similarly since their composition is basis independent as seen by

$$\rho'(g_1)\rho'(g_2) = Q\rho(g_1)Q^{-1}Q\rho(g_2)Q^{-1} = Q\rho(g_1)\rho(g_2)Q^{-1} \, .$$

Two representations can be combined by taking their *direct sum*.

Definition 21: Direct sums

Given representations $\rho_1 : G \to \mathrm{GL}(V_1)$ and $\rho_2 : G \to \mathrm{GL}(V_2)$, their **direct sum** $\rho_1 \oplus \rho_2 : G \to \mathrm{GL}(V_1 \oplus V_2)$ is defined as

$$(\rho_1 \oplus \rho_2)(g) = \begin{bmatrix} \rho_1(g) & 0 \\ 0 & \rho_2(g) \end{bmatrix} \, ,$$

i.e. as the *direct sum* of the corresponding matrices. Its action is therefore given by the independent actions of ρ_1 and ρ_2 on the orthogonal subspaces V_1 and V_2 in $V_1 \oplus V_2$.

The direct sum admits an obvious generalization to an arbitrary number of representations ρ_i:

$$\bigoplus_i \rho_i(g) \;=\; \rho_1(g) \oplus \rho_2(g) \oplus \dots$$

The action of a representation might leave a subspace of the representation space invariant. If this is the case, there exists a change of basis to an equivalent representation which is decomposed into the direct sum of two independent representations on the invariant subspace and its orthogonal complement.

Definition 22: Irreducible representations

A representation is called *irreducible* (or **irrep**) if it does not contain any non-trivial invariant subspaces.

For instance, the *trivial* representation in Example 14 is an irreducible representation for any group. We will find more examples in Sec. 2.7.2, where we give an overview of the irreducible representations of all the subgroups of $\mathrm{O}(2)$.

Theorem 2: Decomposition into Irreducible Representations

Any linear representation $\rho : G \to V$ of a compact group G over a field with characteristic zero is a *direct sum* of irreducible representations. Each irrep corresponds to an invariant subspace of the vector space V with respect to the action of ρ.
In particular, any *real* linear representation $\rho : G \to \mathbb{R}^n$ of a compact group G can be decomposed as

$$\rho(g) = Q \left[\bigoplus_{i \in I} \psi_i(g) \right] Q^{-1}$$

where I is an index set specifying the irreducible representations ψ_i contained in ρ and Q is a change of basis.

Therefore, in proofs it is often sufficient to consider irreducible representations. Indeed, we can use this result in Sec. 3.4 to solve the kernel constraint of Steerable CNNs. In addition, irreducible representations are always *indecomposable*, i.e. can not be further decomposed into the direct sum of other representations.

A particularly important representation is the *regular representation*.

Definition 23: Regular Representation

The **regular representation** of a finite group G acts on a vector space $\mathbb{R}^{|G|}$ by permuting its axes. Specifically, associating each axis $\boldsymbol{e}_g$ of $\mathbb{R}^{|G|}$ to an element $g \in G$, the representation of an element $\tilde{g} \in G$ is a permutation matrix which maps $\boldsymbol{e}_g$ to $\boldsymbol{e}_{\tilde{g}g}$.

Example 16: Regular representation of C_4

The regular representation of the group C_4 with elements $\{r_{p\frac{\pi}{2}} | p = 0, \dots, 3\}$ is instantiated by:

g	r_0	$r_{\frac{\pi}{2}}$	r_{π}	$r_{\frac{3\pi}{2}}$
$\rho_{\mathrm{reg}}^{\mathrm{C}_4}(g)$	$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}$

where the p-th axis of $\mathbb{R}^4$ is associated with the element $r_{p\frac{\pi}{2}}$ of C_4.

A vector $\boldsymbol{v} = \sum_g v_g \boldsymbol{e}_g$ in $\mathbb{R}^{|G|}$ can be interpreted as a scalar function

$$\boldsymbol{v} : G \to \mathbb{R},\ g \mapsto v_g$$

on G. Since

$$\rho(h)\ \boldsymbol{v} = \sum_g v_g \boldsymbol{e}_{hg} = \sum_{\tilde{g}} v_{h^{-1}\tilde{g}} \boldsymbol{e}_{\tilde{g}}\ ,$$

the regular representation corresponds to a left translation $[\rho(h)\ \boldsymbol{v}](g) = v_{h^{-1}g}$ of such functions.

A similar representation is the *quotient representation*.

Definition 24: Quotient Representation

The **quotient representation** $\rho_{\text{quot}}^{G/H}$ of G w.r.t. a subgroup H acts on $\mathbb{R}^{|G|/|H|}$ by permuting its axes. Labeling the axes by the cosets gH in the quotient space G/H, it can be defined via its action $\rho_{\text{quot}}^{G/H}(\tilde{g})\boldsymbol{e}_{gH} = \boldsymbol{e}_{\tilde{g}gH}$.

In Appendix D, we give an intuitive explanation of quotient representations in the context of steerable CNNs.

Regular and trivial representations are two special cases of quotient representations which are obtained by choosing $H = \{e\}$ or $H = G$, respectively. Vectors in the representation space $\mathbb{R}^{|G|/|H|}$ can be viewed as scalar functions on the quotient space G/H. For instance, a vector $\boldsymbol{v} = \sum_{gH} v_{gH} \boldsymbol{e}_{gH}$ in $\mathbb{R}^{|G|/|H|}$ can be interpreted as a function

$$\boldsymbol{v} : G/H \to \mathbb{R},\ gH \mapsto v_{gH}$$

on G/H. The action of the quotient representations on v then corresponds to a left translation of these functions on G/H.

Definition 25: Restricted Representation

Any representation $\rho : G \to \mathrm{GL}(\mathbb{R}^n)$ can be uniquely restricted to a representation of a subgroup H of G by restricting its domain of definition:

$$\mathrm{Res}_H^G(\rho) : H \to \mathrm{GL}(\mathbb{R}^n),\ h \mapsto \rho\big|_H(h)$$

2.4 Induced Representation

In this chapter, we focus on *induction*, another method to generate new representations of a group G, in particular from representations of a subgroup H of G. **Induced representations** are of particular relevance for this work as they enable

us to describe mathematically steerable feature fields in convolutional networks. This will be treated in details in Sec. 3.2. To keep the presentation accessible, we first only consider finite groups G and H. We will later extend this concept to more general groups.

Let $\rho : H \to \mathrm{GL}(\mathbb{R}^n)$ be any representation of a subgroup $H < G$. The induced representation $\mathrm{Ind}_H^G(\rho)$ is then defined on the representation space $\mathbb{R}^{n\frac{|G|}{|H|}}$ which can be seen as one copy of $\mathbb{R}^n$ for each of the $|G|/|H|$ cosets gH in the quotient set G/H. In other words, one can define the space were the induced representation acts as $\bigoplus_{gH \in G/H} \mathbb{R}^n \cong \mathbb{R}^{n\frac{|G|}{|H|}}$ and a vector $\boldsymbol{w}$ in this space as:

$$\boldsymbol{w} = \bigoplus_{gH} \boldsymbol{w}_{gH} \ \in \mathbb{R}^{n\frac{|G|}{|H|}}, \tag{2.1}$$

where $\boldsymbol{w}_{gH}$ is some vector in the representation space $\mathbb{R}^n$ of ρ.

For the definition of the induced representation it is more convenient to view this space as the *tensor product* $\mathbb{R}^{|G|/|H|} \otimes \mathbb{R}^n$ and to write a vector $\boldsymbol{w}$ in this space as

$$\boldsymbol{w} = \sum_{gH} \boldsymbol{e}_{gH} \otimes \boldsymbol{w}_{gH} \ \in \mathbb{R}^{n\frac{|G|}{|H|}}, \tag{2.2}$$

where $\boldsymbol{e}_{gH}$ is a basis vector of $\mathbb{R}^{|G|/|H|}$, associated to the coset gH, while $\boldsymbol{w}_{gH} \in \mathbb{R}^n$ is still a vector in the representation space of ρ. The vector $\boldsymbol{e}_{gH} \otimes \boldsymbol{w}_{gH} \in \mathbb{R}^{n\frac{|G|}{|H|}}$ can be interpreted as $\mathrm{vec}\left(\boldsymbol{e}_{gH}\ \boldsymbol{w}_{gH}^T\right)$. If the basis $\{\boldsymbol{e}_{gH}\}_{gH \in G/H}$ is the standard basis of $\mathbb{R}^{\frac{|G|}{|H|}}$ (i.e. $\boldsymbol{e}_{gH,i} = 0$ for any entry i except $\boldsymbol{e}_{gH,i} = 1$ when i is the index of the coset gH), a vector $\boldsymbol{e}_{gH} \otimes \boldsymbol{w}_{gH}$ can be interpreted as the vector $\boldsymbol{w}_{gH} \in \mathbb{R}^n$ padded with zeros to fill the gH-th n-dimensional block of a $n\frac{|G|}{|H|}$-dimensional vector:

$$\boldsymbol{e}_{gH} \otimes \boldsymbol{w}_{gH} = \Big(\ 0 \ \Big| \ \cdots \ \Big| \ 0 \ \Big| \ \underbrace{\boldsymbol{w}_{gH}}_{gH\text{-th block}} \ \Big| \ 0 \ \Big| \ \cdots \ \Big| \ 0 \ \Big)^T$$

The action of $\mathrm{Ind}_H^G(\rho)$ on $\mathbb{R}^{n\frac{|G|}{|H|}}$ can be intuitively understood as

- $i)$ a permutation of the $|G|/|H|$ subspaces (the n-dimensional blocks) associated to the cosets in G/H and
- $ii)$ an action on each of these subspaces via ρ.

To formalize this intuition, note that any element $g \in G$ can be identified by the coset gH to which it belongs and an element $\mathrm{h}(g) \in H$ which specifies its position within

this coset. Hereby $\mathrm{h} : G \to H$ expresses g relative to an arbitrary *representative*[1] $\mathcal{R}(gH) \in G$ of gH and is defined as $\mathrm{h}(g) := \mathcal{R}(gH)^{-1} g$ from which it immediately follows that g is decomposed relative to $\mathcal{R}$ as

$$g = \mathcal{R}(gH)\mathrm{h}(g)\,. \tag{2.3}$$

The action of an element $\tilde{g} \in G$ on a coset $gH \in G/H$ is naturally given by $\tilde{g}gH \in G/H$. This action defines the aforementioned permutation of the n-dimensional subspaces in $\mathbb{R}^{n|G|/|H|}$ by sending $\boldsymbol{e}_{gH}$ in Eq. (2.2) to $\boldsymbol{e}_{\tilde{g}gH}$. Each of the n-dimensional, translated subspaces $\tilde{g}gH$ is in addition transformed by the action of $\rho(\mathrm{h}(\tilde{g}\mathcal{R}(gH)))$. This H-component $\mathrm{h}(\tilde{g}\mathcal{R}(gH)) = \mathcal{R}(\tilde{g}gH)^{-1}\tilde{g}\mathcal{R}(gH)$ of the $\tilde{g}$ action within the cosets accounts for the relative choice of representatives $\mathcal{R}(\tilde{g}gH)$ and $\mathcal{R}(gH)$. Overall, the action of $\mathrm{Ind}_H^G(\rho(\tilde{g}))$ is given by

$$\Big[\mathrm{Ind}_H^G \rho\Big](\tilde{g}) \sum_{gH} \boldsymbol{e}_{gH} \otimes \boldsymbol{w}_{gH} \;:=\; \sum_{gH} \boldsymbol{e}_{\tilde{g}gH} \otimes \rho(\mathrm{h}(\tilde{g}\mathcal{R}(gH)))\, \boldsymbol{w}_{gH}\,, \tag{2.4}$$

which can be visualized as:

$$\mathrm{Ind}_H^G \rho(\tilde{g}) \cdot \begin{bmatrix} \vdots \\ \hline \boldsymbol{w}_{gH} \\ \hline \vdots \\ \hline \vdots \\ \hline \vdots \end{bmatrix} \;=\; \begin{bmatrix} \vdots \\ \hline \vdots \\ \hline \vdots \\ \hline \rho(\mathrm{h}(\tilde{g}\mathcal{R}(gH)))\boldsymbol{w}_{gH} \\ \hline \vdots \end{bmatrix} \begin{matrix} \\ \\ \Big\} gH \\ \\ \Big\} \tilde{g}gH = \tilde{g}\mathcal{R}(gH)H \\ \\ \end{matrix}$$

Both quotient representations and regular representations can be viewed as being induced from trivial representations of a subgroup. Specifically, let $\rho^{\{e\}}_{\text{triv}} : \{e\} \to \mathrm{GL}(\mathbb{R}) = \{(+1)\}$ be the trivial representation of the the trivial subgroup. Then,

$$\mathrm{Ind}_{\{e\}}^G \rho^{\{e\}}_{\text{triv}} : G \to \mathrm{GL}(\mathbb{R}^{|G|})$$

is the *regular representation* which permutes the cosets $g\{e\}$ of $G/\{e\} \cong G$, which are in one to one relation to the group elements themselves. For $\rho^H_{\text{triv}} : H \to \mathrm{GL}(\mathbb{R}) = \{(+1)\}$ being the trivial representation of an arbitrary subgroup H of G, the induced representation

$$\mathrm{Ind}_H^G \rho^H_{\text{triv}} : G \to \mathrm{GL}(\mathbb{R}^{|G|/|H|})$$

[1] Formally, a representative for each coset is chosen by a map $\mathcal{R} : G/H \to G$ such that it projects back to the same coset, i.e. $\mathcal{R}(gH)H = gH$. This map is therefore a *section* of the principal bundle $G \xrightarrow{\pi} G/H$ with fibers isomorphic to H and the projection given by $\pi(g) := gH$.

permutes the cosets gH of H and thus coincides with the *quotient representation* $\rho_{\text{quot}}^{G/H}$.

Note that a vector in $\mathbb{R}^{|G|/|H|} \otimes \mathbb{R}^n$ is in one-to-one correspondence to a function $f : G/H \to \mathbb{R}^n$. The induced representation can therefore equivalently be defined as acting on the space of such functions as[2]

$$[\text{Ind}_H^G \rho(\tilde{g}) \cdot f](gH) = \rho(\text{h}(\tilde{g}\mathcal{R}(\tilde{g}^{-1}gH)))f(\tilde{g}^{-1}gH)\,. \tag{2.5}$$

This definition generalizes to non-finite groups where the quotient space G/H is not necessarily finite anymore.

For the special case of semi-direct product groups $G = N \rtimes H$ it is possible to choose representatives of the cosets gH such that the elements $\text{h}(\tilde{g}\mathcal{R}(g'H)) = \text{h}(\tilde{g})$ become independent of the cosets [10]. This simplifies the action of the induced representation to

$$[\text{Ind}_H^G \rho(\tilde{g}) \cdot f](gH) = \rho(\text{h}(\tilde{g}))\, f(\tilde{g}^{-1}gH) \tag{2.6}$$

All the symmetry groups considered in this work are semi-direct products in the form $G = (\mathbb{R}^2, +) \rtimes H$, with $H \leq \text{O}(2)$ and we always consider features defined over the quotient space $G/H = \mathbb{R}^2$. For this reason, we will only need the simplified formulation in Eq. (2.6) to define Steerable CNNs. This is what we use in Eq. (3.8) for the group $G = \text{E}(2) = (\mathbb{R}^2, +) \rtimes \text{O}(2)$, subgroup $H = \text{O}(2)$ and quotient space $G/H = \text{E}(2)/\text{O}(2) = \mathbb{R}^2$. However, the general formulation of induced representation will be useful to define other representations for the subgroups of $\text{O}(2)$ when designing new models in Sec. 5.1.

2.5 Equivariance and Intertwiners

So far, we have introduced some mathematical concepts which can be used to describe the symmetries of objects and, in particular, of data and features. More precisely, these objects can be formalized as elements of a G-space, whose symmetries are modeled by a group G. In practice, we generally want to build models which process such objects. It is, therefore, useful to study maps between G-spaces.

[2] The rhs. of Eq. (2.4) corresponds to $[\text{Ind}_H^G \rho(\tilde{g}) \cdot f](\tilde{g}gH) = \rho(\text{h}(\tilde{g}\mathcal{R}(gH)))f(gH)$.

Definition 26: Equivariance

Given a group G and two *G-sets* X and Y, a map $f : X \to Y$ is said to be **equivariant** iff

$$\forall x \in X,\ \forall g \in G, \quad f(g.x) = g.f(x)\ .$$

Note that the actions of the group G on the two sets do not need to be the same. A similar concept is that of *invariance*.

Definition 27: Invariance

An **invariant** map is a map $f : X \to Y$ such that:

$$\forall x \in X,\ \forall g \in G, \quad f(g.x) = f(x)\ .$$

Note that invariance is only a special case of *equivariance* where the action of G on the set Y is trivial, i.e.:

$$\forall y \in Y,\ \forall g \in G, \quad g.y = y\ .$$

As argued in Sec. 2.3, we are mostly interested in vectors spaces and linear group actions. The main building blocks in neural networks are learnable linear transformations which map features between different layers.

Definition 28: Intertwiner

Let G be a group and $\rho_1 : G \to \mathrm{GL}(V_1)$ and $\rho_2 : G \to \mathrm{GL}(V_2)$ be two representations, respectively on the vector spaces V_1 and V_2. A linear map W from V_1 to V_2 is an **intertwiner** between ρ_1 and ρ_2 if it is an equivariant map, i.e.:

$$\forall \boldsymbol{v} \in V_1,\ \forall g \in G, \quad W\rho_1(g)\boldsymbol{v} = \rho_2(g)W\boldsymbol{v}$$

and, therefore, iff:

$$\forall g \in G, \quad W\rho_1(g) = \rho_2(g)W\ .$$

For instance, if $V_1 = \mathbb{R}^m$ and $V_2 = \mathbb{R}^n$, $W \in \mathbb{R}^{n\times m}$ is a $n \times m$ real matrix.

The set of all intertwiners between $\rho_1 : G \to \mathrm{GL}(V_1)$ and $\rho_2 : G \to \mathrm{GL}(V_2)$ is denoted as

$$\mathrm{Hom}_G\left(V_1, V_2\right)$$

We can immediately observe that this set is itself a vector space. Indeed, if $W_1, W_2 \in \mathrm{Hom}_G(V_1, V_2)$ are intertwiners between ρ_1 and ρ_2, for any scalar a[3] and any $g \in G$:

$$(W_1 + W_2)\,\rho_1(g) = W_1\rho_1(g) + W_2\rho_1(g) = \rho_2(g)W_1 + \rho_2(g)W_2 = \rho_2(g)\,(W_1 + W_2)$$
$$(aW_1)\,\rho_1(g) = aW_1\rho_1(g) = a\rho_2(g)W_1 = \rho_2(g)\,(aW_1)$$

This means that in order to fully parametrize the space of intertwiners it is sufficient to find a basis for this space.

In the special case the representations considered are *irreducible*, the following important theorem describes the space of existing intertwiners:

Theorem 3: Schur's Representation Lemma

Let $\rho_1 : G \to V_1$ and $\rho_2 : G \to V_2$ be irreducible representations of a group G. Let $A : V_1 \to V_2$ be a linear map such that $\rho_2(g)A = A\rho_1(g),\ \forall g \in G$ (i.e. A is an *intertwiner*). Then, either:

- A is the *null map*, or
- A is an *isomorphism*, i.e. ρ_1 and ρ_2 are *equivalent representations* (Def. 20) and A is the change of basis between ρ_1 and ρ_2

Moreover, in the complex field, a stronger version of Thm. 3 holds:

Theorem 4: Schur's Representation Lemma (Complex Field)

Let $\rho : G \to V$ be a *complex* irreducible representation of a group G. Let $A : V \to V$ be a linear map such that $\rho(g)A = A\rho(g),\ \forall g \in G$. Then, A lives in a 1-dimensional space and is a scalar multiple of the identity, i.e.:

$$\exists \lambda \in \mathbb{C},\ \text{s.t. } A = \lambda I$$

Note that, given two arbitrary complex representations ρ_1 and ρ_2 of G, if one knows their decomposition in terms of complex irreps $\rho_1 = A\left(\bigoplus_{i\in I}\psi_i\right)A^{-1}$ and $\rho_2 = B\left(\bigoplus_{j\in J}\psi_j\right)B^{-1}$, the space $\mathrm{Hom}_G(\rho_1, \rho_2)$ is isomorphic to

$$\mathrm{Hom}_G(\rho_1, \rho_2) \cong \bigoplus_{i\in I}\bigoplus_{j\in J}\mathrm{Hom}_G(\psi_i, \psi_j)$$

and, therefore, can be completely parametrized by taking the union of the 1-dimensional bases spanning each $\mathrm{Hom}_G(\psi_i, \psi_j)$ subspace.

[3] a is a scalar in the *field* over which the vector spaces are defined.

2.6 Character Theory

A powerful tool often used in Representation theory to study and classify the representations of a group is the *character*. We now introduce some important results from *Character Theory* [38] which we will later need in Sec. 3.4.

Definition 29: Character

Let G be a group and V a vector space over a field F. Given a representation $\rho : G \to \mathrm{GL}(V)$, the *character* of ρ is a function

$$\chi_\rho : G \to F \ , \ g \mapsto \chi_\rho(g) := \mathrm{Tr}(\rho(g))$$

which maps a group element g to the *trace* of its representation $\rho(g)$.

Note that the characters of *equivalent* representations (see Def. 20) are the same. Indeed, if $\forall g \in G, \ \rho_1(g) = D\rho_2(g)D^{-1}$, then $\forall g \in G$

$$\chi_{\rho_1}(g) = \mathrm{Tr}(\rho_1(g)) = \mathrm{Tr}(D\rho_2(g)D^{-1}) = \mathrm{Tr}(\rho_2(g)) = \chi_{\rho_2}(g) \tag{2.7}$$

thanks to the properties of the trace. Moreover, it can be shown that any representation of a group G is determined up isomorphism by its character [4], i.e. ρ_1 and ρ_2 are equivalent representations of a group G if and only if $\chi_{\rho_1} = \chi_{\rho_2}$. Another useful property is that the character of the direct sum of two representations is equal to the sum of their characters, i.e. $\forall g \in G$

$$\chi_{\rho_1 \oplus \rho_2}(g) = \mathrm{Tr}((\rho_1 \oplus \rho_2)(g)) = \mathrm{Tr}(\rho_1(g)) + \mathrm{Tr}(\rho_2(g)) = \chi_{\rho_1}(g) + \chi_{\rho_2}(g) \ . \tag{2.8}$$

For simplicity, for the rest of this section we will restrict our consideration to *finite groups*. However, all the results can be easily generalized to *compact groups* by replacing summations with integrals [20].

We can define an *inner product* between characters. Given a finite group G and two characters $\alpha, \beta : G \to \mathbb{C}$, their inner product is defined as:

$$\langle \alpha, \beta \rangle := \frac{1}{|G|} \sum_{g \in G} \alpha(g)\beta(g^{-1}) \tag{2.9}$$

We can now introduce one of the most important theorems in Character theory. We will first state its most common and elegant version, although it is specific for

[4] This is only true for representations over field of *characteristic* 0. This includes the field of real $\mathbb{R}$ and complex $\mathbb{C}$ numbers.

complex representations. We then provide a more general statement which holds for other fields and, in particular, for the real field, which we are interested in.

Theorem 5: Schur's Orthogonality Relation (Complex Field)

Let G be a finite group, ψ_1, ψ_2 two *irreducible* complex representations of G and $\chi_{\rho_1}, \chi_{\rho_2} : G \to \mathbb{C}$ their characters. Then:

$$\langle \chi_{\psi_1}, \chi_{\psi_2} \rangle = \begin{cases} 1 & \text{if } \phi_1 \text{ and } \psi_2 \text{ are } \textit{equivalent} \text{ representations} \\ 0 & \text{otherwise} \end{cases}$$

More generally[5]:

Theorem 6: Schur's Orthogonality Relation (General Field)

Let G be a finite group, ψ_1, ψ_2 two *irreducible* representations of G over a field F [a] and $\chi_{\rho_1}, \chi_{\rho_2} : G \to F$ their characters.
Then:

$$\langle \chi_{\psi_1}, \chi_{\psi_2} \rangle = \begin{cases} d & \text{if } \phi_1 \text{ and } \psi_2 \text{ are } \textit{equivalent} \text{ representations} \\ 0 & \text{otherwise} \end{cases}$$

where $d \in \mathbb{N}^+$ [b].

[a] It is necessary that the *characteristic* of the field F does not divide the order $|G|$ of G. Both $\mathbb{C}$ and $\mathbb{R}$ have characteristic 0 and, therefore, satisfy this condition.
[b] In case F is a *splitting field* for G, e.g. $F = \mathbb{C}$, then $d = 1$.

This result is extremely useful to describe a general representation in terms of its irreducible components. This enables us to easily reduce the study of any representation of a group to the study of its irreducible representations. More precisely, recalling Thm. 2, given a finite group G and the set of its irreps $\{\psi_i : G \to \mathrm{GL}(V_i)\}_i$, any representation $\rho : G \to \mathrm{GL}(V)$ can be expressed a direct sum of of irreps, i.e.:

$$\rho(g) = Q \left[\bigoplus_{i \in I} \psi_i(g) \right] Q^{-1}$$

where I is a set indexing the irreps in $\{\psi_i\}_i$, potentially containing multiple copies of the same irrep. Then, the following result holds:

[5] https://groupprops.subwiki.org/wiki/Character_orthogonality_theorem

Theorem 7: Orthogonal Projection Formula

Given a finite group G and an irreducible *complex* representation ψ, the number of copies (**multiplicity**) m of ψ in a *complex* representation ρ of G is equal to the *inner product* of their characters, i.e. $\langle \chi_\rho, \chi_\psi \rangle = m$.
In a general field F, it holds:

$$\langle \chi_\rho, \chi_\psi \rangle = m \cdot \langle \chi_\psi, \chi_\psi \rangle$$

Let's prove this statement. First, defining $P(g) = \bigoplus_{i \in I} \psi_i(g)$, and therefore $\rho(g) = QP(g)Q^{-1}$, by using the properties in Eq. (2.7) and Eq. (2.8), we obtain:

$$\chi_\rho(g) = \chi_P(g) = \sum_{i \in I} \chi_{\psi_i}(g) \, .$$

We can now use this identity together with Thm. 6 to compute the inner product between the character of ρ and the character of an irrep ψ_j:

$$\begin{aligned}
\langle \chi_\rho, \chi_{\psi_j} \rangle &= \langle \sum_{i \in I} \chi_{\psi_i}, \chi_{\psi_j} \rangle && \text{using the last identity} \\
&= \sum_{i \in I} \langle \chi_{\psi_i}, \chi_{\psi_j} \rangle && \text{using the bilinearity of the inner product} \\
&= \sum_{i \in I} \delta_{ij} d_j && \text{using Thm. 6} \\
&= m_j d_j
\end{aligned}$$

where $\delta_{ij} = 0$ if $i \neq j$ and 1 otherwise, $d_j = \langle \chi_{\psi_j}, \chi_{\psi_j} \rangle$ and m_j is the number of occurrences of the index j in the set I, i.e. the *multiplicity* of ψ_j in ρ.

This provides us with a useful algorithm to compute the multiplicity of each irrep ψ_j in an arbitrary representation ρ of G. Indeed, if G is a finite group, we can numerically compute the characters χ_ρ and χ_{ψ_j} and the inner products $d_j = \langle \chi_{\psi_j}, \chi_{\psi_j} \rangle$ and $\langle \chi_\rho, \chi_{\psi_j} \rangle$. The multiplicity of m_j of ψ_j in ρ will then be their ratio. This will be used in Sec. 3.4 to reduce the kernel constraint of Steerable CNNs in simpler constraints which depend only on irreps.

2.7 Isometries of the Euclidean Plane

In this last section, we briefly introduce some groups of relevance for this work. As we focus on the two-dimensional setting, we consider the general group of all isometries of the plane.

The Euclidean group $\mathrm{E}(2)$ is the group of all isometries of the plane $\mathbb{R}^2$ and consists of translations, rotations and reflections. In computer vision and image analysis, many interesting patterns often appear in arbitrary positions and arbitrary orientations.

	order $\lvert G\rvert$	$G \leq \mathrm{O}(2)$		$(\mathbb{R}^2,+) \rtimes G$
orthogonal	-	$\mathrm{O}(2)$		$\mathrm{E}(2) \cong (\mathbb{R}^2,+) \rtimes \mathrm{O}(2)$
special orthogonal	-	$\mathrm{SO}(2)$		$\mathrm{SE}(2) \cong (\mathbb{R}^2,+) \rtimes \mathrm{SO}(2)$
cyclic	N	C_N		$(\mathbb{R}^2,+) \rtimes \mathrm{C}_N$
reflection	2	$(\{\pm 1\},*)$	$\cong \mathrm{D}_1$	$(\mathbb{R}^2,+) \rtimes (\{\pm 1\},*)$
dihedral	2N	D_N	$\cong \mathrm{C}_N \rtimes (\{\pm 1\},*)$	$(\mathbb{R}^2,+) \rtimes \mathrm{D}_N$

Tab. 2.1.: Overview over the different groups covered in our framework.

For this reason, the Euclidean group models an important factor of variation of image features. In particular, this applies to symmetric images that do not have a preferred global orientation, like satellite imagery or biomedical images. However, even in globally oriented images, the low-level local features present at the small scale can often occur in multiple positions and orientations, making this group still relevant to study.

The Euclidean group $\mathrm{E}(2)$ can be defined as the semi-direct product (see Def. 18) $\mathrm{E}(2) \cong (\mathbb{R}^2,+) \rtimes \mathrm{O}(2)$ of the group of planar translations $(\mathbb{R}^2,+)$ and the group of planar rotations and reflections $\mathrm{O}(2)$. Note that the orthogonal group $\mathrm{O}(2)$ contains all operations which leaves the origin invariant (rotations and reflections). In order to allow for different levels of equivariance and to cover a wide spectrum of related work we consider subgroups of the Euclidean group of the form $G = (\mathbb{R}^2,+) \rtimes H$, defined by subgroups $H \leq \mathrm{O}(2)$. While $\mathrm{O}(2)$ includes all reflections and continuous rotations, its special orthogonal subgroup $\mathrm{SO}(2)$ models rotations only while $(\{\pm 1\}, *)$ describes reflections along a given axis. We further consider the cyclic groups C_N and dihedral groups D_N which are discrete subgroups of $\mathrm{O}(2)$, containing N discrete rotations by multiples of $\frac{2\pi}{N}$ and N discrete rotations and reflections, respectively. Therefore, C_N and D_N have order N and $2N$. For an overview over the groups and their interrelations see Tab. 2.1.

2.7.1 Conventions and Notation

We now shortly introduce some basic conventions we will use throughout this thesis.

As explained in Def. 18 and done in [10], because the groups $G = (\mathbb{R}^2,+) \rtimes H$ are semi-direct products, any element $g \in G$ can be decomposed as a product $g = th$ where $t \in (\mathbb{R}^2,+)$ and $h \in H$.

We denote rotations in $\mathrm{SO}(2)$ and C_N by r_θ with $\theta \in [0, 2\pi)$ and $\theta \in \left\{p\frac{2\pi}{N}\right\}_{p=0}^{N-1}$, respectively. Since $\mathrm{O}(2) \cong \mathrm{SO}(2) \rtimes (\{\pm 1\},*)$ is also a semi-direct product of the the rotations group $\mathrm{SO}(2)$ and the reflections group $(\{\pm 1\},*)$, any element $h \in \mathrm{O}(2)$ can be uniquely identified by $h = r_\theta s \in \mathrm{O}(2)$ where $s \in (\{\pm 1\},*)$ is a reflection

and $r_\theta \in \mathrm{SO}(2)$ a rotation. Similarly, we write $h = r_\theta s \in \mathrm{D}_N$ for the dihedral group $\mathrm{D}_N \cong \mathrm{C}_N \rtimes (\{\pm 1\}, *)$, where $r_\theta \in \mathrm{C}_N$.

Given a point $\boldsymbol{x} \in \mathbb{R}^2$, we denote its polar coordinates with (r, ϕ), where $r \in \mathbb{R}_0^+$ and $\phi \in [0, 2\pi)$. We will occasionally write $\boldsymbol{x}(r, \phi)$ to indicate the point in the plane $\mathbb{R}^2$ associated with the polar coordinates (r, ϕ).

The action of rotations r_θ on $\mathbb{R}^2$ in polar coordinates $\boldsymbol{x}(r, \phi)$ is given by $r_\theta.\boldsymbol{x}(r, \phi) = \boldsymbol{x}(r, r_\theta.\phi) = \boldsymbol{x}(r, \phi + \theta)$. An element $h = r_\theta s$ of $\mathrm{O}(2)$ or D_N acts on $\mathbb{R}^2$ as $h.\boldsymbol{x}(r, \phi) = \boldsymbol{x}(r, r_\theta s.\phi) = \boldsymbol{x}(r, s\phi + \theta)$ where the symbol s denotes both an element of $(\{\pm 1\}, *)$ and a number in $\{\pm 1\}$.

We will also often use the following matrices. We denote a 2×2 orthonormal matrix with positive determinant, i.e. rotation matrix for an angle θ, by:

$$\psi(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}$$

We define the orthonormal matrix with negative determinant corresponding to a reflection along the horizontal axis as:

$$\xi(s = -1) = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

and a general orthonormal matrix with negative determinant, i.e. reflection with respect to the axis 2θ, as:

$$\begin{bmatrix} \cos(\theta) & \sin(\theta) \\ \sin(\theta) & -\cos(\theta) \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

Hence, we can express any orthonormal matrix in the form:

$$\begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & s \end{bmatrix} = \psi(\theta)\xi(s)$$

for some $s \in \{\pm 1\}$ and $\theta \in [0, 2\pi)$, where $\xi(s) = \begin{bmatrix} 1 & 0 \\ 0 & s \end{bmatrix}$.

2.7.2 Irreducible representations of $H \leq \mathrm{O}(2)$

In this section, we give a short overview of the *real* irreducible representations (irreps) of all subgroups H of $\mathrm{O}(2)$. We will use these representations to build H-steerable CNNs in Sec. 3; in particular, in Sec. 3.6, we will use the representation theory of these groups to describe a variety of equivariant neural networks.

Special Orthogonal Group SO(2) $\mathrm{SO}(2)$ irreps decompose into one-dimensional complex irreps of $\mathrm{U}(1)$ on the complex field. However, since we implement our theory with real valued variables, we are not interested in them here. Except for the trivial representation ψ_0, all the other irreps are 2-dimensional rotation matrices with different frequencies $k \in \mathbb{N}$.

- $\psi_0^{\mathrm{SO}(2)}(r_\theta) = 1$

- $\psi_k^{\mathrm{SO}(2)}(r_\theta) = \begin{bmatrix} \cos(k\theta) & -\sin(k\theta) \\ \sin(k\theta) & \cos(k\theta) \end{bmatrix} = \psi(k\theta), \quad k \in \mathbb{N}^+$

Orthogonal Group O(2) $\mathrm{O}(2)$ has two 1-dimensional "degenerate" irreps: the trivial representation $\psi_{0,0}$ and a representation $\psi_{1,0}$ which assigns ± 1 to reflections. The other representations are rotation matrices precomposed with a reflection.

- $\psi_{0,0}^{\mathrm{O}(2)}(r_\theta s) = 1$

- $\psi_{1,0}^{\mathrm{O}(2)}(r_\theta s) = s \quad$ where $s \in (\{\pm 1\}, *)$

- $\psi_{1,k}^{\mathrm{O}(2)}(r_\theta s) = \begin{bmatrix} \cos(k\theta) & -\sin(k\theta) \\ \sin(k\theta) & \cos(k\theta) \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & s \end{bmatrix} = \psi(k\theta)\xi(s),$
  $k \in \mathbb{N}^+$ and $s \in (\{\pm 1\}, *)$

Cyclic Groups C_N The irreps of C_N are identical to the irreps of $\mathrm{SO}(2)$ up to frequency $\lfloor \frac{N}{2} \rfloor$. Due to the discreteness of rotation angles, higher frequencies are aliased and, therefore, isomorphic to these $\lfloor \frac{N}{2} \rfloor$ irreps.

- $\psi_0^{\mathrm{C}_N}(r_\theta) = 1$

- $\psi_k^{\mathrm{C}_N}(r_\theta) = \begin{bmatrix} \cos(k\theta) & -\sin(k\theta) \\ \sin(k\theta) & \cos(k\theta) \end{bmatrix} = \psi(k\theta), \quad \forall k \in \{1, \ldots, \lfloor \frac{N-1}{2} \rfloor\}$

If N is even, there is an additional 1-dimensional irrep corresponding to frequency $\lfloor \frac{N}{2} \rfloor = \frac{N}{2}$:

- $\psi_{N/2}^{\mathrm{C}_N}(r_\theta) = \cos\left(\frac{N}{2}\theta\right) \in \{\pm 1\}$ since $\theta \in \{p\frac{2\pi}{N}\}_{p=0}^{N-1}$

Dihedral Groups D_N Similarly, D_N consists of irreps of $\mathrm{O}(2)$ up to frequency $\lfloor N/2 \rfloor$.

- $\psi_{0,0}^{\mathrm{D}_N}(r_\theta s) = 1$

– $\psi_{1,0}^{\mathrm{D}_N}(r_\theta s) = s \quad$ where $s \in (\{\pm 1\}, *)$

– $\psi_{1,k}^{\mathrm{D}_N}(r_\theta s) = \begin{bmatrix} \cos(k\theta) & -\sin(k\theta) \\ \sin(k\theta) & \cos(k\theta) \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & s \end{bmatrix} = \psi(k\theta)\xi(s),$
$k \in \{1, \ldots, \lfloor \frac{N-1}{2} \rfloor\}$ and $s \in (\{\pm 1\}, *)$

If N is even, there are two 1-dimensional irreps:

– $\psi_{0,N/2}^{\mathrm{D}_N}(r_\theta s) = \cos\left(\frac{N}{2}\theta\right) \in \{\pm 1\}$ since $\theta \in \{p\frac{2\pi}{N}\}_{p=0}^{N-1}$

– $\psi_{1,N/2}^{\mathrm{D}_N}(r_\theta s) = s\cos\left(\frac{N}{2}\theta\right) \in \{\pm 1\}$ since $\theta \in \{p\frac{2\pi}{N}\}_{p=0}^{N-1}$

3 General $\mathrm{E}(2)$ - Equivariant Steerable CNNs

> *I heard reiteration of the following claim: Complex theories do not work; simple algorithms do. I would like to demonstrate that in the area of science a good old principle is valid: Nothing is more practical than a good theory.*
>
> **— Vladimir N Vapnik**

Deep convolutional networks process input images by progressively combining smaller patterns to generate more complex ones in a sequence of feature maps, exploiting the hierarchical pattern in imagery data. With respect to fully-connected MLPs, the linear layers in CNNs convolve their inputs with multiple learned filters. These layers can be interpreted as linear maps constrained with a *convolutional weight sharing*. This particular functional structure guarantees that convolutional networks are translation *equivariant*: a translation of the input corresponds to a similar translation of all feature maps. Conversely, translating the input of an MLP results in unpredictable transformations of the features. Analogously, the features of conventional CNNs do not transform with a consistent behavior when the input is subject to more general transformations like rotations or reflections.

In this thesis, we develop a unified description and implementation of neural networks equivariant to the *isometries* of the plane $\mathbb{R}^2$, i.e., the Euclidean group $\mathrm{E}(2)$. Our work is based on the framework of *steerable CNNs* [13, 44, 10, 9, 11] which defines a general theory covering equivariant CNNs on any homogeneous space and, in particular, on Euclidean spaces $\mathbb{R}^d$.

3.1 Group Convolution Networks (GCNNs)

Before delving into the steerable CNNs framework, we briefly introduce the more classical *group convolutional neural networks* (GCNNs) from [12]. GCNNs directly generalize conventional CNNs by replacing the operation of convolution, usually defined over planar images, with that of *group-convolution*, i.e., a convolution performed over a group. In practice, we will consider *cross-correlation* instead of

convolution, as commonly implemented in the most popular deep learning frameworks. We will also sometimes use group cross-correlation and group convolution interchangeably as often done in the deep learning literature.

Given an input signal $f : \mathbb{R}^2 \to \mathbb{R}$ and a filter $k : \mathbb{R}^2 \to \mathbb{R}$, the classical definition of cross-correlation is:

$$[k \star f](\Delta \boldsymbol{x}) := \int_{\boldsymbol{x}} k(\boldsymbol{x} - \Delta \boldsymbol{x}) f(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x} \tag{3.1}$$

In a real implementation, the domain needs to be discretized and the integral over the continuous plane is replaced by a sum over the pixels in a grid. Note that the output produced by this operation is technically not defined over the input space $\mathbb{R}^2$ where the signals and the filter are defined but, rather, on the "set of all translations" $\{\Delta \boldsymbol{x}\}$. Because this set happens to be isomorphic to the input space $\mathbb{R}^2$, one can still interpret the output signal as defined over the same space of the input.

However, this formulation can be easily generalized to a broader class of transformations by just considering a larger set of transformations, instead of just the translations $\{\Delta \boldsymbol{x}\}$. Here, we will consider sets of transformations which have a group structure.

Definition 30: Group Cross-Correlation

Given a signal $f : B \to \mathbb{R}$ and a filter $k : B \to \mathbb{R}$ defined over an input space B with an action of a group G, a group cross-correlation is defined as:

$$[k \star_G f](g) := \int_{x \in B} k(g^{-1}x) f(x) dx \tag{3.2}$$

As the classical cross-correlation is equivariant to translations of the input signal, one can verify the operation in Eq. (3.2) is *equivariant* to the action of G on the input f.

Theorem 8: Equivariance of group cross-correlation

Let G be a group with an action on a space B. Given a function $f : B \to \mathbb{R}$, an action of $g \in G$ on f can be defined as:

$$[g.f](x) := f(g^{-1}x)$$

Then, given two functions $k : B \to \mathbb{R}$ and $f : B \to \mathbb{R}$, it holds that:

$$\forall h \in G \quad k \star_G h.f = h.[k \star_G f] \tag{3.3}$$

This can be quickly verified as:

$$\begin{aligned}
[k \star_G h.f](g) &= \int_{g' \in G} k(g^{-1}g') f(h^{-1}g') dg' \\
&= \int_{g'' \in G} k(g^{-1}hg'') f(g'') dg'' \\
&= \int_{g'' \in G} k((h^{-1}g)^{-1}g'') f(g'') dg'' \\
&= [k \star_G f](h^{-1}g) = [h.[k \star_G f]](g)
\end{aligned}$$

where we have used the substitution $g'' = h^{-1}g'$ (and the left-invariance of the Haar measure).

Note that the output of a group convolution is not a signal over the input space B anymore, but rather a function over the group G. Indeed, the intermediate features of a GCNN usually look different from its input. While an input is defined as a signal over a space endowed with a G-action, a feature map at layer l is a multi-channel signal $f_l : G \to \mathbb{R}^{c_l}$ over the group. Because the group G intrinsically has a group action over its own elements (the *group law* itself), Eq. (3.2) can also be used to define the convolution in the intermediate layers of the network. Finally, to achieve group *invariance*, a common approach is to aggregate a feature map over the group, e.g., through averaging or max pooling. Because the first group-convolution layer maps a signal on a G-space B (e.g., an image) to a signal on G, it is often referred to as *uplifting layer* in the literature.

$\mathrm{E}(2)$ **GCNN** In particular, here we are interested in the case where the input base space is the plane $B = \mathbb{R}^2$ and the group G is a subgroup of its isometries, i.e., $G \leq \mathrm{E}(2)$. Moreover, we will only consider those groups which contain the planar translations $(\mathbb{R}^2, +)$ (see example 7), as translation equivariance has proved extremely useful in most image processing applications. Hence, we assume the group takes the form of a semi-direct product $G = (\mathbb{R}^2, +) \rtimes H$ (see Def. 18) with $H \leq \mathrm{O}(2)$. For instance, when $H = \mathrm{SO}(2)$, we have the group $G = (\mathbb{R}^2, +) \rtimes \mathrm{SO}(2) = \mathrm{SE}(2)$ of planar translations and rotations as seen in example 13. [12] first introduced GCNNs and considered discrete groups $H \leq \mathrm{D}_4 \leq \mathrm{O}(2)$ containing rotations by multiples of $\frac{\pi}{2}$ and reflections. Because these are perfect symmetries of the discretized grid, group convolution as in Eq. (3.2) can be implemented perfectly. Note that Eq. (3.2) involves transformed filters like $g.k = k(g^{-1} \cdot)$. As a result, the use of rotations by smaller angles requires some form of interpolations of the filters. [45] implements GCNNs equivariant to multiples of $\frac{2\pi}{N}$, with $N > 4$, defining filters in terms of a steerable basis, enabling the analytical rotation of the filters before they are sampled on the grid. Instead, [2] builds a similar architecture by learning filters in a single orientation and, then, rotating them through bilinear interpolation.

3.1.1 Implementation and steps towards Steerable CNNs

If H is a finite group with $|H| \in \mathbb{N}^+$ elements, group convolution can be efficiently implemented exploiting the *conv2D* function often provided in deep learning frameworks. Assume a single input and output channel for simplicity, and let $f : G \to \mathbb{R}$ be a feature map. Recall that any element $g \in G = (\mathbb{R}^2, +) \rtimes H$ can be uniquely identified by a pair $(t_{\boldsymbol{x}}, h)$ such that $g = t_{\boldsymbol{x}} h$, where $t_{\boldsymbol{x}} \in (\mathbb{R}^2, +)$ (and, therefore, $\boldsymbol{x} \in \mathbb{R}^2$) and $h \in H$ (see Def. 18). Then, we can "reshape" f to a 2-arguments map $f(\boldsymbol{x}, h) := f(t_{\boldsymbol{x}} h)$. By defining $g = t_{\boldsymbol{y}} h$ and $g' = t_{\boldsymbol{x}} h'$ and noting that

$$g^{-1} g' = h^{-1} t_{-\boldsymbol{y}} t_{\boldsymbol{x}} h' = t_{h^{-1}(\boldsymbol{x}-\boldsymbol{y})} h^{-1} h' \quad ,$$

Eq. (3.2) becomes:

$$[k \star_G f](\boldsymbol{y}, h) = \sum_{h' \in H} \int_{\boldsymbol{x} \in \mathbb{R}^2} k(h^{-1}(\boldsymbol{x} - \boldsymbol{y}), h^{-1} h') f(\boldsymbol{x}, h') d\boldsymbol{x} \tag{3.4}$$

Note that, for a fixed h' and $h = e$ the identity, this equation is equivalent to a conventional convolution as in Eq. (3.1) with a filter

$$k_{h'} := k(\,\cdot\,, h') : \mathbb{R}^2 \to \mathbb{R}$$

over the feature map

$$f_{h'} := f(\,\cdot\,, h') : \mathbb{R}^2 \to \mathbb{R} \quad .$$

If $h \neq e$, the convolution is performed with a filter transformed (e.g. rotated) by h:

$$h.k_{h'} := k(h^{-1}\,\cdot\,, h^{-1} h') : \mathbb{R}^2 \to \mathbb{R} \quad .$$

Then, Eq. (3.4) can be visually represented as:

$$\underbrace{\begin{bmatrix} [k \star f]_e \\ [k \star f]_{h_1} \\ [k \star f]_{h_2} \\ \vdots \\ [k \star f]_{h_{|H|-1}} \end{bmatrix}}_{\boldsymbol{k} \star \boldsymbol{f} : \mathbb{R}^2 \to \mathbb{R}^{|H|}} = \underbrace{\begin{bmatrix} k_e & k_{h_1} & \dots & k_{h_{|H|-1}} \\ h_1.k_e & h_1.k_{h_1} & \dots & h_1.k_{h_{|H|-1}} \\ \vdots & \vdots & \ddots & \vdots \\ h_{|H|-1}.k_e & h_{|H|-1}.k_{h_1} & \dots & h_{|H|-1}.k_{h_{|H|-1}} \end{bmatrix}}_{\boldsymbol{k} : \mathbb{R}^2 \to \mathbb{R}^{|H| \times |H|}} \star \underbrace{\begin{bmatrix} f_e \\ f_{h_1} \\ f_{h_2} \\ \vdots \\ f_{h_{|H|-1}} \end{bmatrix}}_{\boldsymbol{f} : \mathbb{R}^2 \to \mathbb{R}^{|H|}} \tag{3.5}$$

This can be implemented as $|H| \times |H|$ conventional convolutions (one for each entry of $\boldsymbol{k}$) but using only $|H|$ different filters $(k_e, \dots, k_{h_{|H|-1}})$. In practice, the $|H|$ input signals $f_e, f_{h_1}, \dots, f_{h_{|H|-1}}$ can be stacked into an $|H|$-channels single input $\boldsymbol{f} : \mathbb{R}^2 \to \mathbb{R}^{|H|}$; similarly, all the $|H|^2$ filters $(k_e, \dots, k_{h_{|H|-1}}$ and their transformed copies) can be combined in a unique kernel $\boldsymbol{k} : \mathbb{R}^2 \to \mathbb{R}^{|H| \times |H|}$. Then, group convolution is implemented as a conventional convolution with input and output

feature maps having $|H|$ channels. However, in contrast to conventional convolution, the filter $\boldsymbol{k}$ is not completely free; instead, only the first row is freely learnable while the other rows contain transformed versions of it (by the action of a different group element $h \in H$).

At last, let's study the action of a group element $g = t_{\boldsymbol{y}}h \in G$ on $\boldsymbol{f}$. We first note that

$$[g.f](\boldsymbol{x}, h') = [(t_{\boldsymbol{y}}h).f](\boldsymbol{x}, h') = f((t_{\boldsymbol{y}}h)^{-1}\boldsymbol{x}, h^{-1}h') = f(h^{-1}(\boldsymbol{x} - \boldsymbol{y}), h^{-1}h')$$

In other words, $g = t_{\boldsymbol{y}}h$ first transforms each channel independently $f_{h'}$ by g as $[g.f_{h'}](\boldsymbol{x}) = f_{h'}(g^{-1}\boldsymbol{x})$, i.e. it shifts it by $\boldsymbol{y}$ and transforms it by h. Then, g permutes the channels, moving the channel $h^{-1}h'$ to h' or, equivalently, moving h' to hh'. The first transformation is the same for all channels $f_{h'}$ of $\boldsymbol{f}$. Hence, we can write

$$[g.\boldsymbol{f}](\boldsymbol{x}) = \pi_h \boldsymbol{f}(g^{-1}\boldsymbol{x}) \tag{3.6}$$

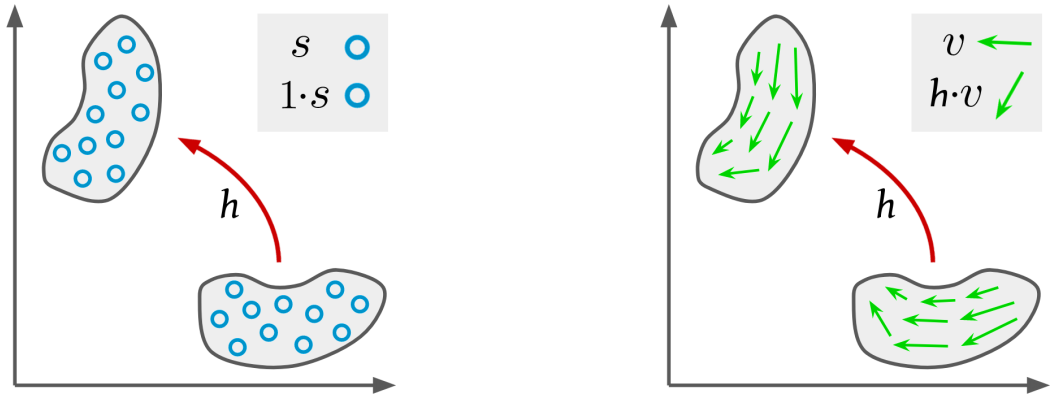
where π_h is a permutation of the $|H|$ channels of $\boldsymbol{f}$ which sends channel h' to hh'. We will find a similar formulation in the next sections when describing steerable CNNs. Indeed, we will soon see that GCNNs are just a special case of them.

3.2 Feature Fields

In Sec. 3.1.1, we have seen how a GCNN equivariant to $G = (\mathbb{R}^2, +) \rtimes H$ can be implemented as a conventional CNN with structured filters. In particular, Eq. (3.5) and (3.6) show that the action of G on a feature map of the CNN endows its channels with additional structure, associating each of the $|H|$ channels to a different element $h \in H$. Steerable CNNs further generalize this concept to *steerable feature fields*. A feature field $f : \mathbb{R}^2 \to \mathbb{R}^c$ associates a c-dimensional feature vector $f(x) \in \mathbb{R}^c$ to each point x in the base space $B = \mathbb{R}^2$. With respect to conventional CNNs, steerable feature fields are paired with transformation laws which define how they are transformed by the action of G. At layer l, the feature field $f_l : \mathbb{R}^2 \to \mathbb{R}^{c_l}$ transforms under $g = th \in G = (\mathbb{R}^2, +) \rtimes H$ as

$$[g.f_l](\boldsymbol{x}) = \rho_l(h) f_l(g^{-1}\boldsymbol{x}) \tag{3.7}$$

where $\rho_l : H \to \mathrm{GL}(\mathbb{R}^{c_l})$ associates an invertible $c_l \times c_l$ matrix to each element $h \in H$, specifying how the c channels of each feature vector $f(x)$ are mixed. More precisely, ρ_l is a *group representation* of H, see Sec. 2.3. Recall that a representation ρ needs to satisfy $\rho(h\tilde{h}) = \rho(h)\rho(\tilde{h})$ and models the group law $h \cdot \tilde{h}$ as the matrix multiplication



scalar field $\rho(h) = 1$ vector field $\rho(h) = h$

Fig. 3.1.: Transformation behavior of ρ-fields under a rotation by h. Each element $f(\boldsymbol{x})$ is being moved to a new position $h\boldsymbol{x}$ and additionally undergoes a change of orientation prescribed by $\rho(h)$.

of $\rho(h)$ and $\rho(\tilde{h})$. The reader may have already noticed the similarity between Eq. (3.6) and Eq. (3.7): the permutation matrices π_h in Eq. (3.6) corresponds to a specific choice of ρ_l, the *regular representation* of H (Def. 23). We will often refer to this kind of field as **regular field**.

It is worth now discussing two other important examples of feature fields. The first one is **scalar fields** $s : \mathbb{R}^2 \to \mathbb{R}$. For instance, scalar fields describe gray-scale images or temperature and pressure fields. An element $g = t_{\boldsymbol{y}} h \in G \leq \mathrm{E}(2)$ acts on a scalar field by moving each point $\boldsymbol{x}' \in \mathbb{R}^2$ to a new position $\boldsymbol{x} = g\boldsymbol{x}' = h\boldsymbol{x}' + \boldsymbol{y}$, i.e.:

$$[g.s](\boldsymbol{x}) := s\left(g^{-1}\boldsymbol{x}\right) = s\left(h^{-1}(\boldsymbol{x} - \boldsymbol{y})\right)$$

see Fig. 3.1, left. The second example is that of **vector fields** $v : \mathbb{R}^2 \to \mathbb{R}^2$. Examples of vector fields are optical flows or the gradient of images. These fields transform as

$$[g.v](\boldsymbol{x}) := h \cdot v\left(g^{-1}\boldsymbol{x}\right) = h \cdot v\left(h^{-1}(\boldsymbol{x} - \boldsymbol{y})\right) \quad .$$

In contrast to the case of scalar fields, a vector $v(\boldsymbol{x}')$ is not only moved to a new position $\boldsymbol{x} = g\boldsymbol{x}'$ but it also *changes orientation* through the action of $h \in H$; see Fig. 3.1, right.

Indeed, when $H \leq \mathrm{O}(2)$, a transformation law enriches the features with a notion of *orientation*. This is similar to the concept of *capsules* from [37]. Indeed, the entries of a feature vector $f(\boldsymbol{x})$ can be interpreted as the coefficients which describe a coordinate-independent geometric feature with respect to a particular reference frame (i.e., image orientation).

In general, the transformation law of a feature field $f : \mathbb{R}^2 \to \mathbb{R}^c$ is completely determined by its **feature type** ρ, where $\rho : H \mapsto \mathrm{GL}(\mathbb{R}^c)$ is a c-dimensional representation of H. We will refer to a field of type ρ as a **ρ-field**. Formally, the transformation law in Eq. 3.7 of a ρ-field f is the **induced representation**[1] $\mathrm{Ind}_H^G \rho$ of $G = (\mathbb{R}^2, +) \rtimes H$ (see Sec. 2.4):

$$[g.f](\boldsymbol{x}) := \Big(\Big[\mathrm{Ind}_H^G \rho\Big](t_{\boldsymbol{y}} h) \cdot f\Big)(\boldsymbol{x}) = \rho(h) \cdot f\left(h^{-1}(\boldsymbol{x} - \boldsymbol{y})\right) \quad . \tag{3.8}$$

Note that, here, we used the simpler definition of the induced representation for semi-direct products as in Eq. (2.6). As in the previous examples, an induced representation transforms a feature field by moving the feature vectors from $h^{-1}(\boldsymbol{x} - \boldsymbol{y})$ to a new position $\boldsymbol{x}$ and mixing the channels with $\rho(h)$. It turns out that scalar fields correspond to the **trivial representation** $\rho(h) = 1 \; \forall h \in H$ (example 14) which reflects that scalar values do not change when being moved. Analogously, vector fields are associated with the standard representation $\rho(h) = h$, where the elements of $H = \mathrm{SO}(2)$ or $H = \mathrm{O}(2)$ are interpreted as 2×2 matrices as in example 15.

In a specific layer, like the features of conventional CNN contain many channels, the features of steerable CNNs can be composed of many feature fields $f_i : \mathbb{R}^2 \to \mathbb{R}^{c_i}$, each of its own *type* $\rho_i : H \to \mathrm{GL}(\mathbb{R}^{c_i})$. The individual feature fields $\{f_i\}_i$ are stacked in a single feature field $f = \bigoplus_i f_i$ which, then, transforms under the **direct sum** $\rho = \bigoplus_i \rho_i$ of the individual representations (see Def. 21). Note that the block-diagonal structure of the direct sum ensures that individual feature fields transform independently from each other. As a simple example, consider an RGB image $f : \mathbb{R}^2 \to \mathbb{R}^3$. Because colors do not change and do not mix when the image is rotated, they can be interpreted as three independent scalar fields. The stacked field then is of type $\bigoplus_{i=1}^3 1 = \mathrm{id}_{3\times 3}$, i.e. the direct sum of three trivial representations.

In a real application, the input and output types of the model are intrinsically determined by the task. However, like the number of channels in the hidden layers of a conventional network is a design choice, one still needs to choose the types ρ_i of each intermediate feature field as hyper-parameters. In Sec. 3.6, we elaborate further on different possible representations while in Sec. 5.1 we perform an extensive experimental comparison of them.

3.3 Steerable Convolution

In Sec. 3.2, we defined a way to describe the transformation laws of steerable features. Now, we want to ensure that a transformation of the features in a layer

[1] Induced representations are the most general transformation laws compatible with convolutions [10, 9].

results in a similar transformation in the features of the following layers. While the transformation law of the input (e.g., the input image) is given by the task and, therefore, guaranteed to describe the input features, a general layer or neural network does not ensure its output transforms as desired. We have already seen in Thm. 8 that group convolution is equivariant and, therefore, guarantees that the action of the group G on the input commutes with the convolution layers of the network. When we wrote group convolution in terms of a conventional convolution in Eq. (3.5), we observed that it corresponds to a convolution with structured filters. While the structure in Eq. (3.5) is specific for input and output *regular fields*, in this section we will generalize this result for any pair of *field types*.

To ensure steerable feature spaces transform as defined by their field types, each layer of a steerable CNN needs to commute with the group's action, i.e., it needs to be group equivariant. As proven for Euclidean groups in [44]:

Theorem 9: Kernel Constraint: H-Steerability

The most general *equivariant linear map* between steerable feature spaces of type ρ_{in} and ρ_{out} [a], is given by *convolutions* with H-*steerable kernels*

$$k : \mathbb{R}^2 \to \mathbb{R}^{c_{\text{out}} \times c_{\text{in}}}$$

satisfying a kernel constraint

$$k(h\boldsymbol{x}) \;=\; \rho_{\text{out}}(h)\, k(\boldsymbol{x})\, \rho_{\text{in}}(h^{-1}) \quad \forall h \in H,\ \boldsymbol{x} \in \mathbb{R}^2 \tag{3.9}$$

[a] i.e. transforming under $\text{Ind}_H^G \rho_{\text{in}}$ and $\text{Ind}_H^G \rho_{\text{out}}$

A more general statement, valid for arbitrary homogeneous spaces, has been derived in [10, 9].

Note 1: Discretized Kernel

Note that $k : \mathbb{R}^2 \to \mathbb{R}^{c_{\text{out}} \times c_{\text{in}}}$ associates a matrix of shape $(c_{\text{out}}, c_{\text{in}})$ to each point $\boldsymbol{x} \in \mathbb{R}^2$ in the plane. The kernel k can then *discretized* on a $X \times Y$ grid, generating the tensor of shape $(c_{\text{out}}, c_{\text{in}}, X, Y)$ commonly used in most deep learning frameworks.

In other words, the condition in Thm. 9 relates the kernel evaluated on transformed coordinates $h\boldsymbol{x}$ to the kernel on non-transformed coordinates $\boldsymbol{x}$ and, therefore, the responses associated to transformed inputs. We can prove that a convolution kernel of this form guarantees that its output transforms according to $\text{Ind}_H^G \rho_{\text{out}}$ when its input is transformed by $\text{Ind}_H^G \rho_{\text{in}}$. Here, we will only show that the H-steerability (Thm. (9)) of convolution kernels is *sufficient* for equivariance. A complete proof

that shows H-steerability is not only sufficient but also *necessary* can be found in [44].

Assume two feature fields $f_{\text{in}} : \mathbb{R}^2 \to \mathbb{R}^{c_{\text{in}}}$ of type ρ_{in} and $f_{\text{out}} : \mathbb{R}^2 \to \mathbb{R}^{c_{\text{out}}}$ of type ρ_{out} to be given. The group $G = (\mathbb{R}^2, +) \rtimes H \leq \mathrm{E}(2)$ acts on them as in Eq. (3.8):

$$
\begin{aligned}
[g.f_{\text{in}}] &:= \left(\left[\mathrm{Ind}_H^G \rho_{\text{in}}\right](t_{\boldsymbol{y}}h)\ f_{\text{in}}\right)(\boldsymbol{x}) = \rho_{\text{in}}(h)\, f_{\text{in}}\left(h^{-1}(\boldsymbol{x}-\boldsymbol{y})\right) \\
[g.f_{\text{out}}] &:= \left(\left[\mathrm{Ind}_H^G \rho_{\text{out}}\right](t_{\boldsymbol{y}}h)\ f_{\text{out}}\right)(\boldsymbol{x}) = \rho_{\text{out}}(h) f_{\text{out}}\left(h^{-1}(\boldsymbol{x}-\boldsymbol{y})\right)\,.
\end{aligned}
$$

The convolution (actually, correlation) of a feature field with an H-steerable kernel $k : \mathbb{R}^2 \to \mathbb{R}^{c_{\text{out}} \times c_{\text{in}}}$ is defined like the conventional one in Eq. (3.1):

$$
f_{\text{out}}(\boldsymbol{x}_1) := [k \star f_{\text{in}}](\boldsymbol{x}_1) = \int_{\mathbb{R}^2} k(\boldsymbol{x}_0 - \boldsymbol{x}_1) f_{\text{in}}(\boldsymbol{x}_0)\, \mathrm{d}\boldsymbol{x}_0\,.
$$

The response of convolution when the input f_{in} is transformed by $g = t_{\boldsymbol{y}}h \in G$ is:

$$
\begin{aligned}
[k \star g.f_{\text{in}}](\boldsymbol{x}_1) &= \int_{\mathbb{R}^2} k(\boldsymbol{x}_0 - \boldsymbol{x}_1)[(t_{\boldsymbol{y}}h).f_{\text{in}}](\boldsymbol{x}_0)\, \mathrm{d}\boldsymbol{x}_0 \\
&= \int_{\mathbb{R}^2} k(\boldsymbol{x}_0 - \boldsymbol{x}_1)\rho_{\text{in}}(h)\, f_{\text{in}}(h^{-1}(\boldsymbol{x}_0 - \boldsymbol{y}))\, \mathrm{d}\boldsymbol{x}_0 \\
&\overset{(1)}{=} \int_{\mathbb{R}^2} \rho_{\text{out}}(h) k(h^{-1}(\boldsymbol{x}_0 - \boldsymbol{x}_1))\rho_{\text{in}}(h^{-1})\rho_{\text{in}}(h)\, f_{\text{in}}(h^{-1}(\boldsymbol{x}_0 - \boldsymbol{y}))\, \mathrm{d}\boldsymbol{x}_0 \\
&= \rho_{\text{out}}(h) \int_{\mathbb{R}^2} k(h^{-1}(\boldsymbol{x}_0 - \boldsymbol{y}) - h^{-1}(\boldsymbol{x}_1 - \boldsymbol{y})) f_{\text{in}}(h^{-1}(\boldsymbol{x}_0 - \boldsymbol{y}))\, \mathrm{d}\boldsymbol{x}_0 \\
&\overset{(2)}{=} \rho_{\text{out}}(h) \int_{\mathbb{R}^2} k(\tilde{\boldsymbol{x}_0} - h^{-1}(\boldsymbol{x}_1 - \boldsymbol{y})) f_{\text{in}}(\tilde{\boldsymbol{x}_0})\, \mathrm{d}\tilde{\boldsymbol{x}_0} \\
&= \rho_{\text{out}}(h) f_{\text{out}}(h^{-1}(\boldsymbol{x}_1 - \boldsymbol{y})) \\
&= [g.f_{\text{out}}](\boldsymbol{x}_1)\,,
\end{aligned}
$$

i.e. it satisfies the equivariance condition:

$$
k \star g.f_{\text{in}} = g.[k \star f_{\text{in}}]\,.
$$

During the derivation, in (1) we used the steerability of k (Thm. 9) to apply the identity $k(\boldsymbol{x}) = \rho_{\text{out}}(h) k(h^{-1}\boldsymbol{x}) \rho_{\text{in}}(h^{-1})$. In (2), we used the substitution $\tilde{\boldsymbol{x}_0} = g^{-1}\boldsymbol{x}_0 = h^{-1}\boldsymbol{x}_0 - \boldsymbol{y}$. Note that $\left|\det\left(\frac{\partial \boldsymbol{x}_0}{\partial \tilde{\boldsymbol{x}_0}}\right)\right| = |\det(g)| = 1$ for an orthogonal transformation $g \in G$ and, therefore, the integral measure does not change.

Finally, note that the space of all (unconstrained) kernels (as considered in a conventional CNN) is a vector space. Because the equivariance condition in Thm. 9 is a linear constraint on this vector space, the set of kernels fulfilling this constraint is a vector space itself and, in particular, it is a subspace of the unconstrained vector space. As a result, any equivariant kernels can be expanded in terms of a basis for the H-steerable kernel space. Thus, to parametrize equivariant kernels, we build

one such basis and combine its elements with a set of learnable weights. The lower dimensionality of the equivariant kernel space enforces a stronger form of weight sharing and, therefore, improves the parameter efficiency of the models, as done by translational weight sharing in conventional CNNs with respect to MLPs.

3.4 Irreps Decomposition

The constraint from Thm. 9 depends on the pair of input and output types ρ_{in} and ρ_{out}. Because different pairs of types have different solutions, in principle, one needs to solve an independent constraint for each pair of input/output types appearing in the network.

The authors of [44] proposed a numerical method to compute these solutions for arbitrary pairs of *irreducible representations* (irreps) (Def. 22) which is based on the *Clebsch-Gordan decomposition* of their tensor products. While this method can be applied to arbitrary pairs of representations, it becomes prohibitively expensive for large representations, including some of those considered here. A more detailed comparison with this method can be found in Appendix C.

In this work, to efficiently compute a basis for arbitrary representations ρ_{in} and ρ_{out}, we decompose the constraint into a set of much simpler constraints defined in terms of the irreps contained in ρ_{in} and ρ_{out}. This approach relies the property described in Thm. 2, i.e., any representation $\rho : H \to \mathbb{R}^c$ of a compact group H can be decomposed into a direct sum of irreps (up to a change of basis):

$$\rho = Q^{-1} \left[\bigoplus_{i \in I} \psi_i \right] Q$$

where Q is the change of basis matrix, $\{\psi_i\}_i$ are the irreps of H and I is an index set encoding the types and multiplicities of irreps in ρ. If the set of irreps $\{\psi_i\}_i$ is known, a representation ρ can be decomposed by exploiting some results from character theory and linear algebra. We presented an introduction to character theory in Sec. 2.6 and described a method to decompose the representations of *finite* groups in Thm. 6. We discuss the implementation of group representations in Sec. 4.3.

By decomposing ρ_{in} and ρ_{out} as above in Eq. (3.9), we obtain

$$\forall h \in H,\ \boldsymbol{x} \in \mathbb{R}^2,$$
$$k(h\boldsymbol{x}) = Q_{\text{out}}^{-1} \left[\bigoplus_{i \in I_{\text{out}}} \psi_i(h) \right] Q_{\text{out}}\, k(\boldsymbol{x})\, Q_{\text{in}}^{-1} \left[\bigoplus_{j \in I_{\text{in}}} \psi_j^{-1}(h) \right] Q_{\text{in}} \quad (3.10)$$

which, with a change of variable $\kappa := Q_{\text{out}}\, k\, Q_{\text{in}}^{-1}$, becomes

$$\forall h \in H,\ \boldsymbol{x} \in \mathbb{R}^2$$
$$\kappa(h\boldsymbol{x}) \;=\; \left[\bigoplus_{i\in I_{\text{out}}} \psi_i(h)\right] \quad \kappa(\boldsymbol{x}) \quad \left[\bigoplus_{j\in I_{\text{in}}} \psi_j^{-1}(h)\right] \tag{3.11}$$

The new kernel κ is the kernel k expressed relative to the irrep bases. This last expression can be visually represented as:

$$\underbrace{\begin{pmatrix} \kappa^{i_1j_1}(gx) & \kappa^{i_1j_2}(gx) & \dots \\ \kappa^{i_2j_1}(gx) & \kappa^{i_2j_2}(gx) & \dots \\ \vdots & \vdots & \ddots \end{pmatrix}}_{\kappa(gx)} = \underbrace{\begin{pmatrix} \psi_{i_1}(g) & & \\ & \psi_{i_2}(g) & \\ & & \ddots \end{pmatrix}}_{\bigoplus_{i\in I_{\text{out}}}\psi_i(g)} \cdot \underbrace{\begin{pmatrix} \kappa^{i_1j_1}(x) & \kappa^{i_1j_2}(x) & \dots \\ \kappa^{i_2j_1}(x) & \kappa^{i_2j_2}(x) & \dots \\ \vdots & \vdots & \ddots \end{pmatrix}}_{\kappa(x)} \cdot \underbrace{\begin{pmatrix} \psi_{j_1}^{-1}(g) & & \\ & \psi_{j_2}^{-1}(g) & \\ & & \ddots \end{pmatrix}}_{\bigoplus_{j\in I_{\text{in}}}\psi_j^{-1}(g)}$$

The *block-diagonal* structure of the direct-sum representations in the right hand side implies that the constraint decomposes into a number of *independent* constraints

$$\kappa^{ij}(hx) \;=\; \psi_i(h)\, \kappa^{ij}(x)\, \psi_j^{-1}(h) \qquad \forall g \in G,\ x \in \mathbb{R}^2 \tag{3.12}$$

on blocks κ^{ij} in κ, one for each pair $i \in I_{\text{out}}$ and $j \in I_{\text{in}}$. Each constraint corresponds to an invariant subspace of the full kernels space.

We can build a basis $\mathcal{K}$ for the space of all kernels equivariant with respect to ρ_{in} and ρ_{out}, i.e., satisfying the full constraint in Eq. (3.9), as follows. First, we compute a basis $\mathcal{K}^{ij} = \{b_1^{ij}, \cdots, b_{d_{ij}}^{ij}\}$ for the space of H-steerable kernels satisfying the independent constraint (3.12) on κ^{ij} for each pair of irreps (ψ_i, ψ_j). We can take the union of all these bases, by zero-padding them appropriately, to obtain a basis for the full kernel κ, expressed in the irreps basis. A basis element b_l^{ij} is zero-padded to fill the block corresponding to κ^{ij} (i-column, j-th row) in a matrix of the same shape of κ:

$$\overline{b}_l^{ij} \;:=\; \begin{pmatrix} \ddots & \vdots & \vdots & \vdots & \iddots \\ \dots & 0 & 0 & 0 & \dots \\ \dots & 0 & b_l^{ij} & 0 & \dots \\ \dots & 0 & 0 & 0 & \dots \\ \iddots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}.$$

In other words, given the set of bases $\{\mathcal{K}^{ij}\}_{i,j}$ for the individual blocks $\{\kappa^{ij}\}_{i,j}$ of κ, we build the following $d = \sum_{ij} d_{ij}$-dimensional basis

$$\bar{\mathcal{K}} = \{\bar{\kappa}_1, \cdots, \bar{\kappa}_d\} \;:=\; \bigcup_{i \in I_{\text{out}}} \bigcup_{j \in I_{\text{in}}} \left\{ \bar{b}_1^{ij}, \cdots, \bar{b}_{d_{ij}}^{ij} \right\} \tag{3.13}$$

for the space of kernels satisfying Eq. (3.11).

Finally, we can apply the changes of basis Q_{in} and Q_{out}^{-1} to the elements of $\bar{\mathcal{K}}$ to obtain a basis for the original kernel k fulfilling Eq. (3.9):

$$\begin{aligned} \mathcal{K} = \{\kappa_1, \cdots, \kappa_d\} \;&:= \{Q_{\text{out}}^{-1}\, \bar{\kappa} Q_{\text{in}} | \bar{\kappa} \in \bar{\mathcal{K}}\} \\ &= \bigcup_{i \in I_{\text{out}}} \bigcup_{j \in I_{\text{in}}} \left\{ Q_{\text{out}}^{-1}\, \bar{b}_1^{ij} Q_{\text{in}}, \cdots, Q_{\text{out}}^{-1}\, \bar{b}_{d_{ij}}^{ij} Q_{\text{in}} \right\} \end{aligned} \tag{3.14}$$

If the basis $\mathcal{K}^{ij}$ of each block ij is complete, the completeness of the full basis $\mathcal{K}$ follows by construction. Finally, we point out that this approach can be used to solve the kernel constraints of any steerable CNNs and it is not limited to $\mathrm{E}(2)$ or its subgroups.

3.5 Kernel Constraint Solution for $H \leq \mathrm{O}(2)$

Although the method proposed in the previous section can be used to solve the kernel constraint for any group, in this work, we are interested in the specific case of planar isometries, particularly $G = (\mathbb{R}^2, +) \rtimes H \leq \mathrm{E}(2)$ with H subgroup of $\mathrm{O}(2)$. Because H is a subgroup of $\mathrm{O}(2)$, its action on $\mathbb{R}^2$ is norm-preserving, i.e.

$$||h.\boldsymbol{x}||_2 = ||\boldsymbol{x}||_2 \quad \forall h \in H, \boldsymbol{x} \in \mathbb{R}^2 \quad .$$

It follows that both the general constraint in Thm. 9 and the irreps constraint in Eq. (3.12) do not restrict the radial component of the kernels but only affect their *angular parts*. To deal with these two components independently, it is convenient to express the kernel in *polar coordinates* using the map $\boldsymbol{x} : (r, \phi) \mapsto r \cdot (\cos\phi, \sin\phi)^T$ with $r \in \mathbb{R}_0^+$ and $\phi \in [0, 2\pi)$. Moreover, the irreducible representations of H are always associated to a unique *angular frequency* (see Sec. 2.7.2). This suggests expressing the kernel in terms of an (angular) Fourier series

$$k_{\alpha\beta}^{ij}(\boldsymbol{x}(r,\phi)) \;=\; A_{\alpha\beta,0}(r) + \sum_{\mu=1}^{\infty} \Big[A_{\alpha\beta,\mu}(r) \cos(\mu\phi) \;+\; B_{\alpha\beta,\mu}(r) \sin(\mu\phi) \Big] \tag{3.15}$$

with real-valued, radially dependent coefficients $A_{\alpha\beta,\mu} : \mathbb{R}_0^+ \to \mathbb{R}$ and $B_{\alpha\beta,\mu} : \mathbb{R}^+ \to \mathbb{R}$ for each entry $k_{\alpha\beta}^{ij}$ of each block k^{ij}.

We can then replace the kernel with the expansion above in the irreps constraints in Eq. (3.12). By projecting the resulting expression on each element of the harmonic basis, the kernel constraint translates into a constraint on the expansion coefficients. Its solution requires many of the coefficients to be zero and, therefore, defines a subset of the initial harmonic basis. The block k^{ij} is then parameterized in terms of this subset using the non-zero coefficients. The completeness of the initial harmonic basis directly implies the basis built this way is complete. Similar strategies were used for the specific case of C_N in [45], of $\mathrm{SO}(2)$ in [48] and of $\mathrm{SO}(3)$ in [44].

In Table 3.1, we report the angular bases for each pair of irreducible representations of $\mathrm{O}(2)$. Note that each element of the basis is associated with a unique angular frequency. Using the method just described, we explicitly derive the bases for spaces of equivariant kernels for all subgroups $H \in \mathrm{O}(2)$ and all pairs of their irreps in Appendix B. The final solutions for $\mathrm{SO}(2)$, $(\{\pm 1\}, *)$, C_N and D_N are found in Tables B.1, B.3, B.4 and B.5. Note that the bases associated with these subgroups $H < \mathrm{O}(2)$ are larger than that associated with $\mathrm{O}(2)$ and, therefore, parametrize a larger space of filters. Indeed, smaller subgroups enforce weaker constraints on the kernels. Thus, an higher level of equivariance results in both a guaranteed behavior under transformations of the input during inference and an improved parameter efficiency of the model during training.

ψ_i \ ψ_j	trivial	sign-flip	frequency $n \in \mathbb{N}^+$
trivial	$[1]$	$\varnothing$	$\begin{bmatrix}\sin(n\phi), & -\cos(n\phi)\end{bmatrix}$
sign-flip	$\varnothing$	$[1]$	$\begin{bmatrix}\cos(n\phi), & \sin(n\phi)\end{bmatrix}$
frequency $m \in \mathbb{N}^+$	$\begin{bmatrix}\sin(m\phi)\\ -\cos(m\phi)\end{bmatrix}$	$\begin{bmatrix}\cos(m\phi)\\ \sin(m\phi)\end{bmatrix}$	$\begin{bmatrix}\cos((m-n)\phi) & -\sin((m-n)\phi)\\ \sin((m-n)\phi) & \cos((m-n)\phi)\end{bmatrix}, \begin{bmatrix}\cos((m+n)\phi) & \sin((m+n)\phi)\\ \sin((m+n)\phi) & -\cos((m+n)\phi)\end{bmatrix}$

Tab. 3.1.: Bases for the angular parts of $\mathrm{O}(2)$-steerable kernels satisfying the irrep constraint Eq. (3.12) for different pairs of input field irreps ψ_j and output field irreps ψ_i.The different types of irreps are explained in Sec. 2.7.2.

3.6 Representations and Non-Linearities

Although a general theory of steerable CNNs exists, it does not prefer any field type when implementing them. Which choice of representation ρ of H is more suitable when designing equivariant networks is still not clear. In this section, we explore different such choices.

Equivariant Linear Networks We first consider a simplified setting. Assume a linear model consisting only of convolutions with H-steerable kernels. In this case, applying any change of basis to the field types of an intermediate layer l will leave the model unchanged. Recall that a change of basis P transforms a representation

ρ_l to an *equivalent representation* $\widetilde{\rho}_l := P^{-1}\rho_l P$. Assume an irrep decomposition $\rho_l = Q_l^{-1}\left[\bigoplus_{i\in I_l}\psi_i\right]Q_l$ of ρ_l as used in Sec. 3.4 to solve the kernel constraint. Then, the equivalent representation $\widehat{\rho}_l$ will decompose as $\widetilde{\rho}_l = \widetilde{Q}_l^{-1}\left[\bigoplus_{i\in I_l}\psi_i\right]\widetilde{Q}_l$, where $\widetilde{Q}_l = Q_l P$. However, because the steerable basis derived in the irreps space in Eq. (3.13) does not depend on the change of basis Q_l, it is also not affected by the new change of basis P. Thus, it is sufficient to define field types only in terms of direct sums of irreps $\rho_l = \bigoplus_{i\in I_l}\psi_i$. In this setting, this simplifies the problem of choosing representations to that of choosing the multiplicities of each irrep.

However, usually neural networks alternate convolution layers with other layers, like non-linearities, which might be sensitive to the representations used. Indeed, a non-linearity needs to be equivariant with respect to the specific representation associated with its input. For this reason, the set of suitable non-linearities is limited by the choice of field types of the features. At the same time, a non-linearity is equivariant only to a restricted range of representations. It is, therefore, natural to study representations together with their compatible non-linearities. In the rest of this section, we will review different such combinations from the current literature.

We only consider spatially-localized non-linearities σ, i.e. which transforms a feature field $f : \mathbb{R}^2 \to \mathbb{R}^{c_{\text{in}}}$ by acting on each feature vector $f(\boldsymbol{x}) \in \mathbb{R}^{c_{\text{in}}}$ independently:

$$[\sigma(f)](\boldsymbol{x}) = \sigma(f(\boldsymbol{x})) \quad \forall \boldsymbol{x} \in \mathbb{R}^2$$

In general, the input and output types ρ_{in} and ρ_{out} of $\sigma : \mathbb{R}^{c_{\text{in}}} \to \mathbb{R}^{c_{\text{out}}}$ do not need to be the same. If the non-linearity σ is equivariant to H acting on a single feature vector $f(\boldsymbol{x})$ through ρ_{in} and ρ_{out}, i.e.

$$\sigma(\rho_{\text{in}}(h)f(\boldsymbol{x})) = \rho_{\text{out}}(h)\sigma(f(\boldsymbol{x}))\ \forall h \in H$$

then it is also equivariant to the full group G (acting through its induced representation) when applied on the entire feature field f. See Appendix A for a proof.

Unitary representations A general class of representations are *unitary representations*, i.e. representations which associates unitary matrices to the group elements and, so, whose action on a vector preserves its norm:

$$||\rho(h)f(\boldsymbol{x})||_2 = ||f(\boldsymbol{x})||_2 \ \ \forall\ h \in H \quad .$$

As shown in Appendix A.1, Any non-linearity which only acts on the *norm* of a feature vector but preserve its orientation is equivariant with respect to any unitary representations of H. Such non-linearities can be written as

$$\sigma : f(\boldsymbol{x}) \mapsto \eta(||f(\boldsymbol{x})||_2)\frac{f(\boldsymbol{x})}{||f(\boldsymbol{x})||_2}$$

where $\eta : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is a non-linearity such that $\eta(0) = 0$. In [48, 44], *Norm-ReLUs* was used as non-linearity and can be defined using $\eta(||f(\boldsymbol{x})||_2) = \mathrm{ReLU}(||f(\boldsymbol{x})||_2 - b)$, with learnable bias $b \in \mathbb{R}^+$. The authors of [37] used *squashing non-linearities* $\eta(||f(\boldsymbol{x})||_2) = \frac{||f(\boldsymbol{x})||_2^2}{||f(\boldsymbol{x})||_2^2+1}$. A conditional version of norm non-linearities was used in [44]: *Gated non-linearities* scale the norm of a feature vector $f(\boldsymbol{x})$ by a learned sigmoid gate $\frac{1}{1+e^{-s(\boldsymbol{x})}}$, parameterized by a scalar feature field $s : \mathbb{R}^2 \to \mathbb{R}$. Here, $\eta(||f(\boldsymbol{x})||_2) = ||f(\boldsymbol{x})||_2 \frac{1}{1+e^{-s(\boldsymbol{x})}}$. Because all representations we consider are unitary, we can always use norm non-linearities. In particular, any *irreducible representations* ψ_i of the groups considered here are unitary[2]. We discuss all irreps and their properties in more detail in Sec. 2.7.2.

Regular representation A popular choice when working with *finite* groups H (e.g. C_N or D_N) is the *regular representation* ρ_{reg}^H, introduced in Def. 23. This representation has size equal to the order $|H|$ of the group, e.g., N for C_N and $2N$ for D_N, and acts by permuting the $|H|$ channels of a ρ_{reg}^H-field. As a result, it is always possible to use point-wise non-linearities like ReLU; see Appendix A.2 for a proof. As mentioned in Sec. 3.1 and 3.2, regular steerable CNNs are equivalent to group-convolution architectures. Although this design generally shows very good performance, it requires high dimensional feature fields as each individual field needs $|H|$ channels. This kind of design was used for planar images in [12, 45, 22, 2, 18, 40, 33, 17], for spherical data in [8, 11] and for volumetric convolutions in [46, 47]. Additionally, a translation of the convolutional feature maps in vanilla CNNs can be interpreted as the action of the regular representation of the discrete translation group $(\mathbb{Z}^2, +)$.

Quotient representation A similar representation which acts through permutation matrices is the *quotient representation*; see Def. 24. It follows that, like regular representation, a quotient representation supports point-wise non-linearities. Instead of permuting $|H|$ channels indexed by elements of H, it permutes $|H|/|K|$ channels indexed by cosets hK in the quotient space H/K (Def. 13), for a subgroup $K \leq H$. Note that the regular representation ρ_{reg}^H is a special case for $K = e$, i.e. the trivial group. Then, a quotient representation $\rho_{\mathrm{quot}}^{H/K}$ only uses $|H|/|K|$ channels, $|K|$ times less that the regular one. This comes with the cost of having more symmetric filters, parametrized by a smaller H-steerable kernel basis. In practice, instead of containing $|H|$ transformed versions of an unconstrained filter, they use

[2] This is in general true for any *compact* group.

only $|H|/|K|$ transformed versions of a filter, which is K-invariant. We include an extended discussion about the symmetries enforced by quotient representations, together with some practical examples, in Appendix D. Quotient representations were previously considered in [13, 27].

Induced representation Regular and quotient representations are further generalized by *induced representations*, introduced in Sec. 2.4. In particular, they are built through *induction* from the trivial representation of the trivial group, $\rho_{\text{reg}}^H = \text{Ind}_{\{e\}}^H 1$, or of a subgroup $K \leq H$, $\rho_{\text{quot}}^H = \text{Ind}_K^H 1$, respectively. Induction generalizes quotient representations by allowing one to build new representations $\rho_{\text{ind}} = \text{Ind}_K^H \tilde{\rho} : H \to \text{GL}(\mathbb{R}^{c \cdot |H:K|})$ from any representation $\tilde{\rho} : K \to \text{GL}(\mathbb{R}^c)$ of the subgroup K; $|H : K|$ denotes the *index* of K in H (see Def. 12) which is equal to $|H|/|K|$ if H and K are finite groups. It is important to note that the induction from K to H, although conceptually equivalent to the induction from H to $G = (\mathbb{R}^2, +) \rtimes H$ used to define steerable CNNs in Eq. (3.8), is used here to construct a representation acts on the channels of a single feature vector rather than on the full feature field. Finally, non-linearities equivariant to an induced representation can be built from non-linearities supported by $\tilde{\rho}$.

Group Pooling and Vector Field non-linearities When using regular or quotient fields, one use the *group pooling* operation $\max : \mathbb{R}^c \to \mathbb{R},\ f(\boldsymbol{x}) \to \max(f(\boldsymbol{x}))$ to extract the maximum value of the input field. This kind of operation was previously used in [12, 45, 2, 47, 46]. Note that the representation ρ defining the input field transforms the c input channels via permutation matrices. Since the $\max$ operation is invariant to permutations, its output is invariant to the transformation of the channels through ρ. This implies that the output field is a *scalar field*. Unfortunately, this operation discards all information about the orientation of the features. In order to preserve this information while still reducing the size of the features, the authors of [30] suggested *vector field non-linearities* when using the regular representation of a discrete rotation group C_N. In particular, this non-linearity computes both the maximum response $\max(f(\boldsymbol{x}))$ and its index $\arg\max(f(\boldsymbol{x})) \in \{0, \dots, N-1\}$. Note that this index corresponds to a rotation angle $\theta_{f(\boldsymbol{x})} = \frac{2\pi}{N} \arg\max(f(\boldsymbol{x}))$. Then, the vector field non-linearity $\sigma_{\text{vect}} : \mathbb{R}^N \to \mathbb{R}^2$ maps the input feature vector $f(\boldsymbol{x})$ to a two dimensional vector $v(\boldsymbol{x}) = \max(f(\boldsymbol{x}))(\cos(\theta_{f(\boldsymbol{x})}), \sin(\theta_{f(\boldsymbol{x})}))^T$. We prove the equivariance of this non-linearity in A.3.

Tensor Product Any pair of feature fields $f_1 : \mathbb{R}^2 \to \mathbb{R}^{c_1}$ and $f_2 : \mathbb{R}^2 \to \mathbb{R}^{c_2}$ can be combined via the *tensor product* $f_1 \otimes f_2 : \mathbb{R}^2 \to \mathbb{R}^{c_1 c_2}$. If the input fields transform respectively as ρ_1 and ρ_2, their product transforms under the *tensor product representation* $\rho_1 \otimes \rho_2 : H \to \text{GL}(\mathbb{R}^{c_1 \cdot c_2})$. The tensor product is already a non-linear operation, hence it is not usually combined with other non-linearities and can be

used to build equivariant networks with arbitrary representations. This operation has been discussed in [25, 44, 26, 1].

Direct Sum One can also always combine two or more feature fields $\{f_i : \mathbb{R}^2 \to \mathbb{R}^{c_1}\}_i$ by concatenating their channels, i.e. taking the *direct sum* $\bigoplus_i f_i : \mathbb{R}^2 \to \mathbb{R}^c$, with $c = \sum_i c_i$. We also used this operation in Sec. 3.2 to construct feature spaces made of different feature fields. The resulting field transforms according to the *direct sum representation* $\rho = \bigoplus_i \rho_i$, as defined in Def. 21. Note that the input fields $\{f_i\}_i$ will still transform independently and each field f_i can still be interpreted independently. Thus, it is possible to apply different non-linearities σ_i to each field f_i and, if σ_i is equivariant with respect to ρ_i, the whole layer is equivariant with respect to ρ; see Appendix A for a proof.

Finally, because the theory of steerable CNNs does not favor any of these designs, we perform an extensive experimental study to compare them in Sec. 5.1.

3.7 Group Restriction

The rationale for the development of equivariant networks, and the reason for their success, is exploiting the symmetries in the data. However, many interesting tasks are characterized only by few symmetries, if any, in their patterns, limiting the possible applications of such models. Moreover, the amount of symmetries can change when observing the signals contained in the data at different scales. In particular, this is common in natural images. While small patterns (from smaller details to edges or intensity gradients) usually appear in arbitrary orientations and reflections, these symmetries tend to disappear on larger length scales. For instance, pictures are usually aligned in a vertical direction, with large patterns appearing only in one orientation. As a practical example, consider a dataset of faces or natural landscapes. In both cases, images have a strong notion of vertical alignment (e.g., the eyes are always above the nose and the mouth). Nevertheless, by looking only at small areas of the images, one finds patterns repeating in arbitrary orientations. Note also that, often, these images are still symmetric to reflections along the vertical axis. See, for example, Fig. 3.2: while the sunflowers are rotationally symmetric, the image has a vertical alignment, with the sky always shown above the ground. This hierarchical

Fig. 3.2.: Example of natural image showing local rotational symmetry but a global vertical orientation. Credit: MikeLynch, CC BY-SA 3.

structure has already been observed in the features of conventional CNNs [50]. Because of the local field of view of their convolution layers (due to the limited size of their filters), the first layers learn to detect small edges and corners. In contrast, deeper layers learn gradually more complex and larger patterns, which tend to appear in fewer orientations. This suggests a neural network design where each layer is adapted to the symmetries which are manifested in the length scale of its field of view.

The loss of symmetries in the hierarchy of feature maps can be modeled by relaxing the equivariance constraint on the kernels at different depths of the network. Mathematically, this can be done through the *restriction* of the equivariance group $G = (\mathbb{R}^2, +) \rtimes H$ to a subgroup $G' = (\mathbb{R}^2, +) \rtimes K \leq G$, where $K \leq H$. For instance, when working with the data in the previous examples, one could use a network which is $H = \mathrm{O}(2)$ rotation and reflection equivariant in the first layers but only $K = (\{\pm 1\}, *)$ reflection equivariant in the following ones.

If restriction from H to K is performed after the l-th layer, it is necessary to reinterpret the features produced by the l-th layer, transforming according to H, such that they are compatible with the $l+1$-th layer, transforming according to K. More precisely, if the l-th layer produces a ρ-field, with $\rho : H \to \mathrm{GL}(\mathbb{R}^c)$, in order to guarantee equivariance to K of the following layer, the field then needs to be reinterpreted as a $\tilde{\rho}$-field, with $\tilde{\rho} : K \to \mathrm{GL}(\mathbb{R}^c)$ a representation of K compatible with ρ, i.e.:

$$\forall k \in K \leq H \rho(k) = \tilde{\rho}(k) \quad .$$

Note that $\tilde{\rho}$ can be constructed just by restricting the domain of ρ to K. Such representation is called the the *restricted representation* of ρ (Def. 25):

$$\tilde{\rho} \;:=\; \mathrm{Res}^H_K(\rho) : \; K \to \mathrm{GL}(\mathbb{R}^c), \quad k \mapsto \rho(k)\,. \tag{3.16}$$

Because the method in Sec. 3.3 allows us to build steerable convolution layers for arbitrary representations of K, we can always immediately work with Res^H_K ρ-fields. We give more details about the implementation of group restriction in Sec. 4.3.1.

One can also imagine the opposite setting, where the data does not show any real symmetry on the small scale. However, a global symmetry emerges when discarding small details and looking at the patterns at a larger scale. An equivariant model able to exploit this could be designed by lifting a ρ-field, equivariant to a subgroup $K \leq H$ describing the local symmetry, to an induced Ind^H_K ρ-field, equivariant to the global symmetry H.

In Sec. 5.2, 5.4, 5.5 and 5.6 we examine the effectiveness of enforcing local equivariance via the use of group restrictions on datasets with different levels of global symmetries.

Implementation: *E2CNN* Library

4

> *Beware of bugs in the above code; I have only proved it correct, not tried it.*
>
> **— Donald E. Knuth**

4.1 Convolution Layer

As derived in Sec. 3.3, $\mathrm{E}(2)$-steerable CNNs require convolution layers to use $\mathrm{O}(2)$-steerable filters. Then, a real implementation of the convolution layer needs to include the following three steps:

1. derivation of a basis for the steerable kernels,
2. its contraction using the learned expansion coefficients to build the kernel, and
3. execution of the convolution with the kernel just built.

Because the steerable basis only depends on the input and output representations, which are defined during the design of the model, and does not change throughout the training, it can always be precomputed, avoiding unnecessary overhead during training.

Therefore, the construction of a convolution layer resembles the following process. First, the user chooses the input and output representations ρ_{in} and ρ_{out} of $H \leq \mathrm{O}(2)$, defining the input and output field types. We can immediately generate a basis $\mathcal{K} = \{\kappa_1, \dots \kappa_d\}$ as in Sec. 3.4 for the space of equivariant kernels satisfying Thm. 9.

However, the method described in Sec. 3.4 requires the decomposition of ρ_{in} and ρ_{out} into direct sums of irreps to be available. We can find the types and multiplicities of the irreps appearing in the decomposition of both representations using character theory, see Sec. 2.6 and Sec. 4.3. In general, the change of basis matrices can be computed numerically by solving the linear system $\rho(h) = Q^{-1}[\bigoplus_{i \in I} \psi_i(h)]Q \; \forall h \in H$. Because most of the representations we use are regular, quotient or induced

representations, a more efficient method can be used in practice. We describe in more detail the methods used to decompose representations in Sec. 4.3.

Knowing the decompositions of both ρ_{in} and ρ_{out}, for each pair of input/output irreps ψ_i and ψ_j we can retrieve the basis $\mathcal{K}^{ij} = \{b_1^{ij}, \cdots, b_{d_{ij}}^{ij}\}$ analytically derived in Appendix B and reported in Appendix B.1. Combining these solutions with the change of basis matrices Q_{in} and Q_{out} found during the irreps decomposition, we construct the full basis $\mathcal{K} = \{\kappa_1, \cdots, \kappa_d\}$ as in Eq. 3.14.

As discussed in Sec. 3.5, because $H \leq \mathrm{O}(2)$, the kernel constraint does not restrict the radial component of the filters. Indeed, the analytical bases derived in Appendix B are defined only over the angular component of the filters and so will be the constructed basis $\mathcal{K}$. Hence, we still need to select a basis for the radial part of the filters. A common choice [48, 45, 44] is using Gaussian radial profiles

$$\eta_R(r) = \exp\left(\frac{1}{2\sigma^2}(r - R)^2\right) \tag{4.1}$$

of width σ, centered at radii $R = 1, \ldots, \lfloor s/2 \rfloor$.

4.1.1 Discretization and Anti-Aliasing

In practice, to perform numerical computations, we need to discretize continuous signals in the inputs and the features. As commonly done in computer vision, planar signals are sampled on a pixel grid $\mathbb{Z}^2$. It is important to draw attention to the fact that this discretization reduces the continuous symmetries of the plane $\mathbb{R}^2$ to only the discrete symmetries of the grid $\mathbb{Z}^2$, which include only subgroups of D_4. Indeed, these were the first groups employed in the literature [12, 18, 13]. Thus, one can only properly enforce equivariance only to these subgroups, whereas equivariance to larger groups can only be approximate. We discretize the convolution filters by sampling all elements of the analytical kernel basis $\{\kappa_1, \ldots, \kappa_d\}$ on a $s \times s$ grid and store them in a $(d, c_{\text{out}}, c_{\text{in}}, s, s)$-dimensional array. Because the sampling and the linear combination of the basis commute [45], contracting the sampled basis with learned weights is equivalent to contracting the continuous (analytical) basis and, then, sampling the resulting filter on the grid.

Unfortunately, *aliasing* can occur when sampling continuous signals. Briefly, aliasing is observed when different signals become indistinguishable after being sampled on a discrete set of points. It is essential to prevent this kind of effect when implementing steerable convolution to ensure approximate equivariance. Recall that the kernel bases for steerable filters are constituted by a subset of the angular harmonics as in Table 3.1. In particular, each element of the basis is associated with a single angular

frequency and the frequencies present in the basis depend on the specific input and output irreps. However, if the sampling rate of the basis is not high enough, a high-frequency basis element might be identical to a lower-frequency filter, which does not belong to the analytical basis. This can result in non-equivariant elements in the discrete kernel basis.

To avoid aliasing and guarantee approximate equivariance of the kernels, we band-limit the kernel bases. As each element of our bases is associated with only one angular frequency, anti-aliasing only requires discarding elements with a frequency higher than a chosen cutoff frequency. In Eq. (4.1), we parametrized the radial component of the filters with different Gaussian radial profiles. This basis splits a steerable filter in multiple rings with different radii, one for each element of the radial basis. We observe that larger rings correspond to higher sampling rates of the angular basis. This suggests to band-limit the angular bases at different radii using different cutoff frequencies.

We experiment with different band-limiting policies. Although larger rings always feature higher sampling rates, the largest rings can lose part of their support as they partially fall outside the grid. We empirically observe that the high-frequency elements sampled in the outermost ring often introduce high equivariance errors. For this reason, we have found beneficial to both reduce the width (σ in Eq. (4.1)) and use a lower cutoff frequency in the largest rings.

4.1.2 Block-wise basis expansion

As the features of conventional CNNs contain multiple channels, the features of steerable CNNs are commonly built as a stack of multiple feature fields, see Sec. 3.2. The *direct sum* of feature types induces a block-diagonal structure in the representation acting on the features. This allows us to decompose the kernel constraint in Thm. 9 into simpler constraints, enabling a more efficient implementation.

Consider a convolution layer with input type ρ_{in} and output type ρ_{out}. Assume that the features comprise multiple individual feature fields and, therefore, the input and output representations are direct sum representations of the form $\rho_{\text{in}} = \bigoplus_\delta \rho_{\text{in},\delta}$ and $\rho_{\text{out}} = \bigoplus_\gamma \rho_{\text{out},\gamma}$. Then, the constraint in Thm. 9 on the full kernel k, mapping between ρ_{in} and ρ_{out}, corresponds to independent constraints on each block $k^{\gamma\delta}$ of k, which maps between individual fields of type $\rho_{\text{in},\delta}$ and $\rho_{\text{out},\gamma}$. Thus, we can build a basis for k by computing a basis $\{k_1^{\gamma\delta}, \dots, k_{d^{\gamma\delta}}^{\gamma\delta}\}$ for each different pair $(\rho_{\text{in},\delta}, \rho_{\text{out},\gamma})$ of input and output types.

This decomposition should not be confused with the one used in Sec. 3.4. Although they share the same idea, in Sec. 3.4 we decompose the constraint associated to a single pair of individual input and output fields of types $\rho_{\text{in},\delta}$ and $\rho_{\text{out},\gamma}$ into simpler constraints associated to their irreps. Here, instead, we decompose the constraint associated to the full feature fields of type ρ_{in} and ρ_{out} into constraints associated to each pair of individual input and output fields $\rho_{\text{in},\delta}$ and $\rho_{\text{out},\gamma}$, i.e. the constraints solved in Sec. 3.4.

During training, in each forward pass, we build the convolution filters by linearly combining the elements of the kernel bases using learnable weights. More precisely, each block $k^{\gamma\delta}$ is associate with a sampled basis $\{k_1^{\gamma\delta}, \dots, k_{d^{\gamma\delta}}^{\gamma\delta}\}$, stored as a $(d^{\gamma\delta}, c_{\text{out},\gamma}, c_{\text{in},\delta}, s, s)$-dimensional tensor, and a $(d^{\gamma\delta})$-dimensional tensor of learnable weights. Then, each block $k^{\gamma\delta}$ is built by contracting the sampled basis tensor along its first dimension using its weights tensor, which can be implemented with a simple `matmul` call.

Note that, if $\rho_{\text{in},\delta}$ occurs $n > 1$ times in the input type ρ_{in} or $\rho_{\text{out},\gamma}$ occurs $m > 1$ times in the output type ρ_{out}, k contains $n \cdot m$ blocks sharing the same input and output field types and, therefore, the same basis. Instead of looping over all $m \cdot n$ blocks, we can group them together and simultaneously contracting the basis with $m \cdot n$ different set of weights. Unfortunately, if more different field types are present in the input or output representations, the $m \cdot n$ blocks might not be contiguous inside k. If that is the case, sparse access of the memory using advanced indexing[1] is necessary. Although this is still more efficient than iterating over all blocks, it can result in poor runtime performances with respect to conventional CNNs. However, when both the input and output fields contain a single field type (not necessarily the same), the basis expansion can be efficiently performed with a single batched matrix multiplication followed by a reshaping of the tensor. Fortunately, this is often the case. Indeed, group-convolution based architectures (GCNNs) fall in this category too.

Because the resulting kernels look precisely like conventional convolution filters, we can leverage the optimized standard convolution routine present in most common deep learning frameworks. Furthermore, when using features of a single type as above and fairly large inputs, basis expansion has only a minor contribution to the total cost of the layer with respect to the actual convolution. Finally, at test time, as the learnable weights do not need to be modified, there is no need to perform the basis expansion at each forward pass. Instead, the kernels can be built once and the steerable convolution layer converted to a conventional convolution layer without any additional computational cost with respect to conventional CNNs.

[1] https://numpy.org/doc/stable/reference/arrays.indexing.html

4.2 Equivariant Statistics and Batch-Normalization

In this section we discuss some simple statistics of steerable feature fields. The following results are particularly useful to define and to implement the *batch normalization* layers in our steerable CNN architectures.

Let G be a group and ρ a representation of G on a vector space V. We assume that **a symmetry in the data also implies a symmetry in its distribution**, i.e.

$$\forall g \in G \quad \forall \boldsymbol{y} \in V \quad \Pr(\boldsymbol{y}) = \Pr(g.\boldsymbol{y})$$

and, so:

$$\forall g \in G \quad \forall \boldsymbol{y} \in V \quad \Pr(\boldsymbol{y}) = \Pr(\rho(g)\boldsymbol{y})$$

Define $G\backslash V$ the right quotient space of V with respect to the left action of the group G, i.e. $G\backslash V = \{G.\boldsymbol{y} \mid \boldsymbol{y} \in V\}$, where a coset is defined as $G.\boldsymbol{y} = \{\rho(g)\boldsymbol{y} \mid g \in G\}$. See Def. 11 and Def. 13. Note that all vectors within the same coset $G.\boldsymbol{y}$ have the same probability, i.e. $\boldsymbol{x} \in G.\boldsymbol{y} \implies \Pr(\boldsymbol{x}) = \Pr(\boldsymbol{y})$. Then, defining a section $s : G\backslash V \to G$ (Def. 14) mapping each coset to a *representative* element, for any vector $\boldsymbol{y}$ there exists an element $g \in G$ such that $\boldsymbol{y} = \rho(g)s(G.\boldsymbol{y})$, i.e. every vector can be built by transforming the representative of its coset $G.\boldsymbol{y}$ with the representation of an element $g \in G$. If the representation ρ is *faithful*, the element g is unique.

Let's call R the set of all *representative* elements, i.e. $R = \{\boldsymbol{r} = s(G.\boldsymbol{y}) \mid G.\boldsymbol{y} \in G\backslash V\}$. Then, it follows that:

$$\begin{aligned}
\mathop{\mathbb{E}}_{\boldsymbol{y}\in V}[\boldsymbol{y}] &= \mathop{\mathbb{E}}_{\boldsymbol{y}\in V}[\rho(g)s(G.\boldsymbol{y})] \\
&= \mathop{\mathbb{E}}_{g\in G}\mathop{\mathbb{E}}_{\boldsymbol{r}\in R}[\rho(g)\boldsymbol{r}] \\
&= \mathop{\mathbb{E}}_{g\in G}[\rho(g)\mathop{\mathbb{E}}_{\boldsymbol{r}\in R}[\boldsymbol{r}]] \\
&= \mathop{\mathbb{E}}_{g\in G}[\rho(g)]\mathop{\mathbb{E}}_{\boldsymbol{r}\in R}[\boldsymbol{r}]
\end{aligned}$$

Assuming a normalized *Haar* measure μ over G[2], the expectation over G can be written as $\mathbb{E}_{g\in G}[\rho(g)] = \int_G \mathrm{d}\mu(g)\rho(g)$. First, we assume ρ is an *irreducible representation* of G, which we indicate with ψ for consistency; then, the integral $\int_G \mathrm{d}\mu(g)\psi(g)$ is always equal to the null matrix containing only zeros except when ψ is the *trivial representation* $\psi_0(g) = 1 \ \ \forall g \in G$, whose integral is equal to 1. This implies that the

[2]We can choose a counting measure for discrete groups such that the integral becomes the usual sum over the group, normalized by the group's size.

mean of any vector transforming according to a non-trivial irreducible representation is always zero.

Assume now a general representation ρ with an irreps decomposition

$$\rho(g) = D \left(\bigoplus_{i \in I} \psi_i(g) \right) D^{-1} .$$

Then the expectation becomes

$$\mathop{\mathbb{E}}_{g \in G}[\rho(g)] = \int_G \mathrm{d}\mu(g) \rho(g) = D \underbrace{\left(\bigoplus_{i \in I} \int_G \mathrm{d}\mu(g) \psi_i(g) \right)}_{S} D^{-1}$$

The matrix $S = \bigoplus_{i \in I} \int_G \mathrm{d}\mu(g) \psi_i(g)$ is then a null matrix, containing only ones in the i-th entry of the diagonal if and only if ψ_i is a trivial representation. Therefore, one can pre-compute the matrix $P = \mathbb{E}_{g \in G}[\rho(g)] = DSD^{-1}$ and then estimate the mean of $\boldsymbol{y}$ from N samples $\{\boldsymbol{y}_i\}_{i=1}^N$ as $\bar{\boldsymbol{y}} = P \frac{1}{N} \sum_i \boldsymbol{y}_i$. It follows that we can estimate the mean by computing the sample average and then multiplying it by the pre-computed matrix P.

Note that the *regular representation* of any group G only contains a single copy of the trivial representation; in addition, the column of D corresponding to this irrep always contains the same value in all rows. It follows that P is a constant matrix, containing the same value in all entries. When G is a discrete group, one can choose an orthonormal basis such that $P_{ij} = \frac{1}{|G|}$. As a result, the multiplication by P only averages the estimated mean $\frac{1}{N} \sum_i \boldsymbol{y}_i$ over the whole group G, which corresponds to the usual sharing of the bias parameter in the *batch normalization* in GCNNs [12].

Assuming centralized data, i.e. $\mathbb{E}_{\boldsymbol{y} \in V} = 0$, the covariance matrix is:

$$\begin{aligned}
\mathop{\mathbb{E}}_{\boldsymbol{y} \in V}[\boldsymbol{y}\boldsymbol{y}^T] &= \mathop{\mathbb{E}}_{\boldsymbol{y} \in V}[\rho(g) s(G.\boldsymbol{y}) s(G.\boldsymbol{y})^T \rho(g)^T] \\
&= \mathop{\mathbb{E}}_{g \in G} \mathop{\mathbb{E}}_{\boldsymbol{r} \in R}[\rho(g) \boldsymbol{r}\boldsymbol{r}^T \rho(g)^T] \\
&= \mathop{\mathbb{E}}_{g \in G}[\rho(g) \mathop{\mathbb{E}}_{\boldsymbol{r} \in R}[\boldsymbol{r}\boldsymbol{r}^T] \rho(g)^T] \\
&= \int_g \rho(g) \mathop{\mathbb{E}}_{\boldsymbol{r} \in R}[\boldsymbol{r}\boldsymbol{r}^T] \rho(g)^T \, \mathrm{d}\mu(g)
\end{aligned}$$

Now, note that if ρ is an orthonormal representation, the resulting matrix is an *intertwiner* of ρ, that is, it commutes with ρ for any $g \in G$. Besides, if ρ is an irreducible representation, using Thm. 3, it follows that the covariance matrix needs to be an isomorphism. From this moment, for simplicity, we restrict our consideration only to subgroups of $\mathrm{O}(2)$ and real representations. This implies that $\mathbb{E}_{\boldsymbol{y} \in V}[\boldsymbol{y}\boldsymbol{y}^T]$ is either a scalar (for $d = 1$-dimensional irreps) or a scalar multiple of a 2×2

orthogonal matrix (for $d = 2$-dimensional representations). However, because the covariance matrix is also symmetric and semi-positive definite, it must be a multiple of the identity, i.e. $\exists \lambda \geq 0$ s.t. $\mathbb{E}_{\boldsymbol{y} \in V}[\boldsymbol{y}\boldsymbol{y}^T] = \lambda I$. Note that

$$d\lambda = \mathrm{Tr}(\mathop{\mathbb{E}}_{\boldsymbol{y} \in V}[\boldsymbol{y}\boldsymbol{y}^T]) = \mathop{\mathbb{E}}_{\boldsymbol{y} \in V}[||\boldsymbol{y}||_2^2] \quad ,$$

where d is the dimensionality of ρ. Therefore, we can estimate the covariance of $\boldsymbol{y}$ from N samples $\{\boldsymbol{y}_i\}_{i=1}^N$ as $Q = \frac{1}{d}\bar{\lambda} I$, where $\bar{\lambda} = \frac{1}{N-1}\sum_i ||\boldsymbol{y}_i||_2^2$.

Consider now an arbitrary representation ρ with an irreps decomposition $\rho = D\left(\bigoplus_{i \in I} \psi_i\right) D^T$. Then, for a vector $\boldsymbol{y}$ in the representation space V of ρ, we write $\boldsymbol{y} = D\left(\bigoplus_{i \in I} \boldsymbol{y}_i\right)$, where $\boldsymbol{y}_i$ is the projection of $\boldsymbol{y}$ in the subspace of ψ_i. Then, the covariance matrix can be written as

$$\begin{aligned}
\mathop{\mathbb{E}}_{\boldsymbol{y} \in V}[\boldsymbol{y}\boldsymbol{y}^T] &= \int_g \rho(g) \mathop{\mathbb{E}}_{\boldsymbol{r} \in R}[\boldsymbol{r}\boldsymbol{r}^T] \rho(g)^T \, \mathrm{d}\mu(g) \\
&= D \underbrace{\int_g \left(\bigoplus_{i \in I} \psi_i(g)\right) \mathop{\mathbb{E}}_{\boldsymbol{r} \in R} \left[\left(\bigoplus_{i \in I} \boldsymbol{r}_i\right)\left(\bigoplus_{i \in I} \boldsymbol{r}_i\right)^T\right] \left(\bigoplus_{i \in I} \psi_i(g)\right)^T \mathrm{d}\mu(g)}_{\tilde{Q}} D^T \\
&= D\tilde{Q}D^T
\end{aligned}$$

Note that the matrix $\tilde{Q}$ has a block structure, where the block (i, j) contains $\tilde{Q}_{ij} = \int_g \psi_i(g) \, \mathbb{E}_{\boldsymbol{r} \in R}\left[\boldsymbol{r}_i \boldsymbol{r}_j^T\right] \psi_i(g)^T \, \mathrm{d}\mu(g)$, for $i, j \in I$. It follows that $\tilde{Q}_{ij}$ is an intertwiner between ψ_i and ψ_j and, therefore, either it is a null matrix, if $\psi_i \ncong \psi_j$, or $\exists \lambda_{ij} \geq 0$ s.t. $\tilde{Q}_{ij} = \lambda_{ij} I$, if $\psi_i \cong \psi_j$. In general, this condition is not sufficient to obtain a diagonal covariance $\mathbb{E}_{\boldsymbol{y} \in V}[\boldsymbol{y}\boldsymbol{y}^T]$. Nevertheless, by assuming $\lambda_{ij} = 0$ whenever $i \neq j$, $\tilde{Q}$ assumes a block-diagonal form. By further assuming $\lambda_{ii} = \lambda_{jj} =: \lambda$ for any $i, j \in I$, it follows that $\tilde{Q} = \lambda I$ and, therefore, $\mathbb{E}_{\boldsymbol{y} \in V}[\boldsymbol{y}\boldsymbol{y}^T] = \lambda I$.

Note that when ρ is the *regular* or a *quotient* representation of H, this corresponds to the common assumption made in the *Batch Normalization* layer of GCNNs [12] (and similarly in CNNs). Consequently, we use this same assumption in the implementation of our equivariant batch normalization for these representations.

4.3 Representations

A representation $\rho : H \to \mathbb{R}^d$ is implemented as a direct sum of its irreps combined with an invertible $d \times d$ change of basis. Therefore, for any group H, we create all its irreducible representations $\{\psi_i\}_i$ in a functional form (in this way, we can generate all irreps even for infinite groups, which have an infinite number of irreps). A representation $\rho = D\left(\bigoplus_{i \in \mathcal{I}} \psi_i\right) D^{-1}$ is then identified by the change of basis D

and the index set $\mathcal{I}$. This enables us to immediately decompose the kernel constraint in terms of irreps as in Sec. 3.4.

In order to deal with arbitrary representations ρ of finite groups H, we can use Character theory as explained in Sec. 2.6 to find the index set I and, then, numerically solve the linear system $\forall h \in H \quad \rho(h) = D\left(\bigoplus_{i\in\mathcal{I}} \psi_i(h)\right) D^{-1}$ for D, which is equivalent to

$$\forall h \in H \quad \left(I \otimes \rho(h) - \left(\bigoplus_{i\in\mathcal{I}} \psi_i(h) \right)^T \otimes I \right) \mathrm{vec}(D) = 0$$

where I is the $d \times d$ identity matrix. In practice, because all the representations used are either irreducible representations or induced representations, we never need to use this algorithm. In the second case, which includes regular and quotient representations as special cases, we directly build the representation by first computing the multiplicity of the irreps in $\mathcal{I}$ and then generating the change of basis D using the method described in Appendix G.

4.3.1 Group Restriction

To efficiently implement *group restriction*, we leverage the following property. Given $H \leq G$ and a representation $\rho = D\left(\bigoplus_{i\in\mathcal{I}} \psi_i\right) D^{-1}$ of G, it holds that:

$$\mathrm{Res}^G_H \rho = D \left(\bigoplus_{i\in\mathcal{I}} \mathrm{Res}^G_H \psi_i \right) D^{-1} \quad .$$

$\mathrm{Res}^G_H \psi_i$ is not in general an irrep of H, but will be equal to $\mathrm{Res}^G_H \psi_i = D_i \left(\bigoplus_{j\in\mathcal{I}_i} \sigma_j \right) D_i^{-1}$, where $\{\sigma_j\}_j$ are irreps of H. We can, therefore, analytically restrict each irrep of each group, i.e. pre-compute (in a functional form) all possible $\mathrm{Res}^G_H \psi_i$. Then, the restriction of an arbitrary representation can be computed as above.

4.3.2 Representation Disentanglement

Sometimes, it happens that, through restriction, a representation can split in two or more independent representations. For instance, the restriction of the regular representation ρ^G_{reg} of a group G to a subgroup H results in a representation containing $|G:H|$ copies of the regular representation of H, i.e. one for each coset in G/H. Recall that $|G:H|$ is the *index* of H in G, see Def. 12. However, depending on how the representation ρ^G_{reg} is defined, the restricted representation might not be block-diagonal and, so, the subgroup's regular representations might not be clearly separated. In other words, $\exists P$ s.t. $\rho^G_{\text{reg}} = P \bigoplus_i^{|G:H|} \rho^H_{\text{reg}} P^{-1}$. Because any element

$g \in G$ can be uniquely identified by an element $h \in H$ and a coset $gH \in G/H$, the matrix P is necessarily a permutation matrix.

For example, this happens when restricting the regular representation of $G = \mathrm{D}_3$:

g	e	r	r^2	f	rf	$r^2 f$
$\rho_{\mathrm{reg}}^{\mathrm{D}_3}(g)$	$\begin{bmatrix} 1&0&0&0&0&0\\ 0&1&0&0&0&0\\ 0&0&1&0&0&0\\ 0&0&0&1&0&0\\ 0&0&0&0&1&0\\ 0&0&0&0&0&1 \end{bmatrix}$	$\begin{bmatrix} 0&0&1&0&0&0\\ 1&0&0&0&0&0\\ 0&1&0&0&0&0\\ 0&0&0&0&0&1\\ 0&0&0&1&0&0\\ 0&0&0&0&1&0 \end{bmatrix}$	$\begin{bmatrix} 0&1&0&0&0&0\\ 0&0&1&0&0&0\\ 1&0&0&0&0&0\\ 0&0&0&0&1&0\\ 0&0&0&0&0&1\\ 0&0&0&1&0&0 \end{bmatrix}$	$\begin{bmatrix} 0&0&0&1&0&0\\ 0&0&0&0&0&1\\ 0&0&0&0&1&0\\ 1&0&0&0&0&0\\ 0&0&1&0&0&0\\ 0&1&0&0&0&0 \end{bmatrix}$	$\begin{bmatrix} 0&0&0&0&1&0\\ 0&0&0&1&0&0\\ 0&0&0&0&0&1\\ 0&1&0&0&0&0\\ 1&0&0&0&0&0\\ 0&0&1&0&0&0 \end{bmatrix}$	$\begin{bmatrix} 0&0&0&0&0&1\\ 0&0&0&0&1&0\\ 0&0&0&1&0&0\\ 0&0&1&0&0&0\\ 0&1&0&0&0&0\\ 1&0&0&0&0&0 \end{bmatrix}$

to the reflection group $H = (\{\pm 1\}, *) \cong \mathrm{C}_2$:

h	e	f
$\left(\mathrm{Res}_{\mathrm{C}_2}^{\mathrm{D}_3} \rho_{\mathrm{reg}}^{\mathrm{D}_3}(g)\right)(h)$	$\begin{bmatrix} 1&0&0&0&0&0\\ 0&1&0&0&0&0\\ 0&0&1&0&0&0\\ 0&0&0&1&0&0\\ 0&0&0&0&1&0\\ 0&0&0&0&0&1 \end{bmatrix}$	$\begin{bmatrix} 0&0&0&1&0&0\\ 0&0&0&0&0&1\\ 0&0&0&0&1&0\\ 1&0&0&0&0&0\\ 0&0&1&0&0&0\\ 0&1&0&0&0&0 \end{bmatrix}$

Indeed, in $\mathrm{Res}_{\mathrm{C}_2}^{\mathrm{D}_3} \rho_{\mathrm{reg}}^{\mathrm{D}_3}(g)$, the three pairs of entries $(1,4)$, $(2,6)$ and $(3,5)$ never mix with each other but only permute internally. Moreover, each pair transforms according to $\rho_{\mathrm{reg}}^{\mathrm{C}_2}$. Through a permutation of the entries P, it is possible to make all the entries belonging to the same pair contiguous. This reshuffled representation is then equal to $\rho_{\mathrm{reg}}^{\mathrm{C}_2} \oplus \rho_{\mathrm{reg}}^{\mathrm{C}_2} \oplus \rho_{\mathrm{reg}}^{\mathrm{C}_2}$. Though theoretically equivalent, an implementation of this representation where the entries are contiguous is convenient when computing functions over single fields like batch normalization.

Given a d-dimensional representation $\rho = D\left(\bigoplus_{i \in \mathcal{I}} \psi_i\right) D^{-1} : G \to \mathrm{GL}(\mathbb{R}^d)$, one can always find a permutation matrix P which decomposes ρ is a direct sum $\rho = P\left(\rho_1 \oplus \rho_2 \oplus \cdots \oplus \rho_n\right) P^T$ of $n \geq 1$ sub-representations. In general, this decomposition is a direct consequence of the sparsity of the matrix D. Indeed, to find P, it is sufficient to check for each dimension $j \in \{1, \ldots, d\}$ in the representation space of ρ, for which irreps ψ_i the block $D_{j,i}$ of D has at least a non-zero entry. This property defines a *bipartite graph* between the set of irreps $\{\psi_i\}_{i \in \mathcal{I}}$ and the dimensions $\{j = 1, \ldots, d\}$ in the representation space of ρ. Note that the connected components of this graph correspond to the sub-representations $\{\rho_k\}$ in ρ. Thus, by finding these connected components, one obtains the matrix P and the decomposition above.

4.4 *e2cnn* Library

An implementation of the methods described in this work has been released in the form of a *Python* library based on *PyTorch* at https://github.com/QUVA-Lab/e2cnn. We provide an implementation of different mathematical objects (e.g. groups, representations, irreps, direct sum, induced representations) and many equivariant layers and operations (e.g. steerable convolution Sec. 3.3, equivariant non-linearities and invariant pooling Sec. 3.6, batch normalization Sec. 4.2, dropout, spatial pooling and weight initialization).

We extend *PyTorch*'s tensors to *geometric tensors* to be able to describe feature fields. Recall that feature fields are typed features associated with a transformation law, Sec 3.2. Analogously, a geometric tensor is a wrapper class containing both a `torch.Tensor` instance and a *field type*. Field types are objects describing the symmetry $G = (\mathbb{R}^2, +) \rtimes H$ considered and the transformation law of feature fields under the action of the symmetry group H, i.e. a representation $\rho = \bigoplus_i \rho_i : H \to \mathrm{GL}(() \mathbb{R}^c)$. All equivariant layers are associated to a pair of input and output types, enabling a simple form of dynamic type-checking. This prevents one from feeding feature fields into layers whose transformations laws are not compatible, reducing the risk of breaking the model's equivariance during its design.

We build a high-level user interface to abstract away most of the complexity arising from group representation theory and the details about the kernel space constraints and their solutions. The following code snippets are an example of how the first convolution layer of a C_8-equivariant network is built and used.

```
1    from e2cnn import gspaces
2    from e2cnn import nn
3    import torch
```

First, the necessary packages are imported.

```
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])
```

Now, all that is needed to define a convolution layer is choosing a group $G = (\mathbb{R}^2, +) \rtimes H$ and how it acts on the input and output features. In line 4, we have chosen the symmetry group $H = \mathrm{C}_8$ and indicated that it acts on the plane $\mathbb{R}^2$ via planar rotations. In line 5, we have defined the field type of the input ρ_{in} by

specifying it contains 3 scalar fields, described by trivial representations ψ (e.g. a RGB image, see also Sec. 3.2), i.e. $\rho_{\text{in}} = \bigoplus_{i=1}^{3} \psi$. Similarly, in line 6, we have built the output type with 10 regular representations $\rho_{\text{reg}}^{\text{C}_8}$ of C_8, i.e. $\rho_{\text{out}} = \bigoplus_{j=1}^{10} \rho_{\text{reg}}^{\text{C}_8}$.

```
7  conv = nn.R2Conv(feat_type_in, feat_type_out, kernel_size=5)
8  relu = nn.ReLU(feat_type_out)
```

Hence, we can construct the steerable convolution layer just by passing the input and output field types, line 7. This layer also supports most of `torch.Conv2d`'s parameters, e.g. `kernel_size`, `padding` or `stride`. Because the regular representation acts with permutation matrices, it supports point-wise non-linearities like *ReLU*. Therefore, in line 8, we define a *ReLU* layer which can act on the output of the convolution layer. Note that this layer does not change the field type, so it is not necessary to specify an output type.

```
9  x = torch.randn(16, 3, 32, 32)
10 x = nn.GeometricTensor(x, feat_type_in)
11
12 y = relu(conv(x))
```

In line 9, we construct an example of input features as a `torch.Tensor` and in line 10 wrap it in a `GeometricTensor`. Note that we associated it with the input type of the convolution layer defined earlier. Finally, we can compose the layers by applying them sequentially on the input tensor as usual, line 11.

5 Experiments

> *In theory, there is no difference between theory and practice; but in practice, there is.*
>
> — **Anonymous**

The theory described in the previous chapters supports a wide range of models, allowing for many different groups, representations and non-linearities; see Sec. 3.6. As no specific choice is preferred, in Sec. 5.1, we perform a broad experimental study comparing many different architectural designs. In order to ensure a fair comparison of the models, experiments are carried out on three variants of the MNIST dataset with different types of symmetries: the classical dataset with untransformed digits, a version where digits are randomly rotated and one with both rotated and reflected digits. The symmetries of the three datasets are respectively $\{e\}$, $\mathrm{SO}(2)$ and $\mathrm{O}(2)$.

In Sec. 5.2, we compare *local* equivariance versus *global invariance*. In particular, we find that enforcing higher levels of invariance can sometimes be a wrong bias and result in a loss of accuracy. Therefore, we try to bypass this problem by exploiting local symmetries through the use of group restriction. To verify the hypothesis that equivariant networks exhibit higher data efficiency, we compare their convergence speed in Sec. 5.3. Furthermore, in Sec. 5.4, we design two new equivariant architectures that surpass the current state-of-the-art on the MNIST rot dataset.

In Sec. 5.5 and 5.6, we use the observations from the previous experiments to design equivariant versions of some popular image classification models and evaluate them on the CIFAR-10, CIFAR-100 and STL-10 datasets. These three datasets do not show strong global rotational symmetries but, rather, consist of natural images with preferred orientations. Although a classical globally-equivariant design might not be optimal, these datasets can show many local symmetries which could be exploited by reducing the degree of equivariance at different depths of the models (see also Sec. 3.7). Hence, they represent a suitable test to understand if our design principle is beneficial.

The code for our experiments will be uploaded at the following URL: `https://github.com/gabri95/e2cnn_experiments`.

5.1 Model benchmarking on transformed MNIST datasets

In this section, we analyze the relative performances of several equivariant architectures supported by the framework here described. Models are compared on three versions of MNIST associated with different global symmetries:

- $\{e\}$: **MNIST-12k**
- $\mathrm{SO}(2)$: rotated MNIST (also **MNIST-rot**): this dataset was originally built from MNIST 12K by rotating each digit by a random angle.
- $\mathrm{O}(2)$: **MNIST-**$\mathrm{O}(2)$: we generate it from MNIST-rot by randomly mirroring each digit in the training set and extending the test with a mirrored version of it.

All datasets contain 12K training samples (split in $10\text{K} + 2\text{K}$ between training and validation) and 50K test images (100K in MNIST-$\mathrm{O}(2)$). The different datasets help analyze the effects of different equivariance groups H in the presence of different global symmetries. For a fair comparison between models with smaller equivariance groups and because it would be used in a real application, during training we augment images with random transformations among those in the dataset used.

We report the performances of 57 different models on these three datasets in Tab. 5.1. Each model is identified by the entries in the first four columns, i.e., equivariance group H, group representations, non-linearities and invariant maps. We cite works that previously proposed a similar model design in the fifth column. For each experiment, we report the mean and standard deviation of the test errors computed over (at least) 6 runs. All architectures are built from the base one in Tab. E.1 using the group, representations, non-linearities and the invariant map specified in each row and adapting the width of each layer to maintain the number of parameters of the models (approximatively) constant. As a result, different models can have a different number of channels, depending on the particular group (and representation) chosen. In order to build models that are globally invariant to their equivariance group, all architectures include an *invariant map* layer before the fully connected layers. This module maps the feature fields to scalar fields and is followed by spatial pooling, generating an invariant feature vector. We try to maintain the size of this invariant feature vector approximatively fixed by scaling the width of the output of the previous convolution according to the particular invariant map considered.

We will now analyze the results reported in Tab. 5.1. More details on the training procedure are in Appendix E.1.

	group	feature types (representation)		non-linearity	invariant map	citation	MNIST O(2)	MNIST rot	MNIST 12k
1	$\{e\}$	(conventional CNN)		ELU	-	-	5.53 ± 0.20	2.87 ± 0.09	0.91 ± 0.06
2	C_1					[45, 2]	5.19 ± 0.08	2.48 ± 0.13	0.82 ± 0.01
3	C_2					[45, 2]	3.29 ± 0.07	1.32 ± 0.02	0.87 ± 0.04
4	C_3					-	2.87 ± 0.04	1.19 ± 0.06	0.80 ± 0.03
5	C_4					[12, 13, 45, 2, 18]	2.40 ± 0.05	1.02 ± 0.03	0.99 ± 0.03
6	C_6	regular	ρ_{reg}	ELU	G-pooling	[22]	2.08 ± 0.03	0.89 ± 0.03	0.84 ± 0.02
7	C_8					[45, 2]	1.96 ± 0.04	0.84 ± 0.02	0.89 ± 0.03
8	C_{12}					[45]	1.95 ± 0.07	0.80 ± 0.03	0.89 ± 0.03
9	C_{16}					[45, 2]	1.93 ± 0.04	0.82 ± 0.02	0.95 ± 0.04
10	C_{20}					[45]	1.95 ± 0.05	0.83 ± 0.05	0.94 ± 0.06
11	C_4		$5\rho_{\text{reg}} \oplus 2\rho_{\text{quot}}^{C_4/C_2} \oplus 2\psi_0$			[13]	2.43 ± 0.05	1.03 ± 0.05	1.01 ± 0.03
12	C_8		$5\rho_{\text{reg}} \oplus 2\rho_{\text{quot}}^{C_8/C_2} \oplus 2\rho_{\text{quot}}^{C_8/C_4} \oplus 2\psi_0$			-	2.03 ± 0.05	0.84 ± 0.05	0.91 ± 0.02
13	C_{12}	quotient	$5\rho_{\text{reg}} \oplus 2\rho_{\text{quot}}^{C_{12}/C_2} \oplus 2\rho_{\text{quot}}^{C_{12}/C_4} \oplus 3\psi_0$			-	2.04 ± 0.04	0.81 ± 0.02	0.95 ± 0.02
14	C_{16}		$5\rho_{\text{reg}} \oplus 2\rho_{\text{quot}}^{C_{16}/C_2} \oplus 2\rho_{\text{quot}}^{C_{16}/C_4} \oplus 4\psi_0$			-	2.00 ± 0.01	0.86 ± 0.04	0.98 ± 0.04
15	C_{20}		$5\rho_{\text{reg}} \oplus 2\rho_{\text{quot}}^{C_{20}/C_2} \oplus 2\rho_{\text{quot}}^{C_{20}/C_4} \oplus 5\psi_0$			-	2.01 ± 0.05	0.83 ± 0.03	0.96 ± 0.04
16		regular/scalar	$\psi_0 \xrightarrow{\text{conv}} \rho_{\text{reg}} \xrightarrow{G\text{-pool}} \psi_0$	ELU, G-pooling		[12, 31]	2.02 ± 0.02	0.90 ± 0.03	0.93 ± 0.04
17	C_{16}	regular/vector	$\psi_1 \xrightarrow{\text{conv}} \rho_{\text{reg}} \xrightarrow{\text{vector pool}} \psi_1$	vector field		[30, 29]	2.12 ± 0.02	1.07 ± 0.03	0.78 ± 0.03
18		mixed vector	$\rho_{\text{reg}} \oplus \psi_1 \xrightarrow{\text{conv}} 2\rho_{\text{reg}} \xrightarrow[\text{pool}]{\text{vector}} \rho_{\text{reg}} \oplus \psi_1$	ELU, vector field		-	1.87 ± 0.03	0.83 ± 0.02	0.63 ± 0.02
19	D_1					-	3.40 ± 0.07	3.44 ± 0.10	0.98 ± 0.03
20	D_2					-	2.42 ± 0.07	2.39 ± 0.04	1.05 ± 0.03
21	D_3					-	2.17 ± 0.06	2.15 ± 0.05	0.94 ± 0.02
22	D_4					[12, 13, 43]	1.88 ± 0.04	1.87 ± 0.04	1.69 ± 0.03
23	D_6	regular	ρ_{reg}	ELU	G-pooling	[22]	1.77 ± 0.06	1.77 ± 0.04	1.00 ± 0.03
24	D_8					-	1.68 ± 0.06	1.73 ± 0.03	1.64 ± 0.02
25	D_{12}					-	1.66 ± 0.05	1.65 ± 0.05	1.67 ± 0.01
26	D_{16}					-	1.62 ± 0.04	1.65 ± 0.02	1.68 ± 0.04
27	D_{20}					-	1.64 ± 0.06	1.62 ± 0.05	1.69 ± 0.03
28	D_{16}	regular/scalar	$\psi_{0,0} \xrightarrow{\text{conv}} \rho_{\text{reg}} \xrightarrow{G\text{-pool}} \psi_{0,0}$	ELU, G-pooling		-	1.92 ± 0.03	1.88 ± 0.07	1.74 ± 0.04
29	SO(2)	irreps ≤ 1	$\bigoplus_{i=0}^{1} \psi_i$	ELU, norm-ReLU	conv2triv	-	2.98 ± 0.04	1.38 ± 0.09	1.29 ± 0.05
30		irreps ≤ 3	$\bigoplus_{i=0}^{3} \psi_i$			-	3.02 ± 0.18	1.38 ± 0.09	1.27 ± 0.03
31		irreps ≤ 5	$\bigoplus_{i=0}^{5} \psi_i$			-	3.24 ± 0.05	1.44 ± 0.10	1.36 ± 0.04
32		irreps ≤ 7	$\bigoplus_{i=0}^{7} \psi_i$			-	3.30 ± 0.11	1.51 ± 0.10	1.40 ± 0.07
33		$\mathbb{C}$-irreps ≤ 1	$\bigoplus_{i=0}^{1} \psi_i^{C_m}$			[48]	3.39 ± 0.10	1.47 ± 0.06	1.42 ± 0.04
34		$\mathbb{C}$-irreps ≤ 3	$\bigoplus_{i=0}^{3} \psi_i^{C_m}$			[48]	3.48 ± 0.16	1.51 ± 0.05	1.53 ± 0.07
35		$\mathbb{C}$-irreps ≤ 5	$\bigoplus_{i=0}^{5} \psi_i^{C_m}$			-	3.59 ± 0.08	1.59 ± 0.05	1.55 ± 0.06
36		$\mathbb{C}$-irreps ≤ 7	$\bigoplus_{i=0}^{7} \psi_i^{C_m}$			-	3.64 ± 0.12	1.61 ± 0.06	1.62 ± 0.03
37				ELU, squash		-	3.10 ± 0.09	1.41 ± 0.04	1.46 ± 0.05
38				ELU, norm-ReLU		-	3.23 ± 0.08	1.38 ± 0.08	1.33 ± 0.03
39				ELU, shared norm-ReLU	norm	-	2.88 ± 0.11	1.15 ± 0.06	1.18 ± 0.03
40		irreps ≤ 3	$\bigoplus_{i=0}^{3} \psi_i$	shared norm-ReLU		-	3.61 ± 0.09	1.57 ± 0.05	1.88 ± 0.05
41				ELU, gate	conv2triv	-	2.37 ± 0.06	1.09 ± 0.03	1.10 ± 0.02
42				ELU, shared gate		-	2.33 ± 0.06	1.11 ± 0.03	1.12 ± 0.04
43				ELU, gate	norm	-	2.23 ± 0.09	1.04 ± 0.04	1.05 ± 0.06
44				ELU, shared gate		-	2.20 ± 0.06	1.01 ± 0.03	1.03 ± 0.03
45	O(2)	irreps $= 0$	$\psi_{0,0}$	ELU	-	-	5.46 ± 0.46	5.21 ± 0.29	3.98 ± 0.04
46		irreps ≤ 1	$\psi_{0,0} \oplus \psi_{1,0} \oplus 2\psi_{1,1}$	ELU, norm-ReLU	O(2)-conv2triv	-	3.31 ± 0.17	3.37 ± 0.18	3.05 ± 0.09
47		irreps ≤ 3	$\psi_{0,0} \oplus \psi_{1,0} \bigoplus_{i=1}^{3} 2\psi_{1,i}$			-	3.42 ± 0.03	3.41 ± 0.10	3.86 ± 0.09
48		irreps ≤ 5	$\psi_{0,0} \oplus \psi_{1,0} \bigoplus_{i=1}^{5} 2\psi_{1,i}$			-	3.59 ± 0.13	3.78 ± 0.31	4.17 ± 0.15
49		irreps ≤ 7	$\psi_{0,0} \oplus \psi_{1,0} \bigoplus_{i=1}^{7} 2\psi_{1,i}$			-	3.84 ± 0.25	3.90 ± 0.18	4.57 ± 0.27
50		Ind-irreps ≤ 1	$\text{Ind}\, \psi_0^{\text{SO}(2)} \oplus \text{Ind}\, \psi_1^{\text{SO}(2)}$	ELU, Ind norm-ReLU	Ind-conv2triv	-	2.72 ± 0.05	2.70 ± 0.11	2.39 ± 0.07
51		Ind-irreps ≤ 3	$\text{Ind}\, \psi_0^{\text{SO}(2)} \bigoplus_{i=1}^{3} \text{Ind}\, \psi_i^{\text{SO}(2)}$			-	2.66 ± 0.07	2.65 ± 0.12	2.25 ± 0.06
52		Ind-irreps ≤ 5	$\text{Ind}\, \psi_0^{\text{SO}(2)} \bigoplus_{i=1}^{5} \text{Ind}\, \psi_i^{\text{SO}(2)}$			-	2.71 ± 0.11	2.84 ± 0.10	2.39 ± 0.09
53		Ind-irreps ≤ 7	$\text{Ind}\, \psi_0^{\text{SO}(2)} \bigoplus_{i=1}^{7} \text{Ind}\, \psi_i^{\text{SO}(2)}$			-	2.80 ± 0.12	2.85 ± 0.06	2.25 ± 0.08
54		irreps ≤ 3	$\psi_{0,0} \oplus \psi_{1,0} \bigoplus_{i=1}^{3} 2\psi_{1,i}$	ELU, gate	O(2)-conv2triv	-	2.39 ± 0.05	2.38 ± 0.07	2.28 ± 0.07
55					norm	-	2.21 ± 0.09	2.24 ± 0.06	2.15 ± 0.03
56		Ind-irreps ≤ 3	$\text{Ind}\, \psi_0^{\text{SO}(2)} \bigoplus_{i=1}^{3} \text{Ind}\, \psi_i^{\text{SO}(2)}$	ELU, Ind gate	Ind-conv2triv	-	2.13 ± 0.04	2.09 ± 0.05	2.05 ± 0.05
57					Ind-norm	-	1.96 ± 0.06	1.95 ± 0.05	1.85 ± 0.07

Tab. 5.1.: Benchmarking of H-steerable CNNs on MNIST variants using different groups H, feature types (representations), non-linearities and final invariant maps. The feature types in the second column only indicate the relative frequency of different individual field types and the actual multiplicities of each field type is different for each layer. The fifth column credits previous works which employed a similar model design. See also Sec. 3.6, 5.1 and Appendix D and F.

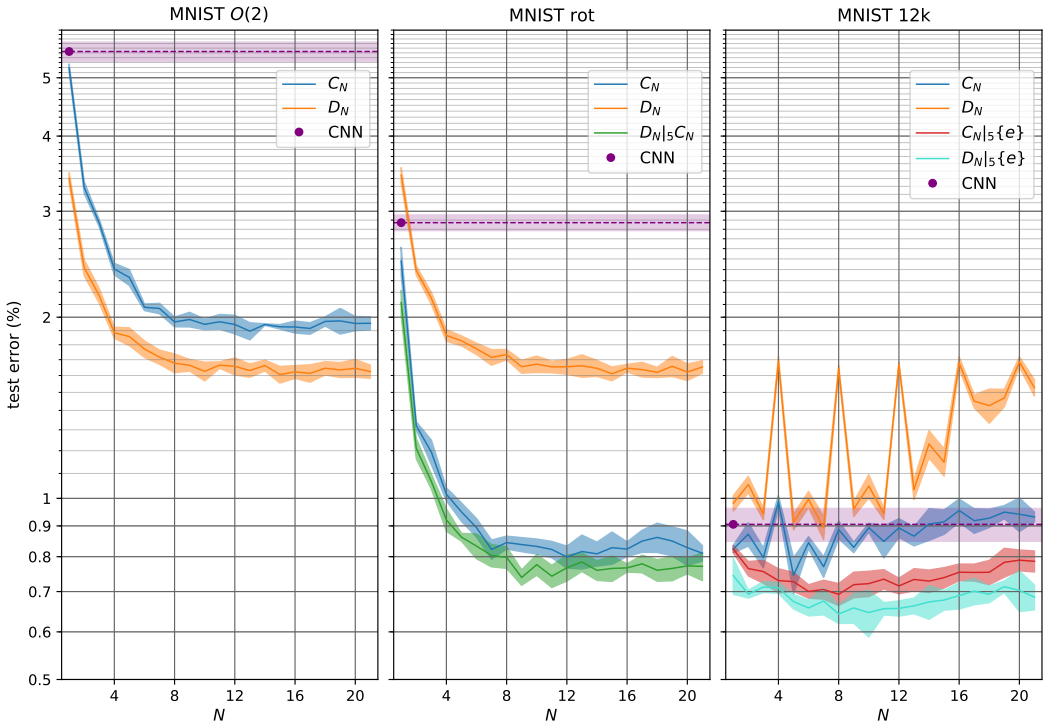


Fig. 5.1.: Test error of C_N and D_N regular steerable CNNs for different rotation orders N on all the three MNIST datasets.
• *Left:* All equivariant models improve upon the non-equivariant CNN baseline on MNIST $\mathrm{O}(2)$, with D_N models achieving best results.
• *Middle:* The baseline CNN and the C_N models obtain lower errors than before, because of the lower complexity of this dataset. Because the D_N models are invariant to reflections, the previous MNIST $\mathrm{O}(2)$ dataset and the current MNIST rot one are indistinguishable. Through restriction, the $\mathrm{D}_N|_5\mathrm{C}_N$ model can exploit local reflectional symmetries being only globally rotation invariant. This results in the best performance.
• *Right:* Globally invariant C_N and D_N models can't improve over the conventional CNN baseline. Again, the use of group restriction enables to construction of better models ($\mathrm{C}_N|_5\{e\}$ and $\mathrm{D}_N|_5\{e\}$).

Regular models: We first discuss models with *regular* feature fields equivariant to the finite groups C_N and D_N, for different number of rotations N (respectively, *rows 2-10 and 19-27* in Tab. 5.1). As discussed in Sec. 3.1 and 3.2, these models are equivalent to the popular group-convolution based architectures (GCNNs) [12, 45]. In the D_N models, we define the action of the reflection element along the vertical axis. Moreover, we adopt ELU [6] and group pooling (Sec. 3.6) as pointwise non-linearity and final invariant map. We find that this configuration generally achieves great performance. This is because such field types are the most expressive (although expensive) ones and allow features to represent arbitrary, unconstrained functions over the group H.

We also visualize the results obtained with regular features in Fig. 5.1. On MNIST $\mathrm{O}(2)$ and MNIST rot, the test error decreases when incrementing the number of rotations N, but the performance saturates between $N = 8$ and $N = 12$. As expected, in

MNIST $\mathrm{O}(2)$, introducing reflection equivariance consistently improves accuracy (compare C_N and D_N models with the same N) as it ensures generalization over reflections, which are part of the transformations of the dataset. Additionally, any group H gives better results than the baseline conventional CNN, which is associated with $H = \{e\}$. Removing reflections produces a simpler dataset, MNIST rot; accordingly, we observe significant improvements in the performance of the CNN baseline and the C_N models. Conversely, the test error of the D_N models is left almost unchanged. This is not surprising as equivariance to reflections implies that inference is independent of reflections of the input. Therefore, these models can not distinguish the two variants of the dataset. In particular, note that the D_1 model, which is equivariant only to vertical reflections (no rotation equivariance), is even worse than the conventional CNN. Indeed, because in this case the dataset does not contain reflected digits, global reflection equivariance only results in a loss of capacity of the network and in less discriminative features before the fully connected classifier. We solve this problem by introducing group restriction to $\mathrm{C}_N \leq \mathrm{D}_N$ before the last convolution layer ($\mathrm{D}_N|_5\mathrm{C}_N$ models in the figure) so that the features produced are not invariant to reflections. We find that these new models slightly improve over the C_N models, supporting our hypothesis that they can exploit the local reflectional symmetries in the images. We will discuss these models in more detail in Sec. 5.2.

Analogously, even in the untransformed dataset MNIST 12k, the pure D_N models have the worst test accuracies. However, we observe that both D_N and C_N models do not show monotonically decreasing trends as in the two previous cases. This is likely explained by the fact that some digits can be approximatively transformed into each other with the symmetries considered. For instance, Fig. 5.2 shows two samples from the original MNIST dataset belonging to different classes. Note that a rotation by $\frac{\pi}{2}$ followed by a vertical reflection transforms the example of 4 into a digit that resembles the example of 7 (first line). Similarly, a vertical reflection followed by a $\frac{\pi}{2}$ rotation transforms the example of 7 into an image that resembles a 4 (second line). These examples would be confused by a D_{4n}-invariant model[1]. When a model is invariant to these transformations, it will find it harder to distinguish such samples. Indeed, Fig. 5.1(right) shows drops in performance when N is a multiple of 2 or 4 or generally when N is large (as H covers approximatively all rotations). Finally, the lowest test errors are achieved combining large equivariance groups H with group

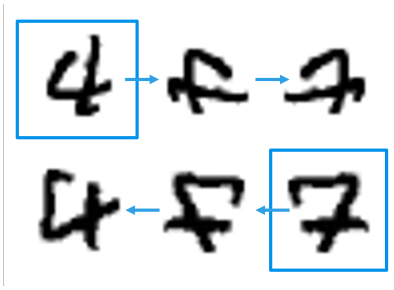


Fig. 5.2.: Two samples of 4 and 7 from the MNIST dataset. A rotation by $\frac{\pi}{2}$ and a reflection can confuse the two examples.

[1] Other examples include **6** and **9** (6 and 9) or $\mathfrak{2}$ and $\mathfrak{Z}$ (2 and 5), which are related by a π rotation.

restriction ($\mathrm{D}_N|_5\{e\}$ and $\mathrm{C}_N|_5\{e\}$ in the figure) - with D_N models outperforming C_N ones-, being able to leverage the local symmetries without losing information about the global orientation.

Quotient models: As introduced in Sec. 3.6, regular representations are a special case of quotient representations. Quotient representations allow more compact features at the cost of enforcing invariance to a subgroup in both the features and the filters. To not excessively reduce the expressiveness of the model, we experiment with feature fields that comprise mixtures of different quotient representations. More details on the chosen quotient representations, together with some insights about the symmetries associated with them, can be found in Appendix D. In particular, we consider C_N-equivariant models in *rows 11-15* of Tab. 5.1. Although the individual feature fields are smaller than the regular ones, by adapting their multiplicities to fix the number of parameters, the resulting quotient models are approximatively as large as the regular ones and, therefore, share their same memory and computational cost. Our experiments do not show any relevant improvement over the regular models. Nevertheless, we point out that the set of existing quotient representations and all their possible combinations is much larger. We do not expect our study to be exhaustive in this sense; a more extensive search in this space might still be necessary.

Group pooling and vector-field non-linearities : Both operations can be used to reduce the size of regular feature fields throughout the network by projecting them, respectively, to scalar or vector fields (see Sec. 3.6). These pooling operations compress the features in the regular fields, which can lead to lower memory and compute requirements. Nevertheless, for a fair comparison, we scale the models to have the same number of parameters. This results in much wider architectures than the ones using regular features (GCNNs) in the first paragraph. For simplicity, in our experiments (*rows 16 and 17*) we only consider C_{16}-equivariant models. As these layers discard part of the information content of the regular features, they result in lower test accuracies with respect to the pure regular models on the two transformed datasets. Contrary to expectations, we find that networks using group pooling - which discards all the orientation information in the features- work better than networks using vector-field non-linearities. This is likely a consequence of the fact that group pooling projects to a smaller space and, therefore, has less trainable parameters per individual field; therefore, when the network is scaled up to match the number of parameters, it results in a wider model. Conversely, on the original MNIST 12k dataset, group pooling performs comparably to regular nets, but the vector-field model achieves the best accuracies among these models. We also design a mixed model with regular features and which applies vector-field non-linearity only to half of its fields at each layer (*row 18*). Although it performs alike with

the pure regular networks on MNIST $\mathrm{O}(2)$ and MNIST rot, it is the best model on MNIST 12k. Finally, we implement a D_{16} equivariant model based on group pooling (row 28). As before, it is always outperformed by the regular models, here even on the untransformed MNIST 12k.

SO(2) irrep models: Because $\mathrm{SO}(2)$ is a continuous group, it has infinitely many elements and it is not possible to build its regular representation explicitly. Instead, to build an $\mathrm{SO}(2)$ equivariant network with finite computational resources, we rely on irreducible representations (see Sec. 2.3 and Sec. 2.7.2). As discussed in Sec. 3.2, scalar and vector fields transform under the irreps $\psi_0^{\mathrm{SO}(2)}$ and $\psi_1^{\mathrm{SO}(2)}$ of $\mathrm{SO}(2)$. We observe that, in practice, none of these models obtain competitive performance with respect to regular models (i.e., GCNNs). This finding is also interesting in a more general context as it suggests that a network design based on irreps, as often done in the 3D setting for $\mathrm{SE}(3) \cong (\mathbb{R}^3, +) \rtimes \mathrm{SO}(3)$-equivariant networks [42, 44, 25, 26, 1], is probably sub-optimal.

We first consider several models with a design similar to the popular Harmonic Networks [48]. In *rows 29-32*, models use features containing all irreps with frequency smaller than a maximum one. We experiment with different maximum frequencies. We choose pointwise ELU on trivial representations (scalar fields) and norm-ReLU on the other ones as non-linearities. Instead of applying an additional invariant map, the last convolution layer maps only to trivial representations to produce invariant features (conv2triv). It emerges that features containing frequencies up to 1 or 3 work equally well, whereas introducing higher frequencies only deteriorates the performance. It is possible that higher frequencies are not necessary to build good representations of the patterns appearing in MNIST, so their use results in a loss of expressiveness of the model by reducing the multiplicity of the more useful low-frequency features. Note that our framework is based on the *real* field $\mathbb{R}$ and, therefore, based on real representations. This differs from the original Harmonic Networks, which employed the *complex* irreps of $\mathrm{SO}(2)$. However, simulating the operations in a complex steerable CNN with real numbers results in an equivalent real steerable CNN with lower-dimensional kernel spaces. A more detailed explanation is provided in Appendix B.3. The models in *rows 33-36* are equivalent to those in *rows 29-32* but only use the kernel subspace which is solution to the kernel constraint associated with complex irreps. All models perform worse than their counterparts with the full real kernel space.

We also consider other variants of irreps-based architectures. For simplicity, we only used irreps with frequency up to 3 in the experiments. In *row 38*, the last convolution does not produce only scalar features but all irreps. Then, the invariant map computes the norm of each non-invariant field producing invariant scalar fields. However, we do not see particular benefits over the previous harmonic networks.

Other variants include different non-linearities and invariant maps. We defer their discussion to Appendix F.

Finally, the models in *rows 41-44* employ gated non-linearities instead of norm-ReLUs. We experiment with two different strategies of applying gates and with two different invariant maps. Model designs based on gated non-linearities achieve the best accuracy among the irreps-based models and obtain the closest performance to the models with regular features. This result is consistent with [44], which first used these non-linearities.

O(2) models: Like $\mathrm{SO}(2)$, the group $\mathrm{O}(2)$ is a continuous group, so we can not use its regular representation. We first implement $\mathrm{O}(2)$-equivariant versions of the $\mathrm{SO}(2)$ models described above. In *rows 46-49*, we use feature fields made of $\mathrm{O}(2)$'s irreps with norm-ReLU non-linearity. While in $\mathrm{SO}(2)$, each irrep occurs only one in the regular representation of the group, the two-dimensional irreps of $\mathrm{O}(2)$ have multiplicity two. Drawing inspiration from this observation, we choose the multiplicity of these irreps in the models' features to be twice that of one-dimensional irreps. We also adapt the final conv2triv invariant map ($\mathrm{O}(2)$-conv2triv). Instead of producing only trivial (invariant) $\psi_{0,0}^{\mathrm{O}(2)}$-fields, the last convolution layer produces half trivial $\psi_{0,0}^{\mathrm{O}(2)}$-flips and half sign-flip $\psi_{1,0}^{\mathrm{O}(2)}$-fields. Sign-flip $\psi_{1,0}^{\mathrm{O}(2)}$ features are scalars whose sign flips under the action of the reflection element in $\mathrm{O}(2)$ but which are invariant to rotations. See Sec. 2.7.2 for more details on the irreps of $\mathrm{O}(2)$. In order to produce completely invariant features, we compute the absolute value of the sign-flip $\psi_{1,0}^{\mathrm{O}(2)}$ fields. As earlier, higher frequency fields result in worse performance. Surprisingly, these models have larger test errors than the corresponding $\mathrm{SO}(2)$ models. It is possible that the kernel space enforced by this choice of fields limits the expressiveness of the model excessively.

A common choice of kernels in geometric deep learning (e.g., used in many graph convolutional networks), are *isotropic kernels*. These kernels only depends on their radial component and, therefore, are invariant to rotations and reflections, i.e. they satisfy $k(gx) = k(x) \quad \forall g \in \mathrm{O}(2)$. We note that these kernels correspond to the solution of the kernel constraint for input and output $\psi_{0,0}^{\mathrm{O}(2)}$-fields, where $\psi_{0,0}^{\mathrm{O}(2)}$ is the trivial representation of $\mathrm{O}(2)$. We implement an $\mathrm{O}(2)$ invariant model using this kernel space in *row 45*: the model's performance is comparable with the baseline CNN on MNIST $\mathrm{O}(2)$, but it is significantly worse on the other two datasets.

To design better $\mathrm{O}(2)$ equivariant models, we build a new architecture by *induction* on the previous $\mathrm{SO}(2)$ architectures. More precisely, we replace each $\psi^{\mathrm{SO}(2)}$-field, with $\psi^{\mathrm{SO}(2)}$ an irrep of $\mathrm{SO}(2)$, with a $\mathrm{Ind}_{\mathrm{SO}(2)}^{\mathrm{O}(2)} \psi^{\mathrm{SO}(2)}$-field. Recall that $\mathrm{Ind}_{\mathrm{SO}(2)}^{\mathrm{O}(2)} \psi^{\mathrm{SO}(2)}$ is a representation of $\mathrm{O}(2)$; see Sec. 2.4 for more details on induced representations. A $\mathrm{Ind}_{\mathrm{SO}(2)}^{\mathrm{O}(2)} \psi^{\mathrm{SO}(2)}$-field can be interpreted as a pair of $\psi^{\mathrm{SO}(2)}$-fields: the action of

a rotation independently transforms each of them with $\psi^{\mathrm{SO}(2)}$ while reflections permute them. Moreover, this construction leads to the same multiplicity for each $\mathrm{O}(2)$ irrep used in the previous $\mathrm{O}(2)$ model. One can verify this by noting that $\mathrm{Ind}_{\mathrm{SO}(2)}^{\mathrm{O}(2)} \psi^{\mathrm{SO}(2)}$ always decomposes in two $\mathrm{O}(2)$ irreps which are isomorphic to ψ when restricted to $\mathrm{SO}(2)$. When ψ is two dimensional, these are two copies of the two dimensional $\mathrm{O}(2)$ irreps; when ψ is the trivial representation, $\mathrm{Ind}_{\mathrm{SO}(2)}^{\mathrm{O}(2)} \psi^{\mathrm{SO}(2)}$ contains both the trivial and the sing-flip representations of $\mathrm{O}(2)$. Therefore, ignoring non-linearities, this model is equivalent to the previous $\mathrm{O}(2)$ model (recall that different bases are irrelevant in linear networks, see Sec. 3.6). Here, however, we employ Ind norm-ReLU non-linearity, which independently applies the usual norm-ReLU to the two ψ-fields contained in a $\mathrm{Ind}\,\psi$-field but sharing the bias between them. Note that this operation does not commute with the change of basis which relates the features of this model to those of the equivalent $\mathrm{O}(2)$'s irreps model, making these two models different. Similarly, we adapt the final invariant map (Ind-conv2triv): we first map to $\mathrm{Ind}_{\mathrm{SO}(2)}^{\mathrm{O}(2)} \psi_0^{\mathrm{SO}(2)}$-fields through convolution and then we pool over the two $\psi^{\mathrm{SO}(2)}$-subfields of each of them. Note that $\mathrm{Ind}_{\mathrm{SO}(2)}^{\mathrm{O}(2)} \psi_0^{\mathrm{SO}(2)}$ can be understood as the regular representation of $(\{\pm 1\}, *) \cong \mathrm{O}(2)/\,\mathrm{SO}(2)$ and, therefore, the last pooling operation is effectively a G-pooling over the reflection subgroup. These models, as reported in *rows 50-53*, achieve significantly lower test errors on all datasets; in particular, they even improve over the $\mathrm{SO}(2)$-equivariant models on MNIST $\mathrm{O}(2)$.

As for $\mathrm{SO}(2)$, architectures that use gated non-linearities achieve the best results among all $\mathrm{O}(2)$ models. In *rows 54-55*, we consider the pure irreps design as initially used with norm-ReLu above. Again, in *rows 56-57* we adapt these architectures using *induced* $\mathrm{SO}(2)$'s irreps (Ind gate). For each $\psi^{\mathrm{SO}(2)}$-subfield in a $\mathrm{Ind}\,\psi^{\mathrm{SO}(2)}$-field, the model learns a different gate. Because a reflection permutes the two subfields, the two gates need to permute accordingly. This is realized by modeling gates as induced scalar fields, i.e., $\mathrm{Ind}_{\mathrm{SO}(2)}^{\mathrm{O}(2)} \psi_0^{\mathrm{SO}(2)}$ fields, instead of $\psi_{0,0}^{\mathrm{O}(2)}$ scalar fields. By combining gated non-linearities and induced features, this design leads to the best $\mathrm{O}(2)$-steerable networks, achieving accuracy close to, although still slightly worse than, those of regular D_N models. We explain in greater detail all induced $\mathrm{O}(2)$ architectures in Appendix F.

5.2 Exploiting local symmetries in MNIST via group restriction

Local patterns in the MNIST dataset can show rotational and reflectional symmetries. This holds for all the three variants of the dataset, even if their global symmetries are different. As noticed in the previous section, while D_N and $\mathrm{O}(2)$-equivariant

restriction depth	MNIST rot		MNIST 12k			
	group	test error (%)	group	test error (%)	group	test error (%)
(0)	C_{16}	0.82 ± 0.02	$\{e\}$	0.82 ± 0.01	$\{e\}$	0.82 ± 0.01
1		0.86 ± 0.05		0.79 ± 0.03		0.80 ± 0.03
2		0.82 ± 0.03		0.74 ± 0.03		0.77 ± 0.03
3	$D_{16} \mid C_{16}$	0.77 ± 0.03	$D_{16} \mid \{e\}$	0.73 ± 0.03	$C_{16} \mid \{e\}$	0.76 ± 0.03
4		0.79 ± 0.03		0.72 ± 0.02		0.77 ± 0.03
5		0.78 ± 0.04		0.68 ± 0.04		0.75 ± 0.02
no restriction	D_{16}	1.65 ± 0.02	D_{16}	1.68 ± 0.04	C_{16}	0.95 ± 0.04

Tab. 5.2.: Group restriction allows the use of larger symmetry groups than the group of global symmetries of the data. Here, we explore the different choices of local symmetries and restriction depth on MNIST rot and MNIST 12k. Delaying restriction to deeper layers generally leads to better performance. Additionally, the use of restriction always improve over the models which are equivariant only to the global symmetries.

models can leverage these local symmetries, they are also globally invariant to them. This implies that the final features used for classification can not carry any information about the orientation or the chirality of the input digit. Obviously, this is not desirable when these symmetries are not present in the data, as this information can be useful to discriminate different digits. Conversely, a model that is equivariant only to the global symmetries does not exploit the local ones, usually a larger group. To solve this issue, in Sec. 3.7 we introduced the *group restriction* operation.

We experiment with different models on MNIST rot and MNIST 12k, which are locally equivariant to either D_N or C_N. Thanks to restriction, the features produced by all models are equivariant only to the group of symmetries of the dataset, which is C_N for MNIST rot and $\{e\}$ (i.e., no equivariance) for MNIST 12k. Because restriction can be introduced at any layer, we study the effect of applying it at different depths. Due to the hierarchical structure of CNNs, the field of view of a neuron in a feature map grows with the depth of the network; as a result, the depth where the restriction is applied is strictly related to the maximum scale at which the local symmetries still hold.

The results of our experiments are summarized in Tab. 5.2. We observe that preserving the local equivariance for more layers generally improves the performance of the model. In particular, it is always optimal to exploit the local symmetries through restriction than considering a model only equivariant to the global ones (compare the row *(0)* with the others in the table). Besides, all models using restriction largely improve over the corresponding globally invariant models (*no restriction* row in the table). In Tab. 5.2, we only reported the results obtained with $N = 16$ but Fig. 5.1 summarizes our experiments for different values of N, when restricting after the 5-th convolution layer ($D_N|_5 C_N$, $D_N|_5\{e\}$ and $C_N|_5\{e\}$ curves). The curves in Fig. 5.1 support our previous observations.

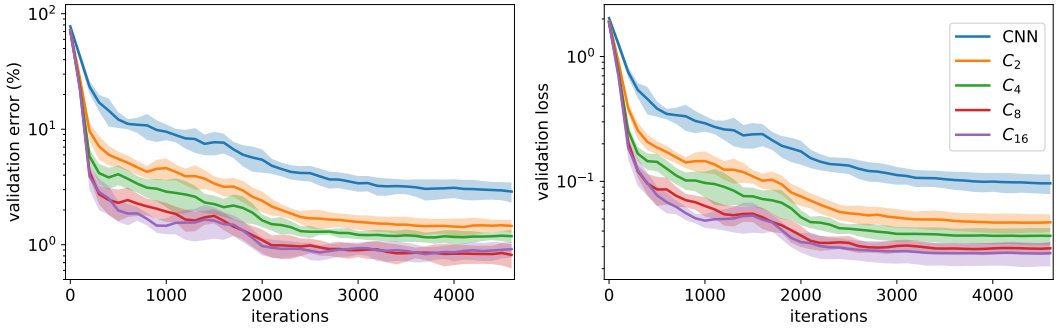


Fig. 5.3.: Validation error and loss registered while training a conventional CNN and C_N-equivariant models on MNIST rot. Larger equivariance groups result in significantly faster converge.

5.3 On the convergence of Steerable CNNs

While conducting our experiments, we noted that equivariant networks converge more rapidly than conventional CNNs. In Fig. 5.3, we illustrate this behavior for the regular C_N models trained on MNIST rot discussed in Sec. 5.1 We observe that larger groups (larger N) always minimize the validation metric faster, although the benefit shown by increasing the group size quickly vanishes after $N = 8$. Note that, because all models are adapted such that the number of parameters is approximatively the same, this result can not be explained only by the lower dimensional kernel space associated with larger groups. Instead, a more plausible answer is the stronger generalization capability of equivariant networks. Indeed, a G-equivariant model reduces the learning task on the smaller quotient space defined by the action of the group G on the data, that is on the space containing *equivalence classes* of samples defined by the equivalence relation $\boldsymbol{x} \sim \boldsymbol{x}' \Leftrightarrow \exists\, g \in G$ s.t. $\boldsymbol{x} = g.\boldsymbol{x}'$. When $G = \{e\}$, i.e., the model is a simple MLP, every sample is different and the network needs to learn each of them independently. When the data points are images, the use of a $G = (\mathbb{R}^2, +)$ equivariant network, i.e., a conventional CNN, enables the generalization over classes of translated images. H-steerable CNNs extend the equivalence classes to $G = (\mathbb{R}^2, +) \rtimes H$, further improving generalization. In other words, observing a single data point $\boldsymbol{x}$ is equivalent to observe its whole orbit $G.\boldsymbol{x} = \{g.x | g \in G\}$. This suggests that equivariance enhances the data efficiency of the learning algorithm and justifies the reduced training time observed in the experiments. This is discussed in more detail in Section 2 of [45].

5.4 Competitive MNIST rot experiments

At last, we try to replicate and improve the current state-of-the-art results on MNIST rot. The current best method from [45] utilizes a C_{16} equivariant group

convolutional network. To prevent aliasing issues when rotating the filters, the authors parametrize the filters in terms of a harmonic angular basis, guaranteeing steerability. The resulting steerable basis can be related to our kernel basis associate with the regular representation of C_{16} by a simple change of basis. This allows us to easily replicate this model. The architecture is very similar to the one used in the benchmark experiments in Sec. 5.1 but uses larger filters, wider layers and a deeper fully connected classifier. The details of the model can be found in Tab. E.2. In Tab. 5.3, we report the current leader-board on MNIST rot. In *row 6*, we successfully reproduce the results from [45]. To improve this model, we experiment with quotient representations and with restriction. In *row 8*, we employ a mixture of quotient fields instead of using pure regular features, as done in Sec. 5.1. This change slightly increases the test accuracy. The introduction of D_{16}-equivariance with restriction to C_{16} after the fifth convolution yields a more substantial improvement and defines a new state-of-the-art in *row 9*.

	model	group	representation	test error (%)
1	[12]	C_4	regular/scalar	3.21 ± 0.0012
2	[12]	C_4	regular	2.28 ± 0.0004
3	[48]	$\mathrm{SO}(2)$	irreducible	1.69
4	[28]	-	-	1.2
5	[30]	C_{17}	regular/vector	1.09
6	Ours	C_{16}	regular	0.716 ± 0.028
7	[45]	C_{16}	regular	0.714 ± 0.022
8	Ours	C_{16}	quotient	0.705 ± 0.025
9	Ours	$\mathrm{D}_{16}\vert_5\mathrm{C}_{16}$	regular	0.682 ± 0.022

Tab. 5.3.: Competitive experiments and current leader-board on MNIST rot.

5.5 CIFAR

As discussed in Sec. 3.7, natural images are commonly characterized by attributes that are invariant to global translations and reflections. In this section and the next one, we experimentally validate our hypothesis that equivariance to local symmetries can be a useful prior for computer vision models when working with natural images.

We first experiment with the CIFAR-10 and CIFAR-100 datasets. We consider WideResNet [49] as a baseline since it is a popular image-classification model and, at the time of writing, one the simplest, yet best performing architectures on these datasets. In particular, we use the WideResnet 28/10 version, which has 28 convolution layers and a widening factor of 10 (see [49] for more details). Therefore, we design a number of D_N and C_N-equivariant variants of WideResNet using our steerable kernels. All the results are reported in Tab. 5.4.

The usual WideResNet architecture is composed of three main blocks, separated by down-sampling layers (actually, strided convolutions), and a final global average

model	groups / citation	AutAugment [14]	CIFAR-10	CIFAR-100
wrn28/10	[49]	✗	3.87	18.80
wrn28/10	$D_1\, D_1\, D_1$	✗	3.36 ± 0.08	17.97 ± 0.11
wrn28/10*	$D_8\, D_4\, D_1$	✗	3.28 ± 0.10	17.42 ± 0.33
wrn28/10	$C_8\, C_4\, C_1$	✗	3.20 ± 0.04	16.47 ± 0.22
wrn28/10	$D_8\, D_4\, D_1$	✗	3.13 ± 0.17	16.76 ± 0.40
wrn28/10	$D_8\, D_4\, D_4$	✗	2.91 ± 0.13	16.22 ± 0.31
wrn28/10	[14]	✓	2.6 ± 0.1	17.1 ± 0.3
wrn28/10*	$D_8\, D_4\, D_1$	✓	2.39 ± 0.11	15.55 ± 0.13
wrn28/10	$D_8\, D_4\, D_1$	✓	2.05 ± 0.03	14.30 ± 0.09

Tab. 5.4.: Test errors on CIFAR-10 and CIFAR-100.
The second column indicates the equivariance group of each of the three main blocks of WideRestNet.
"*" specifies the equivariant model has the same size (number of channels) of the original conventional CNN, i.e. the width has not been scaled up to match the number of parameters.

pooling followed by a linear classifier. Note that each of the blocks works at a different resolution of the image. We leverage this structural configuration in our equivariant design by choosing different symmetries in each block. Group restriction is performed before the first strided convolution at the beginning of each block. The second column in Tab. 5.4 shows the group chosen in the three blocks for our equivariant models or the source of the test errors reported for the conventional models.

Because of their superior performance in the MNIST experiments, in all our architectures, we utilize only regular features in the inner layers. In order to produce invariant features for the final linear classifier, the last convolution layer outputs only trivial (i.e., invariant) features (conv2triv in Sec. 5.1).

To ensure a fair comparison with conventional CNNs, as before, we adapt the equivariant model's width to maintain the number of parameters in each layer approximatively constant. As a result, models that are equivariant to larger groups are also wider and, therefore, computationally more expensive. For this reason, we also compare with thinner equivariant models, denoted with a *, where the widths of the layers have not been scaled. These models have similar computational costs of the original WideResNet.

Furthermore, when considering $N = 8$ rotations, we use 5×5 filters instead of the usual $3x3$. This helps reduce the discretization artifacts produced by rotating a 3×3 filter by 45 degrees and allows us to sample higher frequencies. We used the same training procedure in [49]; see Appendix E.3 for more details. We emphasize that *no further hyperparameter tuning* was performed.

The $\mathrm{D}_1\,\mathrm{D}_1\,\mathrm{D}_1$ model is invariant to global vertical reflections. Its superior performance over the conventional baseline supports our assumption that natural images are globally symmetric to reflections. On the other hand, consider the $\mathrm{C}_8\,\mathrm{C}_4\,\mathrm{C}_1$ model. While it incorporates only (local) rotation but not reflection equivariance, it outperforms the last reflection equivariant model. This encouraging result suggests that exploiting the local rotational symmetries can be beneficial. Therefore, we combine these two orthogonal designs in the $\mathrm{D}_8\,\mathrm{D}_4\,\mathrm{D}_1$ model. We find that this architecture improves over both previous ones on CIFAR-10 but, surprisingly, the $\mathrm{C}_8\,\mathrm{C}_4\,\mathrm{C}_1$ model is still marginally better on CIFAR-100. We hypothesize this might be a consequence of the higher dimensionality of D_N feature fields with respect to the C_N ones, which consequently reduces the number of independent fields, even when the width of the model is scaled up. The highest accuracies are found using $\mathrm{D}_8\,\mathrm{D}_4\,\mathrm{D}_4$-equivariance. This model is also equivariant to global $\frac{\pi}{2}$ rotations, which means that rotational symmetries tend to appear also a larger scale. This is consistent with the results found in [12]. It is particularly interesting that even the smaller wrn28/10* $\mathrm{D}_8\,\mathrm{D}_4\,\mathrm{D}_1$ model largely outperforms the conventional baseline while maintaining approximatively the same computational cost.

Finally, we study the effect of equivariance in a context where a strong data-augmentation strategy already yields considerable gains. To do this, we re-train both $\mathrm{D}_8\,\mathrm{D}_4\,\mathrm{D}_1$ models using the *AutoAugment* (AA) policy from [14]. Like in the previous experiments, both the scaled-up architecture and the one with the original width achieve significantly better results than the conventional baseline.

5.6 STL-10

Although they are common datasets for evaluating deep learning models, CIFAR-10 and CIFAR-100 only include low-resolution images. To verify if our hypotheses hold in more general settings and if our observations generalize to high-resolution images, we repeat similar experiments on the STL-10 dataset [7].

STL-10 contains 5000 labeled training images and 8000 test images, plus 100.000 unlabeled images. All images have 96×96 resolutions. This dataset was initially designed to evaluate semi-supervised learning algorithms and, therefore, contains mostly unlabeled im-

model	groups / citations	#params	test error (%)
wrn16/8	[16]	11M	$12.74_{\pm 0.23}$
wrn16/8*	$\mathrm{D}_1\,\mathrm{D}_1\,\mathrm{D}_1$	5M	$11.05_{\pm 0.45}$
wrn16/8	$\mathrm{D}_1\,\mathrm{D}_1\,\mathrm{D}_1$	10M	$11.17_{\pm 0.60}$
wrn16/8*	$\mathrm{D}_8\,\mathrm{D}_4\,\mathrm{D}_1$	4.2M	$10.57_{\pm 0.70}$
wrn16/8	$\mathrm{D}_8\,\mathrm{D}_4\,\mathrm{D}_1$	12M	$9.80_{\pm 0.40}$

Tab. 5.5.: Test errors of different equivariant models on the STL-10 dataset. "*" indicates that the equivariant model has the same size (number of channels) of the original conventional CNN.

ages. However, its labeled training images are also often used to study supervised methods in low data regime [16]. Here, we replicate the experiments in [16], where the authors train a WideResNet 16/8 on STL-10 using *CutOut* augmentation. At the time of writing, this model holds the state-of-the-art on STL-10 among the methods using supervised learning. We adapt the conventional WideResNet architecture as done in the previous section on the CIFAR experiments. Again, we only replace conventional convolution layers with our D_N-steerable convolution, but we use the same training procedure and do not perform further hyperparameters tuning. In Tab. 5.5, we experiment with four variants of WideResNet. We test a reflection equivariant model ($\mathrm{D}_1\,\mathrm{D}_1\,\mathrm{D}_1$) and a locally rotation- and globally reflection- equivariant model ($\mathrm{D}_8\,\mathrm{D}_4\,\mathrm{D}_1$). In both cases, we consider both a scaled-up architecture, which has the same number of parameters but a larger number of channels and a smaller version (indicated by *) with the same size of the original WideResNet.

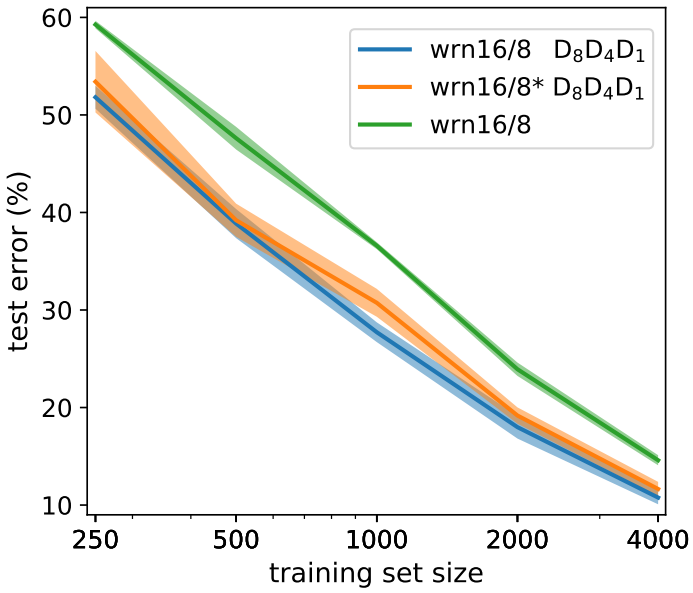


Fig. 5.4.: Data ablation study on STL-10. The equivariant models yield significantly improved results on all dataset sizes.

All four equivariant models achieve consistently higher accuracies than the conventional baseline. As one could expect, the scaled-up $\mathrm{D}_8\,\mathrm{D}_4\,\mathrm{D}_1$ model outperforms the corresponding small architecture (marked with *). However, we point out that the largest gain is accomplished by the smallest model, i.e., by simply enforcing equivariance, while the improvement obtained by further increasing the width of the model is minor. Interestingly, the small $\mathrm{D}_8\,\mathrm{D}_4\,\mathrm{D}_1$ model outperforms even the wider $\mathrm{D}_1\,\mathrm{D}_1\,\mathrm{D}_1$ model, reinforcing again the idea that using higher levels of equivariance is beneficial. Additionally, with respect to the experiments in Sec. 5.5 on the CIFAR datasets, we observe a more substantial boost in performance on STL-10 when we introduce rotation equivariance. This result is probably a consequence of the higher resolution of the images, which favor the presence of more local patterns at more distinct orientations. Finally, we perform a simple data ablation study to verify the data efficiency of equivariance. In Fig. 5.4, we report the test errors of the two $\mathrm{D}_8\,\mathrm{D}_4\,\mathrm{D}_1$ models and the conventional baseline trained on different subsets of the training set. These experiments show that the gain obtained from equivari-

ance increases on smaller datasets. For mode details on the training setting, see Appendix E.4.

5.7 Discussion of the Results

In our experiment, we first extensively compared a large number of equivariant networks designs on variants of the MNIST dataset and established the higher performance of those based on regular feature fields, i.e. group convolution based architectures. We also found improved results by leveraging local symmetries through group restriction. Successively, with these insights, we adapted well established conventional CNN architectures using our steerable convolution layers and evaluated them on three datasets: CIFAR-10, CIFAR-100 and STL-10. Even when steerable convolution was used as a simple drop in replacement for the conventional one and no hyper-parameters tuning was performed, all the models yielded considerable gains in accuracy with respect to the baselines. This behavior was observed in different settings, both with low and high resolution images and even in combination with powerful data augmentation policies.

6 Conclusion

> *The worst thing you can do is to completely solve a problem.*
>
> **— Dan Kleitman**

To build rotation- and reflection-equivariance inside deep learning models, many recent works have proposed alternative architecture designs. In this manuscript, we described a general framework for $\mathrm{E}(2)$ equivariance based on steerable CNNs, unifying most of the previously existing methods.

Reinterpreting these models from a representation-theoretic point of view, we relate many design choices solely to the choice of group representations and non-linearities in the intermediate layers of the networks. Any linear equivariant operation needs to satisfy a precise constraint dictated by its input and output representations. By analytically solving the constraint defined on the convolution kernels of a steerable CNN for any combination of representations of $\mathrm{O}(2)$ (or subgroups), we could replicate many previously proposed models and design entirely new ones. Therefore, we perform an extensive experimental study to compare them. We then introduced group restriction as a means to control each layer's equivariance in terms of the scale of its field of view. Finally, we validated our model design on three natural images datasets, namely CIFAR-10, CIFAR-100 and STL-10. By only replacing the conventional convolution in state-of-the-art deep learning architectures with our steerable convolution, we find considerable gains in performance on all datasets, even without further hyper-parameters tuning.

From a computational cost perspective, the construction of the kernel introduces a small overhead during training. However, during evaluation, the kernels do not need to be updated and, therefore, can be computed once and reused at each forward pass. Thus, once trained, a steerable CNN can be converted into a conventional CNN of the same width with no additional cost. However, thanks to the stronger weight sharing, an equivariant design often benefits from wider architectures, which match the number of parameters of a corresponding conventional CNN. Nevertheless, in our experiments, we find that this adjustment is not necessary to observe improved performance with respect to conventional CNNs. Our experiments also show that

steerable CNNs enjoy a faster convergence, suggesting that it is probably possible to train them using fewer resources.

In conclusion, we expect equivariance to eventually become the preferred network design when data shows symmetries on a global scale, e.g., in biomedical imaging, or even only locally, in light of our results on natural images. Thus, we hope our `e2cnn` library[1] will facilitate the adoption of equivariant networks in the scientific community and encourage further research.

6.1 Future Work

Despite the large experimental study performed in this work, further research is still necessary to explore the vast space of architectures described by the steerable CNN framework. Indeed, although regular fields are generally the best choice, they also have the highest memory and computational cost. This makes them less suitable when using large groups but suggests a potential for lower dimensional, although less expressive representations. In particular, the optimal combination of field types describing a model's features is probably task-specific. For instance, ignoring non-linearities and, therefore, changes of bases for simplicity, the multiplicity of $\mathrm{O}(2)$'s irreps in the field type is directly related to frequencies of the patterns that the features can encode. When working with datasets containing low resolution or smooth images, low-frequency feature types may be sufficient; conversely, high-resolution data may require high-frequency features to ensure the model is sufficiently expressive. An automated method to search this space, avoiding combinatorial explosion, might be necessary.

Finally, because of the increasing interest in 3D applications within the Computer Vision community, a prominent direction of future research is extending the current framework to the group of 3D isometries $\mathrm{E}(3)$ and its subgroups.

[1] https://github.com/QUVA-Lab/e2cnn

Bibliography

[1]Brandon Anderson, Truong-Son Hy, and Risi Kondor. “Cormorant: Covariant Molecular Neural Networks”. In: *arXiv preprint arXiv:1906.04015* (2019) (cit. on pp. 51, 74, 133, 134).

[2]Erik J. Bekkers, Maxime W Lafarge, Mitko Veta, et al. “Roto-translation covariant convolutional networks for medical image analysis”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*. 2018 (cit. on pp. 2, 37, 49, 50, 70).

[3]Davide Boscaini, Jonathan Masci, Simone Melzi, et al. “Learning Class-specific Descriptors for Deformable Shapes Using Localized Spectral Convolutional Networks”. In: *Computer Graphics Forum* (2015) (cit. on p. 2).

[4]J. Bruna, W. Zaremba, A. Szlam, and Y. LeCun. “Spectral Networks and Deep Locally Connected Networks on Graphs”. In: *International Conference on Learning Representations (ICLR)*. 2014 (cit. on p. 2).

[5]Joan Bruna and Stéphane Mallat. “Invariant scattering convolution networks”. In: *IEEE transactions on pattern analysis and machine intelligence* 35.8 (2013), pp. 1872–1886 (cit. on p. 2).

[6]Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. “Fast and accurate deep network learning by exponential linear units (ELUs)”. In: *International Conference on Learning Representations (ICLR)*. 2016 (cit. on p. 71).

[7]Adam Coates, Andrew Ng, and Honglak Lee. “An Analysis of Single-Layer Networks in Unsupervised Feature Learning”. In: *International Conference on Artificial Intelligence and Statistics (AISTATS)*. 2011 (cit. on p. 81).

[8]Taco S. Cohen, Mario Geiger, Jonas Köhler, and Max Welling. “Spherical CNNs”. In: *International Conference on Learning Representations (ICLR)*. 2018 (cit. on pp. 2, 49).

[9]Taco S. Cohen, Mario Geiger, and Maurice Weiler. “A General Theory of Equivariant CNNs on Homogeneous Spaces”. In: (2019) (cit. on pp. 1, 2, 35, 41, 42).

[10]Taco S. Cohen, Mario Geiger, and Maurice Weiler. “Intertwiners between Induced Representations (with Applications to the Theory of Equivariant Neural Networks)”. In: *arXiv preprint arXiv:1803.10743* (2018) (cit. on pp. 1, 2, 24, 30, 35, 41, 42).

[11]Taco S. Cohen, Maurice Weiler, Berkay Kicanaoglu, and Max Welling. “Gauge Equivariant Convolutional Networks and the Icosahedral CNN”. In: *International Conference on Machine Learning (ICML)*. 2019 (cit. on pp. 1, 2, 35, 49).

[12]Taco S. Cohen and Max Welling. “Group equivariant convolutional networks”. In: *International Conference on Machine Learning (ICML)*. 2016 (cit. on pp. 2, 35, 37, 49, 50, 56, 60, 61, 70, 71, 79, 81).

[13]Taco S. Cohen and Max Welling. “Steerable CNNs”. In: *International Conference on Learning Representations (ICLR)*. 2017 (cit. on pp. 1, 2, 35, 50, 56, 70).

[14]Ekin D. Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V. Le. “AutoAugment: Learning Augmentation Strategies From Data”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019 (cit. on pp. 80, 81).

[15]C.W. Curtis and I. Reiner. *Representation Theory of Finite Groups and Associative Algebras*. Wiley Classics Library. Wiley, 1988 (cit. on p. 150).

[16]Terrance DeVries and Graham W. Taylor. “Improved Regularization of Convolutional Neural Networks with Cutout”. In: *arXiv preprint arXiv:1708.04552* (2017) (cit. on pp. 81, 82, 143, 144).

[17]Nichita Diaconu and Daniel Worrall. “Learning to Convolve: A Generalized Weight-Tying Approach”. In: *International Conference on Machine Learning (ICML)*. 2019 (cit. on p. 49).

[18]Sander Dieleman, Jeffrey De Fauw, and Koray Kavukcuoglu. “Exploiting Cyclic Symmetry in Convolutional Neural Networks”. In: *International Conference on Machine Learning (ICML)*. 2016 (cit. on pp. 2, 49, 56, 70).

[19]Carlos Esteves, Christine Allen-Blanchette, Ameesh Makadia, and Kostas Daniilidis. “Learning SO(3) Equivariant Representations with Spherical CNNs”. In: *European Conference on Computer Vision (ECCV)*. 2018 (cit. on p. 2).

[20]Gerald B. Folland. “A Course in Abstract Harmonic Analysis”. In: 1995 (cit. on p. 27).

[21]Geoffrey Hinton, Nicholas Frosst, and Sabour Sara. “Matrix capsules with EM routing”. In: *International Conference on Learning Representations (ICLR)*. 2018 (cit. on p. 145).

[22]Emiel Hoogeboom, Jorn W. T. Peters, Taco S. Cohen, and Max Welling. “HexaConv”. In: *International Conference on Learning Representations (ICLR)*. 2018 (cit. on pp. 2, 49, 70).

[23]Chiyu Jiang, Jingwei Huang, Karthik Kashinath, et al. “Spherical CNNs on Unstructured Grids”. In: *International Conference on Learning Representations (ICLR)*. 2019 (cit. on p. 2).

[24]Diederik P Kingma and Jimmy Ba. “Adam: A method for stochastic optimization”. In: *arXiv preprint arXiv:1412.6980* (2014) (cit. on p. 142).

[25]Risi Kondor. “N-body Networks: a Covariant Hierarchical Neural Network Architecture for Learning Atomic Potentials”. In: *arXiv preprint arXiv:1803.01588* (2018) (cit. on pp. 51, 74, 133, 134).

[26]Risi Kondor, Zhen Lin, and Shubhendu Trivedi. “Clebsch–Gordan Nets: A Fully Fourier Space Spherical Convolutional Neural Network”. In: *Conference on Neural Information Processing Systems (NeurIPS)*. 2018 (cit. on pp. 2, 51, 74, 133, 134).

[27]Risi Kondor and Shubhendu Trivedi. “On the generalization of equivariance and convolution in neural networks to the action of compact groups”. In: *International Conference on Machine Learning (ICML)*. 2018 (cit. on pp. 2, 50).

[28]Dmitry Laptev, Nikolay Savinov, Joachim M. Buhmann, and Marc Pollefeys. “TI-Pooling: Transformation-Invariant Pooling for Feature Learning in Convolutional Neural Networks”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2016 (cit. on p. 79).

[29]Diego Marcos, Michele Volpi, Benjamin Kellenberger, and Devis Tuia. “Land cover mapping at very high resolution with rotation equivariant CNNs: Towards small yet accurate models”. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 145 (2018), pp. 96–107 (cit. on p. 70).

[30]Diego Marcos, Michele Volpi, Nikos Komodakis, and Devis Tuia. “Rotation equivariant Vector Field networks”. In: *International Conference on Computer Vision (ICCV)*. 2017 (cit. on pp. 2, 50, 70, 79).

[31]Diego Marcos, Michele Volpi, and Devis Tuia. “Learning rotation invariant convolutional filters for texture classification”. In: *International Conference on Pattern Recognition (ICPR)*. 2016 (cit. on p. 70).

[32]Jonathan Masci, Davide Boscaini, Michael M. Bronstein, and Pierre Vandergheynst. “Geodesic Convolutional Neural Networks on Riemannian Manifolds”. In: *International Conference on Computer Vision Workshop (ICCVW)*. 2015 (cit. on p. 2).

[33]Edouard Oyallon and Stéphane Mallat. “Deep Roto-Translation Scattering for Object Classification”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015 (cit. on pp. 2, 49).

[34]Nathanaël Perraudin, Michaël Defferrard, Tomasz Kacprzak, and Raphael Sgier. “DeepSphere: Efficient Spherical Convolutional Neural Network with HEALPix Sampling for Cosmological Applications”. In: *arXiv:1810.12186 [astro-ph]* (2018). arXiv: 1810.12186 (cit. on p. 2).

[35]Adrien Poulenard and Maks Ovsjanikov. “Multi-Directional Geodesic Neural Networks via Equivariant Convolution”. In: *ACM Transactions on Graphics* (2018) (cit. on p. 2).

[36]Daniel N. Rockmore. “Efficient Computation of Fourier Inversion for Finite Groups”. In: *Journal of the Association for Computing Machinery (ACM)* (1994) (cit. on p. 149).

[37]Sara Sabour, Nicholas Frosst, and Geoffrey E. Hinton. “Dynamic Routing Between Capsules”. In: *Conference on Neural Information Processing Systems (NIPS)*. 2017 (cit. on pp. 40, 49).

[38]Jean-Pierre Serre. “Linear representations of finite groups”. In: (1977) (cit. on pp. 18, 27).

[39]Laurent Sifre and Stéphane Mallat. “Combined scattering for rotation invariant texture analysis.” In: *European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN)*. Vol. 44. 2012, pp. 68–81 (cit. on p. 2).

[40]Laurent Sifre and Stéphane Mallat. “Rigid-motion scattering for texture classification”. In: *arXiv preprint arXiv:1403.1687* (2014) (cit. on pp. 2, 49).

[41]Laurent Sifre and Stéphane Mallat. “Rotation, Scaling and Deformation Invariant Scattering for Texture Discrimination”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)* (2013) (cit. on p. 2).

[42] Nathaniel Thomas, Tess Smidt, Steven M. Kearnes, et al. “Tensor Field Networks: Rotation- and Translation-Equivariant Neural Networks for 3D Point Clouds”. In: *arXiv preprint arXiv:1802.08219* (2018) (cit. on pp. 74, 133, 134).

[43] Bastiaan S. Veeling, Jasper Linmans, Jim Winkens, Taco S. Cohen, and Max Welling. “Rotation Equivariant CNNs for Digital Pathology”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*. 2018 (cit. on p. 70).

[44] Maurice Weiler, Mario Geiger, Max Welling, Wouter Boomsma, and Taco S. Cohen. “3D Steerable CNNs: Learning Rotationally Equivariant Features in Volumetric Data”. In: *Conference on Neural Information Processing Systems (NeurIPS)*. 2018 (cit. on pp. 1–3, 35, 42–44, 47, 49, 51, 56, 74, 75, 133–135).

[45] Maurice Weiler, Fred A. Hamprecht, and Martin Storath. “Learning Steerable Filters for Rotation Equivariant CNNs”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018 (cit. on pp. 2, 37, 47, 49, 50, 56, 70, 71, 78, 79, 141, 142).

[46] Marysia Winkels and Taco S. Cohen. “3D G-CNNs for Pulmonary Nodule Detection”. In: *Conference on Medical Imaging with Deep Learning (MIDL)*. 2018 (cit. on pp. 49, 50).

[47] Daniel E. Worrall and Gabriel J. Brostow. “CubeNet: Equivariance to 3D Rotation and Translation”. In: *European Conference on Computer Vision (ECCV)*. 2018, pp. 585–602 (cit. on pp. 49, 50).

[48] Daniel E. Worrall, Stephan J. Garbin, Daniyar Turmukhambetov, and Gabriel J. Brostow. “Harmonic Networks: Deep Translation and Rotation Equivariance”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017 (cit. on pp. 2, 47, 49, 56, 70, 74, 79, 129–131).

[49] Sergey Zagoruyko and Nikos Komodakis. “Wide Residual Networks”. In: *British Machine Vision Conference (BMVC)*. 2016 (cit. on pp. 79, 80).

[50] Matthew D. Zeiler and Rob Fergus. “Visualizing and understanding convolutional networks”. In: *European Conference on Computer Vision (ECCV)*. Springer, 2014, pp. 818–833 (cit. on p. 52).

A

Equivariant non-linearities in $\mathrm{E}(2)$-steerable CNNs

We have already proved the sufficiency of H-steerable convolution for $G = (\mathbb{R}^2, +) \rtimes H$ equivariance in Sec. 3.3. In this section, we will show the equivariance of the non-linear layers used in the neural networks.

Any non-linear layer Σ considered in this work consists of a non-linear function $\sigma : \mathbb{R}^{c_{\text{in}}} \to \mathbb{R}^{c_{\text{out}}}$ which is applied on each feature vector $f(\boldsymbol{x}) \in \mathbb{R}^{c_{\text{in}}}$ of the input feature field $f : \mathbb{R}^2 \to \mathbb{R}^{c_{\text{in}}}$ individually. In the rest of this section, we will denote the whole layer as Σ and its application on an individual vector as σ, i.e.:

$$\Sigma : f \mapsto \Sigma(f),\ [\Sigma(f)]\,(\boldsymbol{x}) := \sigma\big(f(\boldsymbol{x})\big) \quad .$$

We require Σ to be equivariant with respect to its input and output field types' transformations:

$$\Sigma(\operatorname{Ind} \rho_{\text{in}}(g)\ f) = \operatorname{Ind} \rho_{\text{out}}(g)\ \Sigma(f) \qquad \forall f,\ \forall g \in G$$

By expanding the action of the induced representations as defined in Eq. 3.8, one obtains the following equivalent conditions:

$$\begin{aligned} & \Sigma(\operatorname{Ind} \rho_{\text{in}}(g)\ f) = \operatorname{Ind} \rho_{\text{out}}(g)\ \Sigma(f) && \forall f,\ \forall g \in G \\ \iff \quad & [\Sigma(\operatorname{Ind} \rho_{\text{in}}(g)\ f)]\,(\boldsymbol{x}) = [\operatorname{Ind} \rho_{\text{out}}(g)\ \Sigma(f)]\,(\boldsymbol{x}) && \forall f,\ \forall \boldsymbol{x} \in \mathbb{R}^2,\ \forall g \in G \end{aligned}$$

using the decomposition $g = th \in G = (\mathbb{R}^2, +) \rtimes H$:

$$\begin{aligned} \iff \quad & \sigma\Big(\rho_{\text{in}}(h)\ f(h^{-1}(\boldsymbol{x} - t)\Big) = \rho_{\text{out}}(h)\ \sigma\Big(f(h^{-1}(\boldsymbol{x} - t))\Big) && \forall f,\ \forall \boldsymbol{x} \in \mathbb{R}^2,\ \forall th \in G \\ \iff \quad & \sigma\left(\rho_{\text{in}}(h)\ \boldsymbol{v}\right) = \rho_{\text{out}}(h)\ \sigma\left(\boldsymbol{v}\right) && \forall \boldsymbol{v} \in \mathbb{R}^{c_{\text{in}}},\ \forall h \in H \end{aligned}$$

that is, σ needs to be equivariant with respect to the H-representations ρ_{in} and ρ_{out}.

Therefore, in order to prove the equivariance of a non-linear layer Σ to G, it is necessary and sufficient to show the equivariance of σ to H. As described in Sec. 3.2, a feature field $f : \mathbb{R}^2 \to \mathbb{R}^{c_{\text{in}}}$ can be composed from multiple sub-fields $\{f_i : \mathbb{R}^2 \to \mathbb{R}^{c_i}\}_i$ (with $c_{\text{in}} = \sum_i c_i$) through the *direct sum*, i.e. $f = \bigoplus_i f_i$. Analogously, the field

type ρ_{in} of f is defined as the direct sum $\rho_{\text{in}} = \bigoplus_i \rho_{\text{in}}^i$, where ρ_{in}^i is the field type of the i-th field f_i. In general, the function σ applies an independent (potentially different) non-linearity σ_i to each sub-field f_i. With an abuse of notation, we write $\sigma = \bigoplus_i \sigma_i$ to indicate the function

$$\sigma : f = \bigoplus_i f_i \;\mapsto\; \sigma(f) = \sigma\left(\bigoplus_i f_i\right) := \bigoplus_i \sigma_i(f_i) \quad .$$

Then, if every σ_i is equivariant with respect to its own input and output types' representations ρ_{in}^i and ρ_{out}^i, then σ is equivariant with respect to ρ_{in} and ρ_{out}. This can be quickly verified, as $\forall h \in H$ and $\forall f(\boldsymbol{x}) \in \mathbb{R}^{c_{\text{in}}}$:

$$\begin{aligned}
\sigma(\rho_{\text{in}}(h) f(\boldsymbol{x})) &= \sigma(\bigoplus_i \rho_{\text{in}}^i(h) f_i(\boldsymbol{x})) \\
&= \bigoplus_i \sigma_i(\rho_{\text{in}}^i(h) f_i(\boldsymbol{x})) \\
&= \bigoplus_i \rho_{\text{out}}^i(h) \sigma_i(f_i(\boldsymbol{x})) \\
&= \rho_{\text{out}}(h) \bigoplus_i \sigma_i(f_i(\boldsymbol{x})) \\
&= \rho_{\text{out}}(h) \sigma(f(\boldsymbol{x}))
\end{aligned}$$

In the rest of this section, we prove the equivariance of different non-linearities σ with respect to different H-representations. For simplicity, we will just consider the non-linearity applied on a generic vector $\boldsymbol{v}$ instead of a feature vector $f(\boldsymbol{x})$ defined by a specific field f evaluated on a point $\boldsymbol{x}$.

A.1 Norm non-linearities for unitary representations

A unitary representation $\rho : H \to \mathbb{R}^{c \times c}$ is a representation which evaluates to unitary matrices, that is they preserve the norm of the vectors:

$$||\rho(h)\boldsymbol{v}||_2 = ||\boldsymbol{v}||_2 \quad \forall \boldsymbol{v} \in \mathbb{R}^c,\ \forall h \in H \ .$$

As introduced in Sec. 3.6, a norm non-linearity σ has form

$$\sigma(\boldsymbol{v}) := \eta\left(||\boldsymbol{v}||_2\right) \frac{\boldsymbol{v}}{||\boldsymbol{v}||_2}$$

where $\eta : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is a non-linear function[1] such that $\eta(0) = 0$ which is applied on the norm of the feature vector. Then, because σ leaves the orientation of its input unchanged, it commutes with a unitary representation ρ, indeed $\forall h \in H$:

$$\begin{aligned}\sigma(\rho(h)\ \boldsymbol{v}) &= \eta\left(||\rho(h)\boldsymbol{v}||_2\right) \frac{\rho(h)\boldsymbol{v}}{||\rho(h)\ \boldsymbol{v}||_2} \\ &= \eta\left(||\boldsymbol{v}||_2\right) \frac{\rho(h)\ \boldsymbol{v}}{||\boldsymbol{v}||_2} \\ &= \rho(h)\ \sigma\left(\boldsymbol{v}\right)\end{aligned}$$

A.2 Point-wise nonlinearities for regular and quotient representations

The action of regular representations and, more generally, quotient representations is defined via the group composition as

$$\rho_{\text{quot}}^{H/K}(\tilde{h})\boldsymbol{e}_{hK} := \boldsymbol{e}_{\tilde{h}hK}$$

where the vectors $\{\boldsymbol{e}_{hK} | hK \in H/K\}$, indexed by the cosets in H/K, form a basis for the representation space of $\rho^{H/K}$. Therefore, given a vector $\boldsymbol{v} = \sum_{hK} v_{hK}\boldsymbol{e}_{hK}$ expressed in this basis, these representations act by *permuting* its coefficients $\{v_{hK}\}_{hK}$.

A point-wise non-linearity σ independently applies the same non-linear function $\nu : \mathbb{R} \to \mathbb{R}$ on each entry of an input vector, i.e.:

$$\sigma : \boldsymbol{v} = \sum_i v_i \boldsymbol{e}_i \quad \mapsto \quad \sigma(\boldsymbol{v}) := \sum_i \nu(v_i)\boldsymbol{e}_i \quad .$$

Then, any regular, quotient or, more generally, *permutation* representation[2] ρ commutes with any point-wise σ. Indeed, by denoting with $\boldsymbol{e}_{\rho(h)[i]} := \rho(h)\boldsymbol{e}_i$:

$$\begin{aligned}\sigma(\rho(h)\ \boldsymbol{v}) = \sigma(\rho(h)\ \sum_i v_i \boldsymbol{e}_i) &= \sigma(\sum_i v_i \rho(h)\ \boldsymbol{e}_i) \\ &= \sigma(\sum_i v_i \boldsymbol{e}_{\rho(h)[i]}) \\ &= \sigma(\sum_i v_{\rho(h^{-1})[i]} \boldsymbol{e}_i) \\ &= \sum_i \nu(v_{\rho(h^{-1})[i]})\boldsymbol{e}_i \\ &= \sum_i \nu(v_i)\rho(h)\ \boldsymbol{e}_i = \rho(h)\ \sum_i \nu(v_i)\boldsymbol{e}_i = \rho(h)\ \sigma(\boldsymbol{v})\end{aligned}$$

[1] For stability around the origin, it is preferable to choose a function η s.t. $\eta(x) \leq x$ for x close to 0.
[2] a representation which evaluates to permutation matrices.

A.3 Vector-field non-linearities for regular and standard representations

Each element $r^p \in \mathrm{C}_N$ can be identified with a rotation by an angle $\theta_p = p\frac{2\pi}{N}$, see Examples 4 and 8. At the same time, one can identify each entry of a vector $\boldsymbol{v}$ in the regular representation $\rho_{\text{reg}}^{\mathrm{C}_N}$ with an element $r^p \in \mathrm{C}_N$ - and, therefore, with an angle θ_p - as $\boldsymbol{v} = \sum_{p=0}^{N-1} v_p \boldsymbol{e}_p$.

Then, a vector-field non-linearity $\sigma : \mathbb{R}^N \to \mathbb{R}^2$ maps an N-dimensional vector $\boldsymbol{v}$ transforming according to the regular representation $\rho_{\text{in}} = \rho_{\text{reg}}^{\mathrm{C}_N}$ of C_N to a 2-dimensional vector transforming according to the irreducible representation $\rho_{\text{out}} = \psi_1^{\mathrm{C}_N}$ (see Sec. 2.7.2) as:

$$\sigma(\boldsymbol{v}) := \max \boldsymbol{v} \begin{pmatrix} \cos\theta_{\operatorname{argmax} \boldsymbol{v}} \\ \sin\theta_{\operatorname{argmax} \boldsymbol{v}} \end{pmatrix}$$

The equivariance of this operation can be proven as follows. $\forall r^q \in \mathrm{C}_N$:

$$\begin{aligned} \sigma(\rho_{\text{reg}}^{\mathrm{C}_N}(r^q)\ \boldsymbol{v}) &= \sigma(\rho_{\text{reg}}^{\mathrm{C}_N}(r^q)\ \sum_p v_p \boldsymbol{e}_p) \\ &= \sigma(\sum_p v_p \boldsymbol{e}_{p+q}) \end{aligned}$$

Define $\hat{p} := \operatorname{argmax}_p v_p$. Then, $\max \rho_{\text{reg}}^{\mathrm{C}_N}(r^q)\ \boldsymbol{v} = v_{\hat{p}}$ and $\operatorname{argmax} \rho_{\text{reg}}^{\mathrm{C}_N}(r^q)\ \boldsymbol{v} = \hat{p}+q$.

$$\begin{aligned} &= v_{\hat{p}} \begin{pmatrix} \cos\theta_{\hat{p}+q} \\ \sin\theta_{\hat{p}+q} \end{pmatrix} \\ &= v_{\hat{p}} \begin{pmatrix} \cos\theta_{\hat{p}} + \theta_q \\ \sin\theta_{\hat{p}} + \theta_q \end{pmatrix} \\ &= v_{\hat{p}} \begin{bmatrix} \cos(\theta_q) & -\sin(\theta_q) \\ \sin(\theta_q) & \cos(\theta_q) \end{bmatrix} \begin{pmatrix} \cos\theta_{\hat{p}} \\ \sin\theta_{\hat{p}} \end{pmatrix} \\ &= \psi_1^{\mathrm{C}_N}(r^q)\ \sigma(\boldsymbol{v}) \end{aligned}$$

A.4 Induced non-linearities

Let K be a subgroup of H and $\psi_{\text{in}} : K \to \mathrm{GL}(\mathbb{R}^{d_{\text{in}}})$ and $\psi_{\text{out}} : K \to \mathrm{GL}(\mathbb{R}^{d_{\text{out}}})$ two K-representations. Let also $\nu : \mathbb{R}^{d_{\text{in}}} \to \mathbb{R}^{d_{\text{out}}}$ be a non-linearity which is equivariant to K with respect to ψ_{in} and ψ_{out}. Assume that $\rho_{\text{in}} = \operatorname{Ind}_K^H \psi_{\text{in}} : H \to \mathrm{GL}(\mathbb{R}^{c_{\text{in}}})$ and $\rho_{\text{out}} = \operatorname{Ind}_K^H \psi_{\text{out}} : H \to \mathrm{GL}(\mathbb{R}^{c_{\text{out}}})$, with $c_{\text{in}} = |H : K| d_{\text{in}}$ and $c_{\text{out}} = |H : K| d_{\text{out}}$.

Recall that a vector $\boldsymbol{v} \in \mathbb{R}^{c_{\text{in}}}$ in the representation space of $\text{Ind}_K^H \psi_{\text{in}}$ can be expressed as

$$\boldsymbol{v} = \sum_{hK \in H/K} \boldsymbol{e}_{hK} \otimes \boldsymbol{v}_{hK}$$

where $\boldsymbol{v}_{hK} \in \mathbb{R}^{d_{\text{in}}}$ is the component of $\boldsymbol{v}$ in the subspace associated with the coset hK. See Sec. 2.4. Then, with an abuse of notation, we define the induced non-linearity $\sigma = \text{Ind}_K^H \nu$ which acts by independently applying ν to the components of each of the $|H : K|$ subspaces:

$$\sigma(\boldsymbol{v}) = \sigma\left(\sum_{hK} \boldsymbol{e}_{hK} \otimes \boldsymbol{v}_{hK}\right) := \sum_{hK} \boldsymbol{e}_{hK} \otimes \nu(\boldsymbol{v}_{hK}) \quad .$$

Note that a permutation of the $|H : K|$ subspaces commutes with the application of ν on each of them individually and ν is assumed to be equivariant to the transformation of a subspace through ψ_{in} and ψ_{out}. Therefore, the induced non-linearity σ is equivariant to H with respect to $\rho_{\text{in}} = \text{Ind}_K^H \psi_{\text{in}}$ and $\rho_{\text{out}} = \text{Ind}_K^H \psi_{\text{out}}$. This can be formally proven as follows. First of all, recall that, given a *section* $\mathcal{R} : H/K \to H$, any element $h \in H$ can be uniquely written as $h = \mathcal{R}(hK)\text{k}(h)$ [3]. Then, $\forall \tilde{h} \in H$:

$$\begin{aligned}
\sigma\left(\text{Ind}\, \psi_{\text{in}}(h)\, \boldsymbol{v}\right) &= \sigma\left(\sum_{hK} \boldsymbol{e}_{\tilde{h}hK} \otimes \psi_{\text{in}}\left(\text{k}\left(\tilde{h}\mathcal{R}(hK)\right)\right) \boldsymbol{v}_{hK}\right) \\
&= \sum_{hK} \boldsymbol{e}_{\tilde{h}hK} \otimes \nu\left(\psi_{\text{in}}\left(\text{k}\left(\tilde{h}\mathcal{R}(hK)\right)\right) \boldsymbol{v}_{hK}\right) \\
&= \sum_{hK} \boldsymbol{e}_{\tilde{h}hK} \otimes \psi_{\text{out}}\left(\text{k}\left(\tilde{h}\mathcal{R}(hK)\right)\right) \nu\left(\boldsymbol{v}_{hK}\right) \\
&= \text{Ind}\, \psi_{\text{out}}(h)\, \sigma\left(\boldsymbol{v}\right)
\end{aligned}$$

[3] The function k here is the equivalent of the function h in Sec. 2.4. Note that here we induce from K to H, while there we considered induction from H to G. All the notation used here is equivalent to the one used in Sec. 2.4, except for the fact that all k should be replaced with h while all h should be replaced with g.

B Solutions of the kernel constraints for irreducible representations

In this section we are deriving analytical solutions of the kernel constraints

$$\kappa^{ij}(h\boldsymbol{x}) \;=\; \psi_i(h)\,\kappa^{ij}(\boldsymbol{x})\,\psi_j^{-1}(h) \qquad \forall h \in H,\; \boldsymbol{x} \in \mathbb{R}^2 \tag{B.1}$$

for irreducible representations ψ_i of $\mathrm{O}(2)$ and its subgroups. The linearity of the constraint implies that the solution space of G-steerable kernels forms a linear subspace of the unrestricted kernel space $k \in L^2\left(\mathbb{R}^2\right)^{c_{\text{out}} \times c_{\text{in}}}$ of square integrable functions $k : \mathbb{R}^2 \to \mathbb{R}^{c_{\text{out}} \times c_{\text{in}}}$.

Since our numerical implementation is on the real field we are considering real-valued irreps. It is in general possible to derive all solutions considering complex valued irreps of $H \leq \mathrm{O}(2)$. While this approach would simplify some steps it comes with an overhead of relating the final results back to the real field which leads to further complications, see Appendix B.3. An overview over the real-valued irreps of $H \leq \mathrm{O}(2)$ and their properties was given in Sec. 2.7.2.

We present the analytical solutions of the irrep kernel constraints for all possible pairs of irreps in Sec. B.1. Specifically, the solutions for $\mathrm{SO}(2)$ are given in Tab. B.1 while the solutions for $\mathrm{O}(2)$, $(\{\pm 1\}, *)$, C_N and D_N are given in Tab. B.2, Tab. B.3, Tab. B.4 and Tab. B.5, respectively.

Our derivation of the irrep kernel bases is motivated by the observation that the irreps of $\mathrm{O}(2)$ and subgroups are harmonics, that is, they are associated to one particular angular frequency. This suggests that the kernel constraint in Eq. (B.1) decouples into simpler constraints on individual Fourier modes. In the derivations, presented in Sec. B.2, we are therefore defining the kernels in polar coordinates $x = x(r, \phi)$ and expand them in terms of an orthogonal, angular, Fourier-like basis. A projection on this orthogonal basis then yields constraints on the expansion coefficients. Only specific coefficients are allowed to be non-zero; these coefficients parameterize the complete space of G-steerable kernels satisfying the irrep constraint (B.1). The completeness of the solution follows from the completeness of the orthogonal basis.

We start with deriving the bases for the simplest cases $\mathrm{SO}(2)$ and $(\{\pm 1\}, *)$ in sections B.2.2 and B.2.3. The H-steerable kernel basis for $\mathrm{O}(2)$ forms a subspace of the kernel basis for $\mathrm{SO}(2)$ such that it can be easily derived from this solution by adding the additional constraint coming from the reflectional symmetries in $(\{\pm 1\}, *) \cong \mathrm{O}(2)/\mathrm{SO}(2)$. This additional constraint is imposed in Sec. B.2.4. Since C_N is a subgroup of discrete rotations in $\mathrm{SO}(2)$ their derivation is mostly similar. However, the discreteness of rotation angles leads to N systems of linear congruences modulo N in the final step. This system of equations is solved in Sec. B.2.5. Similar to how we derived the kernel basis for $\mathrm{O}(2)$ from $\mathrm{SO}(2)$, we derive the basis for D_N from C_N by adding reflectional constraints from $(\{\pm 1\}, *) \cong \mathrm{D}_N \,/\, \mathrm{C}_N$ in Sec. B.2.6.

B.1 Analytical solutions of the irrep kernel constraints

Special Orthogonal Group $\mathrm{SO}(2)$

ψ_m \ ψ_n	ψ_0	$\psi_n,\ n \in \mathbb{N}^+$
ψ_0	$\begin{pmatrix} 1 \end{pmatrix}$	$\begin{pmatrix} \cos(n\phi) & \sin(n\phi) \end{pmatrix}, \begin{pmatrix} -\sin(n\phi) & \cos(n\phi) \end{pmatrix}$
$\psi_m,$ $m \in \mathbb{N}^+$	$\begin{pmatrix} \cos(m\phi) \\ \sin(m\phi) \end{pmatrix}, \begin{pmatrix} -\sin(m\phi) \\ \cos(m\phi) \end{pmatrix}$	$\begin{pmatrix} \cos((m-n)\phi) & -\sin((m-n)\phi) \\ \sin((m-n)\phi) & \cos((m-n)\phi) \end{pmatrix}, \begin{pmatrix} -\sin((m-n)\phi) & -\cos((m-n)\phi) \\ \cos((m-n)\phi) & -\sin((m-n)\phi) \end{pmatrix}$ $\begin{pmatrix} \cos((m+n)\phi) & \sin((m+n)\phi) \\ \sin((m+n)\phi) & -\cos((m+n)\phi) \end{pmatrix}, \begin{pmatrix} -\sin((m+n)\phi) & \cos((m+n)\phi) \\ \cos((m+n)\phi) & \sin((m+n)\phi) \end{pmatrix}$

Tab. B.1.: Bases for the angular parts of $\mathrm{SO}(2)$-steerable kernels satisfying the irrep constraint (3.12) for different pairs of input and output field irreps ψ_n and ψ_m. The different types of irreps are explained in Sec. 2.7.2.

Orthogonal Group $\mathrm{O}(2)$

$\psi_{i,m}$ \ $\psi_{j,n}$	$\psi_{0,0}$	$\psi_{1,0}$	$\psi_{1,n},\ n \in \mathbb{N}^+$
$\psi_{0,0}$	$\begin{pmatrix} 1 \end{pmatrix}$	$\varnothing$	$\begin{pmatrix} -\sin(n\phi) & \cos(n\phi) \end{pmatrix}$
$\psi_{1,0}$	$\varnothing$	$\begin{pmatrix} 1 \end{pmatrix}$	$\begin{pmatrix} \cos(n\phi) & \sin(n\phi) \end{pmatrix}$
$\psi_{1,m},$ $m \in \mathbb{N}^+$	$\begin{pmatrix} -\sin(m\phi) \\ \cos(m\phi) \end{pmatrix}$	$\begin{pmatrix} \cos(m\phi) \\ \sin(m\phi) \end{pmatrix}$	$\begin{pmatrix} \cos((m-n)\phi) & -\sin((m-n)\phi) \\ \sin((m-n)\phi) & \cos((m-n)\phi) \end{pmatrix}, \begin{pmatrix} \cos((m+n)\phi) & \sin((m+n)\phi) \\ \sin((m+n)\phi) & -\cos((m+n)\phi) \end{pmatrix}$

Tab. B.2.: Bases for the angular parts of $\mathrm{O}(2)$-steerable kernels satisfying the irrep constraint (3.12) for different pairs of input and output field irreps $\psi_{j,n}$ and $\psi_{i,m}$. The different types of irreps are explained in Sec. 2.7.2.

Reflection group $(\{\pm 1\}, *)$

ψ_i \ ψ_j	ψ_0	ψ_1
ψ_0	$\left[\cos\left(\mu(\phi-\beta)\right)\right]$	$\left[\sin\left(\mu(\phi-\beta)\right)\right]$
ψ_1	$\left[\sin\left(\mu(\phi-\beta)\right)\right]$	$\left[\cos\left(\mu(\phi-\beta)\right)\right]$

Tab. B.3.: Bases for the angular parts of $(\{\pm 1\}, *)$-steerable kernels satisfying the irrep constraint (3.12) for different pairs of input and output field irreps ψ_j and ψ_i for $i, j \in \{0, 1\}$. The different types of irreps are explained in Sec. 2.7.2. The reflection f is defined along the axis identified by the angle β wrt the x-axis. Note that these bases are a special case of those in Tab. B.5 since $(\{\pm 1\}, *) \cong \mathrm{D}_1$.

Cyclic groups C_N

ψ_m \ ψ_n	ψ_0	$\psi_{N/2}$ (if N even)	ψ_n with $n \in \mathbb{N}^+$ and $1 \leq n < N/2$
ψ_0	$\begin{bmatrix}\cos(\hat{t}N\phi)\end{bmatrix}$, $\begin{bmatrix}\sin(\hat{t}N\phi)\end{bmatrix}$	$\begin{bmatrix}\cos\left(\left(\hat{t}+\frac{1}{2}\right)N\phi\right)\end{bmatrix}$, $\begin{bmatrix}\sin\left(\left(\hat{t}+\frac{1}{2}\right)N\phi\right)\end{bmatrix}$	$\begin{bmatrix}-\sin((n+tN)\phi) & \cos((n+tN)\phi)\end{bmatrix}$, $\begin{bmatrix}\cos((n+tN)\phi) & \sin((n+tN)\phi)\end{bmatrix}$
$\psi_{N/2}$ (N even)	$\begin{bmatrix}\cos\left(\left(\hat{t}+\frac{1}{2}\right)N\phi\right)\end{bmatrix}$, $\begin{bmatrix}\sin\left(\left(\hat{t}+\frac{1}{2}\right)N\phi\right)\end{bmatrix}$	$\begin{bmatrix}\cos(\hat{t}N\phi)\end{bmatrix}$, $\begin{bmatrix}\sin(\hat{t}N\phi)\end{bmatrix}$	$\begin{bmatrix}-\sin\left(\left(n+\left(t+\frac{1}{2}\right)N\right)\phi\right) & \cos\left(\left(n+\left(t+\frac{1}{2}\right)N\right)\phi\right)\end{bmatrix}$, $\begin{bmatrix}\cos\left(\left(n+\left(t+\frac{1}{2}\right)N\right)\phi\right) & \sin\left(\left(n+\left(t+\frac{1}{2}\right)N\right)\phi\right)\end{bmatrix}$
ψ_m, $m \in \mathbb{N}^+$ $1 \leq m < N/2$	$\begin{bmatrix}-\sin((m+tN)\phi)\\ \cos((m+tN)\phi)\end{bmatrix}$, $\begin{bmatrix}\cos((m+tN)\phi)\\ \sin((m+tN)\phi)\end{bmatrix}$	$\begin{bmatrix}-\sin\left(\left(m+\left(t+\frac{1}{2}\right)N\right)\phi\right)\\ \cos\left(\left(m+\left(t+\frac{1}{2}\right)N\right)\phi\right)\end{bmatrix}$, $\begin{bmatrix}\cos\left(\left(m+\left(t+\frac{1}{2}\right)N\right)\phi\right)\\ \sin\left(\left(m+\left(t+\frac{1}{2}\right)N\right)\phi\right)\end{bmatrix}$	$\begin{bmatrix}\cos((m-n+tN)\phi) & -\sin((m-n+tN)\phi)\\ \sin((m-n+tN)\phi) & \cos((m-n+tN)\phi)\end{bmatrix}$, $\begin{bmatrix}-\sin((m-n+tN)\phi) & -\cos((m-n+tN)\phi)\\ \cos((m-n+tN)\phi) & -\sin((m-n+tN)\phi)\end{bmatrix}$, $\begin{bmatrix}\cos((m+n+tN)\phi) & \sin((m+n+tN)\phi)\\ \sin((m+n+tN)\phi) & -\cos((m+n+tN)\phi)\end{bmatrix}$, $\begin{bmatrix}-\sin((m+n+tN)\phi) & \cos((m+n+tN)\phi)\\ \cos((m+n+tN)\phi) & \sin((m+n+tN)\phi)\end{bmatrix}$

Tab. B.4.: Bases for the angular parts of C_N-steerable kernels for different pairs of input and output fields irreps ψ_n and ψ_m. The full basis is found by instantiating these solutions for each $t \in \mathbb{Z}$ or $\hat{t} \in \mathbb{N}$. The different types of irreps are explained in Sec. 2.7.2.

Dihedral groups D_N

$\psi_{i,m}$ \ $\psi_{j,n}$	$\psi_{0,0}$	$\psi_{1,0}$	$\psi_{0,N/2}$ (if N even)	$\psi_{1,N/2}$ (if N even)	$\psi_{1,n}$ with $n \in \mathbb{N}^+$ and $1 \leq n < N/2$
$\psi_{0,0}$	$\left[\cos(\hat{t}N\phi)\right]$	$\left[\sin(\hat{t}N\phi)\right]$	$\left[\cos\left(\left(\hat{t}+\frac{1}{2}\right)N\phi\right)\right]$	$\left[\sin\left(\left(\hat{t}+\frac{1}{2}\right)N\phi\right)\right]$	$\left[-\sin((n+tN)\phi) \quad \cos((n+tN)\phi)\right]$
$\psi_{1,0}$	$\left[\sin(\hat{t}N\phi)\right]$	$\left[\cos(\hat{t}N\phi)\right]$	$\left[\sin\left(\left(\hat{t}+\frac{1}{2}\right)N\phi\right)\right]$	$\left[\cos\left(\left(\hat{t}+\frac{1}{2}\right)N\phi\right)\right]$	$\left[\cos((n+tN)\phi) \quad \sin((n+tN)\phi)\right]$
$\psi_{0,N/2}$ (N even)	$\left[\cos\left(\left(\hat{t}+\frac{1}{2}\right)N\phi\right)\right]$	$\left[\sin\left(\left(\hat{t}+\frac{1}{2}\right)N\phi\right)\right]$	$\left[\cos(\hat{t}N\phi)\right]$	$\left[\sin(\hat{t}N\phi)\right]$	$\left[-\sin\left(\left(n+\left(t+\frac{1}{2}\right)N\right)\phi\right) \quad \cos\left(\left(n+\left(t+\frac{1}{2}\right)N\right)\phi\right)\right]$
$\psi_{1,N/2}$ (N even)	$\left[\sin\left(\left(\hat{t}+\frac{1}{2}\right)N\phi\right)\right]$	$\left[\cos\left(\left(\hat{t}+\frac{1}{2}\right)N\phi\right)\right]$	$\left[\sin(\hat{t}N\phi)\right]$	$\left[\cos(\hat{t}N\phi)\right]$	$\left[\cos\left(\left(n+\left(t+\frac{1}{2}\right)N\right)\phi\right) \quad \sin\left(\left(n+\left(t+\frac{1}{2}\right)N\right)\phi\right)\right]$
$\psi_{1,m}$, $m \in \mathbb{N}^+$ $1 \leq m < N/2$	$\begin{bmatrix}-\sin((m+tN)\phi)\\ \cos((m+tN)\phi)\end{bmatrix}$	$\begin{bmatrix}\cos((m+tN)\phi)\\ \sin((m+tN)\phi)\end{bmatrix}$	$\begin{bmatrix}-\sin\left(\left(m+\left(t+\frac{1}{2}\right)N\right)\phi\right)\\ \cos\left(\left(m+\left(t+\frac{1}{2}\right)N\right)\phi\right)\end{bmatrix}$	$\begin{bmatrix}\cos\left(\left(m+\left(t+\frac{1}{2}\right)N\right)\phi\right)\\ \sin\left(\left(m+\left(t+\frac{1}{2}\right)N\right)\phi\right)\end{bmatrix}$	$\begin{bmatrix}\cos((m-n+tN)\phi) & -\sin((m-n+tN)\phi)\\ \sin((m-n+tN)\phi) & \cos((m-n+tN)\phi)\end{bmatrix}$, $\begin{bmatrix}\cos((m+n+tN)\phi) & \sin((m+n+tN)\phi)\\ \sin((m+n+tN)\phi) & -\cos((m+n+tN)\phi)\end{bmatrix}$

Tab. B.5.: Bases for the angular parts of D_N-steerable kernels for different pairs of input and output fields irreps $\psi_{j,n}$ and $\psi_{i,m}$. The full basis is found by instantiating these solutions for each $t \in \mathbb{Z}$ or $\hat{t} \in \mathbb{N}$. The different types of irreps are explained in Sec. 2.7.2. Here, we assumed the reflection f to be defined along the horizontal x-axis ($\beta = 0$). For reflections along different axes (identified by an angle $\beta \neq 0$), one can substitute ϕ with $\phi - \beta$.

B.2 Derivations of the kernel constraints

Here we solve the kernel constraints for the irreducible representations of $H \leq \mathrm{O}(2)$. Since the irreps of H are either 1- or 2-dimensional, we distinguish between mappings between 2-dimensional irreps, mappings from 2- to 1-dimensional and 1- to 2-dimensional irreps and mappings between 1-dimensional irreps. We are first exclusively considering positive radial parts $r > 0$ in the following sections. The constraint at the origin $r = 0$ requires some additional considerations which we postpone to Appendix B.2.6.1.

B.2.1 Conventions and Basic properties

In our derivations, we adopt the same notation introduced in Sec. 2.7.1. Moreover, these properties will be useful later:

$$\psi(\theta)\xi(s) = \xi(s)\psi(s\theta) \tag{B.2}$$

$$\xi(s)^{-1} = \xi(s)^T = \xi(s) \tag{B.3}$$

$$\psi(\theta)^{-1} = \psi(\theta)^T = \psi(-\theta) \tag{B.4}$$

$$\psi(\theta_1)\psi(\theta_2) = \psi(\theta_1 + \theta_2) = \psi(\theta_2)\psi(\theta_1) \tag{B.5}$$

$$\mathrm{Tr}(\psi(\theta)\xi(-1)) = \mathrm{Tr}\begin{bmatrix} \cos(\theta) & \sin(\theta) \\ \sin(\theta) & -\cos(\theta) \end{bmatrix} = 0 \tag{B.6}$$

$$\mathrm{Tr}(\psi(\theta)) = \mathrm{Tr}\begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} = 2\cos(\theta) \tag{B.7}$$

$$\begin{aligned} w_1\cos(\alpha) + w_2\sin(\alpha) &= w_1\cos(\beta) + w_2\sin(\beta) \quad \forall w_1, w_2 \in \mathbb{R} \\ \Leftrightarrow \quad \exists t \in \mathbb{Z} \text{ s.t. } \alpha &= \beta + 2t\pi \end{aligned} \tag{B.8}$$

B.2.2 Special Orthogonal Group $\mathrm{SO}(2)$

2-dimensional irreps:

We first consider the case of 2-dimensional irreps both in the input and in output, that is, $\rho_{\text{out}} = \psi_m^{\mathrm{SO}(2)}$ and $\rho_{\text{in}} = \psi_n^{\mathrm{SO}(2)}$, where $\psi_k^{\mathrm{SO}(2)}(\theta) = \begin{bmatrix} \cos(k\theta) & -\sin(k\theta) \\ \sin(k\theta) & \cos(k\theta) \end{bmatrix}$. This means that the kernel has the form $\kappa^{ij} : \mathbb{R}^2 \to \mathbb{R}^{2\times 2}$. To reduce clutter we will from now on suppress the indices ij corresponding to the input and output irreps in the input and output fields.

We expand each entry of the kernel κ in terms of an (angular) Fourier series[1]

$$\begin{aligned}\kappa(r,\phi) \;=\; \sum_{\mu=0}^{\infty} & A_{00,\mu}(r)\begin{bmatrix}\cos(\mu\phi) & 0\\ 0 & 0\end{bmatrix} + B_{00,\mu}(r)\begin{bmatrix}\sin(\mu\phi) & 0\\ 0 & 0\end{bmatrix}\\ & + A_{01,\mu}(r)\begin{bmatrix}0 & \cos(\mu\phi)\\ 0 & 0\end{bmatrix} + B_{01,\mu}(r)\begin{bmatrix}0 & \sin(\mu\phi)\\ 0 & 0\end{bmatrix}\\ & + A_{10,\mu}(r)\begin{bmatrix}0 & 0\\ \cos(\mu\phi) & 0\end{bmatrix} + B_{10,\mu}(r)\begin{bmatrix}0 & 0\\ \sin(\mu\phi) & 0\end{bmatrix}\\ & + A_{11,\mu}(r)\begin{bmatrix}0 & 0\\ 0 & \cos(\mu\phi)\end{bmatrix} + B_{11,\mu}(r)\begin{bmatrix}0 & 0\\ 0 & \sin(\mu\phi)\end{bmatrix}\end{aligned}$$

and, for convenience, perform a change of basis to a different, non-sparse, orthogonal basis

$$\begin{aligned}\kappa(r,\phi) = \sum_{\mu=0}^{\infty} & w_{0,0,\mu}(r)\begin{bmatrix}\cos(\mu\phi) & -\sin(\mu\phi)\\ \sin(\mu\phi) & \cos(\mu\phi)\end{bmatrix} + w_{0,1,\mu}(r)\begin{bmatrix}\cos(\mu\phi+\frac{\pi}{2}) & -\sin(\mu\phi+\frac{\pi}{2})\\ \sin(\mu\phi+\frac{\pi}{2}) & \cos(\mu\phi+\frac{\pi}{2})\end{bmatrix}\\ & + w_{1,0,\mu}(r)\begin{bmatrix}\cos(\mu\phi) & \sin(\mu\phi)\\ \sin(\mu\phi) & -\cos(\mu\phi)\end{bmatrix} + w_{1,1,\mu}(r)\begin{bmatrix}\cos(\mu\phi+\frac{\pi}{2}) & \sin(\mu\phi+\frac{\pi}{2})\\ \sin(\mu\phi+\frac{\pi}{2}) & -\cos(\mu\phi+\frac{\pi}{2})\end{bmatrix}\\ & + w_{2,0,\mu}(r)\begin{bmatrix}\cos(-\mu\phi) & -\sin(-\mu\phi)\\ \sin(-\mu\phi) & \cos(-\mu\phi)\end{bmatrix} + w_{2,1,\mu}(r)\begin{bmatrix}\cos(-\mu\phi+\frac{\pi}{2}) & -\sin(-\mu\phi+\frac{\pi}{2})\\ \sin(-\mu\phi+\frac{\pi}{2}) & \cos(-\mu\phi+\frac{\pi}{2})\end{bmatrix}\\ & + w_{3,0,\mu}(r)\begin{bmatrix}\cos(-\mu\phi) & \sin(-\mu\phi)\\ \sin(-\mu\phi) & -\cos(-\mu\phi)\end{bmatrix} + w_{3,1,\mu}(r)\begin{bmatrix}\cos(-\mu\phi+\frac{\pi}{2}) & \sin(-\mu\phi+\frac{\pi}{2})\\ \sin(-\mu\phi+\frac{\pi}{2}) & -\cos(-\mu\phi+\frac{\pi}{2})\end{bmatrix}.\end{aligned}$$

The last four matrices are equal to the first four, except for their opposite frequency. Moreover, the second matrices of each row are equal to the first matrices, with a phase shift of $\frac{\pi}{2}$ added. Therefore, we can as well write:

$$\kappa(r,\phi) = \sum_{\mu=-\infty}^{\infty}\sum_{\gamma\in\{0,\frac{\pi}{2}\}} w_{0,\gamma,\mu}(r)\begin{bmatrix}\cos(\mu\phi+\gamma) & -\sin(\mu\phi+\gamma)\\ \sin(\mu\phi+\gamma) & \cos(\mu\phi+\gamma)\end{bmatrix} + w_{1,\gamma,\mu}(r)\begin{bmatrix}\cos(\mu\phi+\gamma) & \sin(\mu\phi+\gamma)\\ \sin(\mu\phi+\gamma) & -\cos(\mu\phi+\gamma)\end{bmatrix}$$

Notice that the first matrix evaluates to $\psi(\mu\phi+\gamma)\xi(1) = \psi(\mu\phi+\gamma)$ while the second evaluates to $\psi(\mu\phi+\gamma)\xi(-1)$. Hence, for $s \in \{\pm 1\}$ we can compactly write:

$$\kappa(r,\phi) = \sum_{\mu=-\infty}^{\infty}\sum_{\gamma\in\{0,\frac{\pi}{2}\}}\sum_{s\in\{\pm 1\}} w_{s,\gamma,\mu}(r)\psi(\mu\phi+\gamma)\xi(s)$$

[1] For brevity, we suppress that frequency 0 is associated to only half the number of basis elements which does not affect the validity of the derivation.

As already shown in Sec. 3.5, we can w.l.o.g. consider the kernels as being defined only on the angular component $\phi \in [0, 2\pi) = S^1$ by solving only for a specific radial component r. As a result, we consider the basis

$$\left\{ b_{\mu,\gamma,s}(\phi) = \psi(\mu\phi + \gamma)\xi(s) \;\middle|\; \mu \in \mathbb{Z},\ \gamma \in \left\{0, \frac{\pi}{2}\right\},\ s \in (\{\pm 1\}, *) \right\} \tag{B.9}$$

of the unrestricted kernel space which we will constrain in the following by demanding

$$\kappa(\phi + \theta) = \psi_m^{\mathrm{SO}(2)}(r_\theta)\kappa(\phi)\psi_n^{\mathrm{SO}(2)}(r_\theta)^{-1} \quad \forall \phi, \theta \in [0, 2\pi), \tag{B.10}$$

where we dropped the unrestricted radial part.

We solve for a basis of the subspace satisfying this constraint by projecting both sides on the basis elements defined above. The inner product on $L^2(S^1)^{2\times 2}$ is hereby defined as

$$\langle k_1, k_2 \rangle = \frac{1}{4\pi} \int d\phi \langle k_1(\phi), k_2(\phi) \rangle_F = \frac{1}{4\pi} \int d\phi \ \mathrm{Tr}\left(k_1(\phi)^T k_2(\phi) \right),$$

where $\langle \cdot, \cdot \rangle_F$ denotes the *Frobenius* inner product between 2 matrices.

First consider the projection of the *lhs* of the kernel constraint (B.10) on a generic basis element $b_{\mu',\gamma',s'}(\phi) = \psi(\mu'\phi + \gamma')\xi(s')$. Defining the operator R_θ by $(R_\theta \kappa)(\phi) := \kappa(\phi + \theta)$, the projection gives:

$$\begin{aligned} \langle b_{\mu',\gamma',s'}, R_\theta \kappa \rangle &= \frac{1}{4\pi} \int d\phi \ \mathrm{Tr}\left(b_{\mu',\gamma',s'}(\phi)^T \left(R_\theta \kappa\right)(\phi) \right) \\ &= \frac{1}{4\pi} \int d\phi \ \mathrm{Tr}\left(b_{\mu',\gamma',s'}(\phi)^T \kappa(\phi + \theta) \right). \end{aligned}$$

By expanding the kernel in the linear combination of the basis we further obtain

$$= \frac{1}{4\pi} \int d\phi \ \mathrm{Tr}\left(b_{\mu',\gamma',s'}(\phi)^T \left(\sum_\mu \sum_\gamma \sum_s w_{s,\gamma,\mu} \psi\left(\mu(\phi + \theta) + \gamma\right) \xi(s) \right) \right),$$

which, observing that the trace, sums and integral commute, results in:

$$\begin{aligned} &= \frac{1}{4\pi} \sum_\mu \sum_\gamma \sum_s w_{s,\gamma,\mu} \ \mathrm{Tr}\left(\int d\phi \ \ b_{\mu',\gamma',s'}(\phi)^T \psi\left(\mu(\phi + \theta) + \gamma\right) \xi(s) \right) \\ &= \frac{1}{4\pi} \sum_\mu \sum_\gamma \sum_s w_{s,\gamma,\mu} \ \mathrm{Tr}\left(\int d\phi \ \ \left(\psi(\mu'\phi + \gamma')\xi(s')\right)^T \psi\left(\mu(\phi + \theta) + \gamma\right) \xi(s) \right) \\ &= \frac{1}{4\pi} \sum_\mu \sum_\gamma \sum_s w_{s,\gamma,\mu} \ \mathrm{Tr}\left(\int d\phi \ \xi(s')^T \psi(\mu'\phi + \gamma')^T \psi\left(\mu(\phi + \theta) + \gamma\right) \xi(s) \right) \end{aligned}$$

Using the properties in Eq. (B.3) and (B.4) then yields:

$$= \frac{1}{4\pi} \sum_{\mu} \sum_{\gamma} \sum_{s} w_{s,\gamma,\mu} \operatorname{Tr} \left(\int d\phi \, \xi(s') \psi(-\mu' \phi - \gamma') \psi \left(\mu(\phi + \theta) + \gamma \right) \xi(s) \right)$$
$$= \frac{1}{2} \sum_{\mu} \sum_{\gamma} \sum_{s} w_{s,\gamma,\mu} \operatorname{Tr} \left(\xi(s') \psi(\gamma - \gamma') \left(\frac{1}{2\pi} \int d\phi \, \psi((\mu - \mu')\phi) \right) \psi(\mu\theta) \xi(s) \right)$$

In the integral, each cell of the matrix $\psi((\mu - \mu')\phi)$ contains either a sine or cosine. As a result, if $\mu - \mu' \neq 0$, all these integrals evaluate to 0. Otherwise, the cosines on the diagonal evaluate to 1, while the sines integrate to 0. The whole integral evaluates to $\delta_{\mu,\mu'} \, \mathrm{id}_{2\times 2}$, such that

$$= \frac{1}{2} \sum_{\gamma} \sum_{s} w_{s,\gamma,\mu'} \operatorname{Tr} \left(\xi(s') \psi(\gamma - \gamma') \psi(\mu'\theta) \xi(s) \right),$$

which, using the property in Eq. (B.2) leads to

$$= \frac{1}{2} \sum_{\gamma} \sum_{s} w_{s,\gamma,\mu'} \operatorname{Tr} \left(\psi(s'(\gamma - \gamma' + \mu'\theta)) \xi(s' * s) \right).$$

Recall the propetries of the trace in Eq. (B.6), (B.7). If $s' * s = -1$, i.e. $s' \neq s$, the matrix has a trace of 0:

$$= \frac{1}{2} \sum_{\gamma} \sum_{s} w_{s,\gamma,\mu'} \delta_{s',s} 2 \cos(s'(\gamma - \gamma' + \mu'\theta))$$

Since $\cos(-\alpha) = \cos(\alpha)$ and $s' \in \{\pm 1\}$:

$$= \frac{1}{2} \sum_{\gamma} \sum_{s} w_{s,\gamma,\mu'} \delta_{s',s} 2 \cos(\gamma - \gamma' + \mu'\theta)$$
$$= \sum_{\gamma} w_{s',\gamma,\mu'} \cos((\gamma - \gamma') + \mu'\theta)$$

Next consider the projection of the *rhs* of Eq. (B.10):

$$\langle b_{\mu',\gamma',s'}, \; \psi_m^{\mathrm{SO}(2)}(r_\theta) \kappa(\cdot) \psi_n^{\mathrm{SO}(2)}(r_\theta)^{-1} \rangle$$
$$= \frac{1}{4\pi} \int d\phi \;\; \operatorname{Tr} \left(b_{\mu',\gamma',s'}(\phi)^T \psi_m^{\mathrm{SO}(2)}(r_\theta) \kappa(\phi) \psi_n^{\mathrm{SO}(2)}(r_\theta)^{-1} \right)$$
$$= \frac{1}{4\pi} \int d\phi \;\; \operatorname{Tr} \left(b_{\mu',\gamma',s'}(\phi)^T \psi(m\theta) \kappa(\phi) \psi(-n\theta) \right)$$

An expansion of the kernel in the linear combination of the basis yields:

$$
\begin{aligned}
&= \frac{1}{4\pi} \int d\phi \ \mathrm{Tr} \left(b_{\mu',\gamma',s'}(\phi)^T \psi(m\theta) \left(\sum_\mu \sum_\gamma \sum_s w_{s,\gamma,\mu} \psi\left(\mu\phi + \gamma\right) \xi(s) \right) \psi(-n\theta) \right) \\
&= \frac{1}{4\pi} \sum_\mu \sum_\gamma \sum_s w_{s,\gamma,\mu} \, \mathrm{Tr} \left(\int d\phi \ b_{\mu',\gamma',s'}(\phi)^T \psi(m\theta) \psi\left(\mu\phi + \gamma\right) \xi(s) \psi(-n\theta) \right) \\
&= \frac{1}{4\pi} \sum_\mu \sum_\gamma \sum_s w_{s,\gamma,\mu} \, \mathrm{Tr} \left(\int d\phi \ \xi(s') \psi(-\mu'\phi - \gamma') \psi(m\theta) \psi\left(\mu\phi + \gamma\right) \xi(s) \psi(-n\theta) \right) \\
&= \frac{1}{2} \sum_\mu \sum_\gamma \sum_s w_{s,\gamma,\mu} \, \mathrm{Tr} \left(\xi(s') \psi(\gamma - \gamma') \psi(m\theta) \left(\frac{1}{2\pi} \int d\phi \ \psi((\mu - \mu')\phi) \right) \xi(s) \psi(-n\theta) \right)
\end{aligned}
$$

Again, the integral evaluates to $\delta_{\mu,\mu'} \, \mathrm{id}_{2\times 2}$:

$$
\begin{aligned}
&= \frac{1}{2} \sum_\mu \sum_\gamma \sum_s w_{s,\gamma,\mu} \delta_{\mu,\mu'} \, \mathrm{Tr} \left(\xi(s') \psi(\gamma - \gamma') \psi(m\theta) \xi(s) \psi(-n\theta) \right) \\
&= \frac{1}{2} \sum_\gamma \sum_s w_{s,\gamma,\mu'} \, \mathrm{Tr} \left(\xi(s') \psi(\gamma - \gamma') \psi(m\theta) \xi(s) \psi(-n\theta) \right) \\
&= \frac{1}{2} \sum_\gamma \sum_s w_{s,\gamma,\mu'} \, \mathrm{Tr} \left(\psi(s'(\gamma - \gamma' + m\theta - ns\theta)) \xi(s' * s) \right)
\end{aligned}
$$

For the same reason as before, the trace is not zero if and only if $s' = s$:

$$
= \frac{1}{2} \sum_\gamma \sum_s w_{s,\gamma,\mu'} \delta_{s',s} 2 \cos(s'(\gamma - \gamma' + m\theta - ns\theta))
$$

Since $\cos(-\alpha) = \cos(\alpha)$ and $s' \in \{\pm 1\}$:

$$
\begin{aligned}
&= \sum_\gamma w_{s',\gamma,\mu'} \cos(\gamma - \gamma' + m\theta - ns'\theta) \\
&= \sum_\gamma w_{s',\gamma,\mu'} \cos((\gamma - \gamma') + (m - ns')\theta)
\end{aligned}
$$

Finally, we require the two projections to be equal for all rotations in $\mathrm{SO}(2)$, that is,

$$
\sum_\gamma w_{s',\gamma,\mu'} \cos((\gamma - \gamma') + \mu'\theta) = \sum_\gamma w_{s',\gamma,\mu'} \cos((\gamma - \gamma') + (m - ns')\theta) \qquad \forall \theta \in [0, 2\pi),
$$

or, explicitly, with $\gamma \in \{0, \frac{\pi}{2}\}$ and $\cos(\alpha + \frac{\pi}{2}) = -\sin(\alpha)$:

$$
\begin{aligned}
& w_{s',0,\mu'} \cos(\mu'\theta - \gamma') && - \ w_{s',\frac{\pi}{2},\mu'} \sin(\mu'\theta - \gamma') \\
= \ & w_{s',0,\mu'} \cos((m - ns')\theta - \gamma') && - \ w_{s',\frac{\pi}{2},\mu'} \sin((m - ns')\theta - \gamma') \qquad \forall \theta \in [0, 2\pi)
\end{aligned}
$$

Using the property in Eq. (B.8) then implies that for each θ in $[0, 2\pi)$ there exists a $t \in \mathbb{Z}$ such that:

$$\begin{aligned} & & \mu'\theta - \gamma' &= (m - ns')\theta - \gamma' + 2t\pi \\ \Leftrightarrow & & (\mu' - (m - ns'))\theta &= 2t\pi \end{aligned} \tag{B.11}$$

Since the constraint needs to hold for any $\theta \in [0, 2\pi)$ this results in the condition $\mu' = m - sn'$ on the frequencies occurring in the $\mathrm{SO}(2)$-steerable kernel basis. Both γ and s are left unrestricted such that we end up with the four-dimensional basis

$$\boxed{\mathcal{K}^{\mathrm{SO}(2)}_{\psi_m \leftarrow \psi_n} = \left\{ b_{\mu,\gamma,s}(\phi) = \psi(\mu\phi + \gamma)\xi(s) \;\middle|\; \mu = (m \text{ - } sn),\ \gamma \in \left\{0, \frac{\pi}{2}\right\},\ s \in \{\pm 1\} \right\}} \tag{B.12}$$

for the angular parts of equivariant kernels for $m, n > 0$. This basis is explicitly written out in the lower right cell of Tab. B.1.

1-dimensional irreps:

For the case of 1-dimensional irreps in both the input and output, i.e. $\rho_{\text{out}} = \rho_{\text{in}} = \psi_0^{\mathrm{SO}(2)}$ the kernel has the form $\kappa^{ij} : \mathbb{R}^2 \to \mathbb{R}^{1\times 1}$. As a scalar function in $L^2(\mathbb{R}^2)$, it can be expressed by the Fourier decomposition of its angular part:

$$\kappa(r, \phi) = w_{0,0} + \sum_{\mu=1}^{\infty} \sum_{\gamma \in \{0, \frac{\pi}{2}\}} w_{\mu,\gamma}(r) \cos(\mu\phi + \gamma)$$

As before, we can w.l.o.g. drop the dependency on the radial part as it is not restricted by the constraint. We are therefore considering the basis

$$\left\{ b_{\mu,\gamma}(\phi) = \cos(\mu\phi + \gamma) \;\middle|\; \mu \in \mathbb{N},\ \gamma \in \begin{cases} \{0\} & \text{if } \mu = 0 \\ \{0, \pi/2\} & \text{otherwise} \end{cases} \right\} \tag{B.13}$$

of angular kernels in $L^2(S^1)^{1\times 1}$. The kernel constraint in Eq. (3.12) then requires

$$\begin{aligned} & \kappa(\phi + \theta) &&= \psi_m^{\mathrm{SO}(2)}(r_\theta)\kappa(\phi)\psi_n^{\mathrm{SO}(2)}(r_\theta)^{-1} && \forall \theta, \phi \in [0, 2\pi) \\ \Leftrightarrow\ & \kappa(\phi + \theta) &&= \kappa(\phi) && \forall \theta, \phi \in [0, 2\pi), \end{aligned}$$

i.e. the kernel has to be *invariant* to rotations.

Again, we find the space of all solutions by projecting both sides on the basis defined above. Here, the projection of two kernels is defined through the standard inner product $\langle k_1, k_2 \rangle = \frac{1}{2\pi} \int d\phi k_1(\phi) k_2(\phi)$ on $L^2(S^1)$.

We first consider the projection of the *lhs*:

$$\begin{aligned}\langle b_{\mu',\gamma'},\ R_\theta\kappa\rangle &= \frac{1}{2\pi}\int d\phi\ b_{\mu',\gamma'}(\phi)\,(R_\theta\kappa)\,(\phi)\\ &= \frac{1}{2\pi}\int d\phi\ b_{\mu',\gamma'}(\phi)\kappa(\phi+\theta)\end{aligned}$$

As before we expand the kernel in the linear combination of the basis:

$$\begin{aligned}&= \sum_{\mu,\gamma} w_{\mu,\gamma}\frac{1}{2\pi}\int d\phi\ b_{\mu',\gamma'}(\phi)\cos(\mu\phi+\mu\theta+\gamma)\\ &= \sum_{\mu,\gamma} w_{\mu,\gamma}\frac{1}{2\pi}\int d\phi\ \cos(\mu'\phi+\gamma')\cos(\mu\phi+\mu\theta+\gamma)\end{aligned}$$

With $\cos(\alpha)\cos(\beta) = \frac{1}{2}\left(\cos(\alpha-\beta)+\cos(\alpha+\beta)\right)$ this results in:

$$\begin{aligned}&= \sum_{\mu,\gamma} w_{\mu,\gamma}\frac{1}{2\pi}\int d\phi\ \frac{1}{2}\big(\cos(\mu'\phi+\gamma'-\mu\phi-\mu\theta-\gamma)\\ &\qquad\qquad + \cos(\mu'\phi+\gamma'+\mu\phi+\mu\theta+\gamma)\big)\\ &= \sum_{\mu,\gamma} w_{\mu,\gamma}\frac{1}{2}\big(\frac{1}{2\pi}\int d\phi\cos((\mu'-\mu)\phi+(\gamma'-\gamma)-\mu\theta)\\ &\qquad + \frac{1}{2\pi}\int d\phi\cos((\mu'+\mu)\phi+(\gamma'+\gamma)+\mu\theta)\big)\\ &= \sum_{\mu,\gamma} w_{\mu,\gamma}\frac{1}{2}\left(\delta_{\mu,\mu'}\cos((\gamma'-\gamma)-\mu\theta)+\delta_{\mu,-\mu'}\cos((\gamma'+\gamma)+\mu\theta)\right)\end{aligned}$$

Since $\mu,\mu'\geq 0$ and $\mu=-\mu'$ imply $\mu=\mu'=0$ this simplifies further to

$$= \frac{1}{2}\sum_{\gamma} w_{\mu',\gamma}\left(\cos((\gamma'-\gamma)-\mu'\theta)+\delta_{\mu',0}\cos(\gamma'+\gamma)\right).$$

A projection of the *rhs* yields:

$$\begin{aligned}\langle b_{\mu',\gamma'},\ \kappa\rangle &= \frac{1}{2\pi}\int d\phi\ b_{\mu',\gamma'}(\phi)\kappa(\phi)\\ &= \sum_{\mu,\gamma} w_{\mu,\gamma}\frac{1}{2\pi}\int d\phi\ b_{\mu',\gamma'}(\phi)\cos(\mu\phi+\gamma)\\ &= \sum_{\mu,\gamma} w_{\mu,\gamma}\frac{1}{2\pi}\int d\phi\ \cos(\mu'\phi+\gamma')\cos(\mu\phi+\gamma)\\ &= \frac{1}{2}\sum_{\gamma} w_{\mu',\gamma}\left(\cos(\gamma'-\gamma)+\delta_{\mu',0}\cos((\gamma'+\gamma))\right)\end{aligned}$$

The projections are required to coincide for all rotations:

$$\langle b_{\mu',\gamma'}, R_\theta \kappa \rangle = \langle b_{\mu',\gamma'}, \kappa \rangle \qquad \forall \theta \in [0, 2\pi)$$

$$\sum_\gamma w_{\mu',\gamma} \left(\cos((\gamma' - \gamma) - \mu'\theta) + \delta_{\mu',0} \cos((\gamma' + \gamma)) \right) = \sum_\gamma w_{\mu',\gamma} \left(\cos(\gamma' - \gamma) + \delta_{\mu',0} \cos((\gamma' + \gamma)) \right)$$

$$\forall \theta \in [0, 2\pi)$$

We consider two cases:

- $\mu' = 0$ In this case, the basis in Eq.(B.13) is restricted to the single case $\gamma' = 0$ (as $\gamma' = \frac{\pi}{2}$ and $\mu' = 0$ together lead to a null basis element). Then:

$$\sum_\gamma w_{0,\gamma} \left(\cos(-\gamma) + \cos(\gamma) \right) = \sum_\gamma w_{0,\gamma} \left(\cos(-\gamma) + \cos(\gamma) \right)$$

As $\gamma \in \{0, \frac{\pi}{2}\}$ and $\cos(\pm\frac{\pi}{2}) = 0$:

$$\begin{aligned} &\Leftrightarrow && w_{0,0} \left(\cos(0) + \cos(0) \right) = w_{0,0} \left(\cos(0) + \cos(0) \right) \\ &\Leftrightarrow && w_{0,0} = w_{0,0} \end{aligned}$$

which is always true.

- $\mu' > 0$ Here:

$$\begin{aligned} & \sum_\gamma w_{\mu',\gamma} \cos((\gamma' - \gamma) - \mu'\theta) &&= \sum_\gamma w_{\mu',\gamma} \cos(\gamma' - \gamma) && \forall \theta \in [0, 2\pi) \\ \Leftrightarrow \quad & w_{\mu',0} \cos(\gamma' - \mu'\theta) + w_{\mu',\frac{\pi}{2}} \sin(\gamma' - \mu'\theta) &&= w_{\mu',0} \cos(\gamma') + w_{\mu',\frac{\pi}{2}} \sin(\gamma') && \forall \theta \in [0, 2\pi) \\ \Leftrightarrow \quad & -\mu'\theta &&= 2t\pi && \forall \theta \in [0, 2\pi), \end{aligned}$$

where Eq. (B.8) was used in the last step. From the last equation one can see that μ' must be zero. Since this contradicts the assumption that $\mu' \geq 0$, no solution exists.

This results in a one dimensional basis of isotropic (rotation invariant) kernels

$$\boxed{\mathcal{K}^{\mathrm{SO}(2)}_{\psi_m \leftarrow \psi_n} = \{ b_{0,0}(\phi) = 1 \}} \tag{B.14}$$

for $m = n = 0$, i.e. trivial representations. The basis is presented in the upper left cell of Tab. B.1.

1 and 2-dimensional irreps:

Finally, consider the case of a 1-dimensional irrep in the input and a 2-dimensional irrep in the output, that is, $\rho_{\text{out}} = \psi_m^{\text{SO}(2)}$ and $\rho_{\text{in}} = \psi_0^{\text{SO}(2)}$. The corresponding kernel $\kappa^{ij} : \mathbb{R}^2 \to \mathbb{R}^{2\times 1}$ can be expanded in the following generalized Fourier series on $L^2(\mathbb{R}^2)^{2\times 1}$:

$$\begin{aligned}\kappa(r,\phi) \;=\; \sum_{\mu=0}^{\infty} A_{0,\mu}(r)\begin{bmatrix}\cos(\mu\phi)\\0\end{bmatrix} \;+\; B_{0,\mu}(r)\begin{bmatrix}\sin(\mu\phi)\\0\end{bmatrix}\\ +\; A_{1,\mu}(r)\begin{bmatrix}0\\\cos(\mu\phi)\end{bmatrix} \;+\; B_{1,\mu}(r)\begin{bmatrix}0\\\sin(\mu\phi)\end{bmatrix}\end{aligned}$$

As before, we perform a change of basis to produce a non-sparse basis

$$\kappa(r,\phi) = \sum_{\mu=-\infty}^{\infty} \sum_{\gamma\in\{0,\frac{\pi}{2}\}} w_{\gamma,\mu}(r)\begin{bmatrix}\cos(\mu\phi+\gamma)\\\sin(\mu\phi+\gamma)\end{bmatrix}.$$

Dropping the radial parts as usual, this corresponds to the complete basis :

$$\left\{ b_{\mu,\gamma}(\phi) = \begin{bmatrix}\cos(\mu\phi+\gamma)\\\sin(\mu\phi+\gamma)\end{bmatrix} \;\middle|\; \mu\in\mathbb{Z},\ \gamma\in\left\{0,\frac{\pi}{2}\right\}\right\} \tag{B.15}$$

of angular kernels on $L^2(S^1)^{2\times 1}$.

The constraint in Eq. (3.12) requires the kernel space to satisfy

$$\begin{aligned} & \kappa(\phi+\theta) \;=\; \psi_m^{\text{SO}(2)}(r_\theta)\kappa(\phi)\psi_0^{\text{SO}(2)}(r_\theta)^{-1} && \forall\theta,\phi\in[0,2\pi)\\ \Leftrightarrow\quad & \kappa(\phi+\theta) \;=\; \psi_m^{\text{SO}(2)}(r_\theta)\kappa(\phi) && \forall\theta,\phi\in[0,2\pi). \end{aligned}$$

We again project both sides of this equation on the basis elements defined above where the projection on $L^2(S^1)^{2\times 1}$ is defined by $\langle k_1,k_2\rangle = \frac{1}{2\pi}\int d\phi\; k_1(\phi)^T k_2(\phi)$.

Consider first the projection of the *lhs*

$$\begin{aligned}\langle b_{\mu',\gamma'},\ R_\theta\kappa\rangle &= \frac{1}{2\pi}\int d\phi\; b_{\mu',\gamma'}(\phi)^T\,(R_\theta\kappa)\,(\phi)\\ &= \frac{1}{2\pi}\int d\phi\; b_{\mu',\gamma'}(\phi)^T\kappa(\phi+\theta)\,,\end{aligned}$$

which, after expanding the kernel in terms of the basis reads:

$$
\begin{aligned}
&= \sum_{\mu,\gamma} w_{\mu,\gamma} \frac{1}{2\pi} \int d\phi \; b_{\mu',\gamma'}(\phi)^T \begin{bmatrix} \cos(\mu(\phi+\theta)+\gamma) \\ \sin(\mu(\phi+\theta)+\gamma) \end{bmatrix} \\
&= \sum_{\mu,\gamma} w_{\mu,\gamma} \frac{1}{2\pi} \int d\phi \; \begin{bmatrix} \cos(\mu'\phi+\gamma') & \sin(\mu'\phi+\gamma') \end{bmatrix} \begin{bmatrix} \cos(\mu(\phi+\theta)+\gamma) \\ \sin(\mu(\phi+\theta)+\gamma) \end{bmatrix} \\
&= \sum_{\mu,\gamma} w_{\mu,\gamma} \frac{1}{2\pi} \int d\phi \; \cos((\mu'-\mu)\phi + (\gamma'-\gamma) - \mu\theta).
\end{aligned}
$$

As before, the integral is non-zero only if the frequency is 0, i.e. iff $\mu' - \mu = 0$ and thus:

$$
= \sum_{\gamma} w_{\mu',\gamma} \cos((\gamma'-\gamma) - \mu'\theta)
$$

For the *rhs* we obtain:

$$
\begin{aligned}
&\langle b_{\mu',\gamma'},\ \psi_m^{\mathrm{SO}(2)}(r_\theta)\kappa(\cdot)\rangle \\
&\qquad = \frac{1}{2\pi} \int d\phi \; b_{\mu',\gamma'}(\phi)^T \psi_m(r_\theta)\kappa(\phi) \\
&\qquad = \sum_{\mu,\gamma} w_{\mu,\gamma} \frac{1}{2\pi} \int d\phi \; b_{\mu',\gamma'}(\phi)^T \psi_m(r_\theta) \begin{bmatrix} \cos(\mu\phi+\gamma) \\ \sin(\mu\phi+\gamma) \end{bmatrix} \\
&\qquad = \sum_{\mu,\gamma} w_{\mu,\gamma} \frac{1}{2\pi} \int d\phi \; \begin{bmatrix} \cos(\mu'\phi+\gamma') & \sin(\mu'\phi+\gamma') \end{bmatrix} \psi_m(r_\theta) \begin{bmatrix} \cos(\mu\phi+\gamma) \\ \sin(\mu\phi+\gamma) \end{bmatrix} \\
&\qquad = \sum_{\mu,\gamma} w_{\mu,\gamma} \frac{1}{2\pi} \int d\phi \; \cos(\mu'\phi + \gamma' - \mu\phi - \gamma - m\theta) \\
&\qquad = \sum_{\mu,\gamma} w_{\mu,\gamma} \frac{1}{2\pi} \int d\phi \; \cos((\mu'-\mu)\phi + \gamma' - \gamma - m\theta)\,.
\end{aligned}
$$

The integral is non-zero only if the frequency is 0, i.e. $\mu' - \mu = 0$:

$$
= \sum_{\gamma} w_{\mu',\gamma} \cos(\gamma' - \gamma - m\theta)
$$

Requiring the projections to be equal implies

$$
\begin{aligned}
& & \langle b_{\mu',\gamma'}, R_\theta \kappa \rangle &= \langle b_{\mu',\gamma'}, \psi_m(r_\theta)\kappa(\cdot) \rangle & \forall \theta \in [0, 2\pi) \\
\Leftrightarrow & & \sum_\gamma w_{\mu',\gamma} \cos(\gamma' - \gamma - \mu'\theta) &= \sum_\gamma w_{\mu',\gamma} \cos(\gamma' - \gamma - m\theta) & \forall \theta \in [0, 2\pi) \\
\Leftrightarrow & w_{\mu',0} \cos(\gamma' - \mu'\theta) + w_{\mu',\frac{\pi}{2}} \sin(\gamma' - \mu'\theta) &= w_{\mu',0} \cos(\gamma' - m\theta) + w_{\mu',\frac{\pi}{2}} \sin(\gamma' - m\theta) & \\
& & & & \forall \theta \in [0, 2\pi) \\
\Leftrightarrow & & \gamma' - \mu'\theta &= \gamma' - m\theta + 2t\pi & \forall \theta \in [0, 2\pi) \\
\Leftrightarrow & & \mu'\theta &= m\theta + 2t\pi & \forall \theta \in [0, 2\pi),
\end{aligned}
$$

where we made use of Eq. (B.8) once again. It follows that $\mu' = m$, resulting in the two-dimensional basis

$$
\mathcal{K}^{\mathrm{SO}(2)}_{\psi_m \leftarrow \psi_n} = \left\{ b_{m,\gamma}(\phi) = \begin{bmatrix} \cos(m\phi + \gamma) \\ \sin(m\phi + \gamma) \end{bmatrix} \;\middle|\; \gamma \in \left\{ 0, \frac{\pi}{2} \right\} \right\} \tag{B.16}
$$

of equivariant kernels for $m > 0$ and $n = 0$. This basis is explicitly given in the lower left cell of Tab. B.1.

2 and 1-dimensional irreps:

The case for 2-dimensional input and 1-dimensional output representations, i.e. $\rho_{\text{in}} = \psi_n^{\mathrm{SO}(2)}$ and $\rho_{\text{out}} = \psi_0^{\mathrm{SO}(2)}$, is identical to the previous one up to a transpose. The final two-dimensional basis for $m = 0$ and $n > 0$ is therefore given by

$$
\mathcal{K}^{\mathrm{SO}(2)}_{\psi_m \leftarrow \psi_n} = \left\{ b_{n,\gamma}(\phi) = \begin{bmatrix} \cos(n\phi + \gamma) & \sin(n\phi + \gamma) \end{bmatrix} \;\middle|\; \gamma \in \left\{ 0, \frac{\pi}{2} \right\} \right\} \tag{B.17}
$$

as shown in the upper right cell of Tab. B.1.

B.2.3 Reflection Group

The action of the reflection group $(\{\pm 1\}, *)$ on $\mathbb{R}^2$ depends on a choice of reflection axis, which we specify by an angle β. More precisely, the element $s \in (\{\pm 1\}, *)$ acts on $\boldsymbol{x} = (r, \phi) \in \mathbb{R}^2$ as

$$s.\boldsymbol{x}(r,\phi) \;:=\; \boldsymbol{x}(r, s.\phi) \;:=\; \boldsymbol{x}(r, 2\beta\delta_{s,-1} + s\phi) \;=\; \begin{cases} \boldsymbol{x}(r,\phi) & \text{if } s = \;\; 1 \\ \boldsymbol{x}(r, 2\beta - \phi) & \text{if } s = \text{-}1\,. \end{cases}$$

The kernel constraint for the reflection group is therefore being made explicit by

$$\begin{aligned} & \kappa(r, s.\phi) \;\; = \rho_{\text{out}}(s)\kappa(r,\phi)\rho_{\text{in}}(s)^{-1} && \forall s \in (\{\pm 1\}, *), \phi \in \; [0, 2\pi) \\ \Leftrightarrow \quad & \kappa(r, \delta_{s,-1}2\beta + s\phi) \;\; = \rho_{\text{out}}(s)\kappa(r,\phi)\rho_{\text{in}}(s)^{-1} && \forall s \in (\{\pm 1\}, *), \phi \in \; [0, 2\pi) \\ \Leftrightarrow \quad & \kappa(r, \delta_{s,-1}2\beta + s\phi) \;\; = \rho_{\text{out}}(s)\kappa(r,\phi)\rho_{\text{in}}(s) && \forall s \in (\{\pm 1\}, *), \phi \in \; [0, 2\pi)\,, \end{aligned}$$

where we used the identity $s^{-1} = s$. For $s = +1$ the constraint is trivially true. We will thus in the following consider the case $s = -1$, that is,

$$\kappa(r, 2\beta - \phi) \;\; = \;\; \rho_{\text{out}}(-1)\kappa(r,\phi)\rho_{\text{in}}(-1) \qquad \forall \phi \in \; [0, 2\pi)\,.$$

In order to simplify this constraint further we define a transformed kernel $\kappa'(r, \phi) \;:=\; \kappa(r, \phi + \beta)$ which is oriented relative to the reflection axis. The transformed kernel is then required to satisfy

$$\kappa'(r, \beta - \phi) \;\; = \;\; \rho_{\text{out}}(-1)\kappa'(r, \phi - \beta)\rho_{\text{in}}(-1) \quad \forall \phi \in \; [0, 2\pi)\,,$$

which, with the change of variables $\phi' \;=\; \phi - \beta$, reduces to the constraint for equivariance under reflections around the x-axis, i.e. the case for $\beta = 0$:

$$\kappa'(r, -\phi') = \;\; \rho_{\text{out}}(-1)\kappa'(r, \phi')\rho_{\text{in}}(-1) \qquad \forall \phi' \in \; [0, 2\pi)\,.$$

As a consequence we can retrieve kernels equivariant under reflections around the β-axis through

$$\kappa(r,\phi) \;\; := \;\; \kappa'(r, \phi - \beta)\,.$$

We will therefore without loss of generality consider the case $\beta \;=\; 0$ only in the following.

1-dimensional irreps:

The reflection group $(\{\pm 1\}, *)$ has only two irreps, namely the trivial representation $\psi_0^{(\{\pm1\},*)}(s) = 1$ and the sign-flip representation $\psi_1^{(\{\pm1\},*)}(s) = s$. Therefore only the 1-dimensional case with a kernel of form $\kappa : \mathbb{R}^2 \to \mathbb{R}^{1\times 1}$ exists. Note that we can write the irreps out as $\psi_f^{(\{\pm1\},*)}(s) = s^f$, in particular $\psi_f^{(\{\pm1\},*)}(-1) = (-1)^f$.

Consider the output and input irreps $\rho_{\text{out}} = \psi_i^{(\{\pm1\},*)}$ and $\rho_{\text{in}} = \psi_j^{(\{\pm1\},*)}$ (with $i, j \in \{0, 1\}$) and the usual 1-dimensional Fourier basis for scalar functions in $L^2(S^1)$ as before:

$$\left\{ b_{\mu,\gamma}(\phi) = \cos(\mu\phi + \gamma) \;\middle|\; \mu \in \mathbb{N},\ \gamma \in \begin{cases} \{0\} & \text{if } \mu = 0 \\ \{0, \pi/2\} & \text{otherwise} \end{cases} \right\} \tag{B.18}$$

Defining the reflection operator S by its action $(S\,\kappa)\,(\phi) := \kappa(-\phi)$, we require the projections of both sides of the kernel constraint on the same basis element to be equal as usual. Specifically, for a particular basis $b_{\mu',\gamma'}$:

$$\langle b_{\mu',\gamma'}, S\,\kappa\rangle = \langle b_{\mu',\gamma'}, \psi_i^{(\{\pm1\},*)}(-1)\kappa(\cdot)\psi_j^{(\{\pm1\},*)}(-1)\rangle$$

The *lhs* implies

$$\begin{aligned}
\langle b_{\mu',\gamma'}, S\,\kappa\rangle &= \sum_{\mu,\gamma} w_{\mu,\gamma} \frac{1}{2\pi} \int d\phi \;\; b_{\mu',\gamma'}(\phi) b_{\mu,\gamma}(-\phi) \\
&= \sum_{\mu,\gamma} w_{\mu,\gamma} \frac{1}{2\pi} \int d\phi \;\; \cos(\mu'\phi + \gamma') \cos(-\mu\phi + \gamma) \\
&= \sum_{\mu,\gamma} w_{\mu,\gamma} \frac{1}{2\pi} \int d\phi \;\; \frac{1}{2} \left(\cos((\mu' + \mu)\phi + (\gamma' - \gamma)) + \cos((\mu' - \mu)\phi + (\gamma' + \gamma))\right) \\
&= \sum_{\gamma} w_{\mu',\gamma} \frac{1}{2} \left(\cos(\gamma' + \gamma) + \delta_{\mu',0} \cos(\gamma' - \gamma)\right)
\end{aligned}$$

while the *rhs* leads to

$$\begin{aligned}
&\langle b_{\mu',\gamma'}, \psi_m^{(\{\pm1\},*)}(-1)\kappa(\cdot)\psi_n^{(\{\pm1\},*)}(-1)\rangle \\
&\qquad = \sum_{\mu,\gamma} w_{\mu,\gamma} \frac{1}{2\pi} \int d\phi \;\; b_{\mu',\gamma'}(\phi) \psi_i^{(\{\pm1\},*)}(-1) b_{\mu,\gamma}(\phi) \psi_j^{(\{\pm1\},*)}(-1) \\
&\qquad = \sum_{\mu,\gamma} w_{\mu,\gamma} \frac{1}{2\pi} \int d\phi \;\; \cos(\mu'\phi + \gamma')(-1)^i \cos(\mu\phi + \gamma)(-1)^j \\
&\qquad = (-1)^{i+j} \sum_{\mu,\gamma} w_{\mu,\gamma} \frac{1}{2\pi} \int d\phi \;\; \cos(\mu'\phi + \gamma') \cos(\mu\phi + \gamma) \\
&\qquad = (-1)^{i+j} \sum_{\gamma} w_{\mu',\gamma} \frac{1}{2} \left(\cos(\gamma' - \gamma) + \delta_{\mu',0} \cos(\gamma' + \gamma)\right) .
\end{aligned}$$

Now, we require both sides to be equal, that is,

$$\sum_{\gamma} w_{\mu',\gamma} \frac{1}{2} \left(\cos(\gamma' + \gamma) + \delta_{\mu',0} \cos(\gamma' - \gamma)\right) = (-1)^{i+j} \sum_{\gamma} w_{\mu',\gamma} \frac{1}{2} \left(\cos(\gamma' - \gamma) + \delta_{\mu',0} \cos(\gamma' + \gamma)\right)$$

and again consider two cases for μ':

- $\mu' = 0$ The basis in Eq.(B.18) is restricted to the single case $\gamma' = 0$. Hence:

$$\sum_{\gamma} w_{0,\gamma} \frac{1}{2} \left(\cos(\gamma) + \cos(-\gamma)\right) = (-1)^{i+j} \sum_{\gamma} w_{0,\gamma} \frac{1}{2} \left(\cos(-\gamma) + \cos(\gamma)\right)$$

  As $\gamma \in \{0, \frac{\pi}{2}\}$ and $\cos(\pm\frac{\pi}{2}) = 0$:

$$\begin{aligned} \Leftrightarrow \quad & w_{0,0} \frac{1}{2} \left(\cos(0) + \cos(0)\right) = (-1)^{i+j} w_{0,0} \frac{1}{2} \left(\cos(-0) + \cos(0)\right) \\ \Leftrightarrow \quad & w_{0,0} = (-1)^{i+j} w_{0,0} \end{aligned}$$

  Which is always true when $i = j$, while it enforces $w_{0,0} = 0$ when $i \neq j$.

- $\mu' > 0$ In this case we get:

$$\begin{aligned} & \sum_{\gamma} w_{\mu',\gamma} \frac{1}{2} \cos(\gamma' + \gamma) = (-1)^{i+j} \sum_{\gamma} w_{\mu',\gamma} \frac{1}{2} \cos(\gamma' - \gamma) \\ \Leftrightarrow \quad & (1 - (-1)^{i+j}) w_{\mu',0} \cos(\gamma') = (1 + (-1)^{i+j}) w_{\mu,\frac{\pi}{2}} \sin(\gamma') \end{aligned}$$

  If $i + j \equiv 0 \mod 2$, the equation becomes $\sin(\gamma') = 0$ and, so, $\gamma' = 0$. Otherwise, it becomes $\cos(\gamma') = 0$, which means $\gamma' = \frac{\pi}{2}$. Shortly, $\gamma' = (i + j \mod 2)\frac{\pi}{2}$.

As a result, only half of the basis for $\beta = 0$ is preserved:

$$\mathcal{K}^{(\{\pm 1\},*),\beta=0}_{\psi_i \leftarrow \psi_j} = \left\{ b_{\mu,\gamma}(\phi) = \cos(\mu\phi + \gamma) \;\middle|\; \mu \in \mathbb{N},\ \gamma = (i + j \mod 2)\frac{\pi}{2},\ \mu > 0 \vee \gamma = 0 \right\} \tag{B.19}$$

The solution for a general reflection axis β is therefore given by

$$\mathcal{K}^{(\{\pm 1\},*),\beta}_{\psi_i \leftarrow \psi_j} = \left\{ b_{\mu,\gamma}(\phi) = \cos(\mu(\phi - \beta) + \gamma) \;\middle|\; \mu \in \mathbb{N}, \gamma = (i + j \mod 2)\frac{\pi}{2}, \mu > 0 \vee \gamma = 0 \right\} \tag{B.20}$$

which is visualized in Tab. B.3 for the different cases of irreps for $i, j \in \{0, 1\}$.

B.2.4 Orthogonal Group $\mathrm{O}(2)$

The orthogonal group $\mathrm{O}(2)$ is the semi-direct product between the rotation group $\mathrm{SO}(2)$ and the reflection group $(\{\pm 1\}, *)$, i.e. $\mathrm{O}(2) \cong \mathrm{SO}(2) \rtimes (\{\pm 1\}, *)$. This justifies a decomposition of the constraint on $\mathrm{O}(2)$-equivariant kernels as the union of the constraints for rotations and reflections. Consequently, the space of $\mathrm{O}(2)$-equivariant kernels is the intersection between the spaces of $\mathrm{SO}(2)$- and reflection-equivariant kernels.

Proof

Sufficiency:

Assume a rotation- and reflection-equivariant kernel, i.e. a kernel which for all $r \in \mathbb{R}_0^+$ and $\phi \in [0, 2\pi)$ satisfies

$$\begin{aligned} \kappa(r, r_\theta \phi) &= \left(\mathrm{Res}^{\mathrm{O}(2)}_{\mathrm{SO}(2)} \rho_{\text{out}}\right)(r_\theta)\, \kappa(r, \phi)\, \left(\mathrm{Res}^{\mathrm{O}(2)}_{\mathrm{SO}(2)} \rho_{\text{in}}\right)^{-1}(r_\theta) \qquad \forall\, r_\theta \in \mathrm{SO}(2) \\ &= \rho_{\text{out}}(r_\theta)\, \kappa(r, \phi)\, \rho_{\text{in}}^{-1}(r_\theta) \end{aligned}$$

and

$$\begin{aligned} \kappa(r, s\phi) &= \left(\mathrm{Res}^{\mathrm{O}(2)}_{(\{\pm 1\}, *)} \rho_{\text{out}}\right)(s)\, \kappa(r, \phi)\, \left(\mathrm{Res}^{\mathrm{O}(2)}_{(\{\pm 1\}, *)} \rho_{\text{in}}\right)^{-1}(s) \quad \forall\, s \in (\{\pm 1\}, *) \\ &= \rho_{\text{out}}(s)\, \kappa(r, \phi)\, \rho_{\text{in}}^{-1}(s)\,. \end{aligned}$$

Then, for any $h = r_\theta s \in \mathrm{O}(2)$, the kernel constraint becomes:

$$\begin{aligned} & \kappa(r, h\phi) = \rho_{\text{out}}(h)\, \kappa(r, \phi)\, \rho_{\text{in}}^{-1}(h) \\ \Leftrightarrow \quad & \kappa(r, r_\theta s \phi) = \rho_{\text{out}}(r_\theta s)\, \kappa(r, \phi)\, \rho_{\text{in}}^{-1}(r_\theta s) \\ \Leftrightarrow \quad & \kappa(r, r_\theta s \phi) = \rho_{\text{out}}(r_\theta)\rho_{\text{out}}(s)\, \kappa(r, \phi)\, \rho_{\text{in}}^{-1}(s)\rho_{\text{in}}^{-1}(r_\theta)\,. \end{aligned}$$

Applying reflection-equivariance this equation simplifies to

$$\Leftrightarrow \quad \kappa(r, r_\theta s \phi) = \rho_{\text{out}}(r_\theta)\, \kappa(r, s\phi)\, \rho_{\text{in}}^{-1}(r_\theta)\,,$$

which, applying rotation-equivariance yields

$$\Leftrightarrow \quad \kappa(r, r_\theta s \phi) = \kappa(r, r_\theta s \phi)\,.$$

Hence any kernel satisfying both $\mathrm{SO}(2)$ and reflection constraints is also $\mathrm{O}(2)$ equivariant.

Necessity:

Trivially, $\mathrm{O}(2)$ equivariance implies equivariance under $\mathrm{SO}(2)$ and reflections. Specifically, for any $r \in \mathbb{R}_0^+$ and $\phi \in [0, 2\pi)$, the equation

$$\kappa(r, h\phi) \;=\; \rho_{\text{out}}(h)\; \kappa(r, \phi)\; \rho_{\text{in}}^{-1}(h) \qquad\qquad \forall\; h = r_\theta s \in \mathrm{O}(2)$$

implies

$$\begin{aligned}\kappa(r, r_\theta\phi) \;&=\; \rho_{\text{out}}(r_\theta)\; \kappa(r, \phi)\; \rho_{\text{in}}^{-1}(r_\theta) \\ &=\; \left(\mathrm{Res}^{\mathrm{O}(2)}_{\mathrm{SO}(2)}\, \rho_{\text{out}}\right)(r_\theta)\; \kappa(r, \phi)\; \left(\mathrm{Res}^{\mathrm{O}(2)}_{\mathrm{SO}(2)}\, \rho_{\text{in}}\right)^{-1}(r_\theta) \qquad \forall\; r_\theta \in \mathrm{SO}(2)\end{aligned}$$

and

$$\begin{aligned}\kappa(r, s\phi) \;&=\; \rho_{\text{out}}(s)\; \kappa(r, \phi)\; \rho_{\text{in}}^{-1}(s) \\ &=\left(\mathrm{Res}^{\mathrm{O}(2)}_{(\{\pm 1\},*)}\, \rho_{\text{out}}\right)(s)\; \kappa(r, \phi)\; \left(\mathrm{Res}^{\mathrm{O}(2)}_{(\{\pm 1\},*)}\, \rho_{\text{in}}\right)^{-1}(s) \quad \forall\; s \in (\{\pm 1\}, *).\end{aligned}$$

This observation allows us to derive the kernel space for $\mathrm{O}(2)$ by intersecting the previously derived kernel space of $\mathrm{SO}(2)$ with the kernel space of the reflection group:

$$\begin{aligned}\mathcal{K}^{\mathrm{O}(2)}_{\rho_{\text{out}} \leftarrow \rho_{\text{in}}} = \{\kappa \mid \kappa(r, h.\phi) = \rho_{\text{out}}(h)\kappa(r, \phi)\rho_{\text{in}}(h)^{-1}\; \forall\; h \in \mathrm{O}(2)\} \\ = \quad \{\kappa \mid \kappa(r, r_\theta.\phi) = \rho_{\text{out}}(r_\theta)\kappa(r, \phi)\rho_{\text{in}}(r_\theta)^{-1}\; \forall\; r_\theta \in \mathrm{SO}(2)\} \\ \cap\; \{\kappa \mid \kappa(r,\ \ s.\phi) = \ \ \rho_{\text{out}}(s)\kappa(r, \phi)\rho_{\text{in}}(s)^{-1}\ \ \forall\; s \in (\{\pm 1\}, *)\}\end{aligned}$$

As $\mathrm{O}(2)$ contains all rotations, it does also contain all reflection axes. Without loss of generality, we define $s \in \mathrm{O}(2)$ as the reflection along the x-axis. A reflection along any other axis β is associated with the group element $r_{2\beta}s \in \mathrm{O}(2)$, i.e. the combination of a reflection with a rotation of 2β. As a result, we consider the basis for reflection equivariant kernels derived for $\beta = 0$ in Eq. (B.19).

Therefore, to derive a basis associated to a pair of input and output representations ρ_{in} and ρ_{out}, we restrict the representations to $\mathrm{SO}(2)$ and the reflection group, compute the two bases using the results found in Appendix B.2.2 and in Appendix B.2.3, and, finally, take their intersection.

2-dimensional irreps:

The restriction of any 2-dimensional irrep $\psi^{\mathrm{O}(2)}_{1,n}$ of $\mathrm{O}(2)$ to the reflection group

decomposes into the direct sum of the two 1-dimensional irreps of the reflection group, i.e. into the diagonal matrix

$$\mathrm{Res}^{\mathrm{O}(2)}_{(\{\pm 1\},*)}\,\psi_{1,n}(s) = \left(\psi_0^{(\{\pm 1\},*)} \oplus \psi_1^{(\{\pm 1\},*)}\right)(s) = \begin{bmatrix} \psi_0^{(\{\pm 1\},*)}(s) & 0 \\ 0 & \psi_1^{(\{\pm 1\},*)}(s) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & s \end{bmatrix}.$$

It follows that the restricted kernel space constraint decomposes into independent constraints on each entry of the original kernel. Specifically, for output and input representations $\rho_{\text{out}} = \psi^{\mathrm{O}(2)}_{1,m}$ and $\rho_{\text{in}} = \psi^{\mathrm{O}(2)}_{1,n}$, the constraint becomes

$$\kappa(s.x) = \underbrace{\left(\begin{array}{c|c} \psi_0^{(\{\pm 1\},*)}(s) & \\ \hline & \psi_1^{(\{\pm 1\},*)}(s) \end{array}\right)}_{\mathrm{Res}^{\mathrm{O}(2)}_{(\{\pm 1\},*)}\,\rho_{\text{out}}(s)} \cdot \underbrace{\left(\begin{array}{c|c} \kappa^{00} & \kappa^{01} \\ \hline \kappa^{10} & \kappa^{11} \end{array}\right)}_{\kappa(x)} \cdot \underbrace{\left(\begin{array}{c|c} \psi_0^{(\{\pm 1\},*)}(s)^{-1} & \\ \hline & \psi_1^{(\{\pm 1\},*)}(s)^{-1} \end{array}\right)}_{\mathrm{Res}^{\mathrm{O}(2)}_{(\{\pm 1\},*)}\,\rho_{\text{in}}(s)}$$

We can therefore solve for a basis for each entry individually following Appendix B.2.3 to obtain the complete basis

$$\{b^{00}_{\mu,0}\ (\phi) = \begin{bmatrix} \cos(\mu\phi) & 0 \\ 0 & 0 \end{bmatrix} \Big|\ \mu \in \mathbb{N}\ \} \cup \{b^{01}_{\mu,\frac{\pi}{2}}\ (\phi) = \begin{bmatrix} 0 & \sin(\mu\phi) \\ 0 & 0 \end{bmatrix} \Big|\ \mu \in \mathbb{N}^+\} \cup$$
$$\{b^{10}_{\mu,\frac{\pi}{2}}\ (\phi) = \begin{bmatrix} 0 & 0 \\ \sin(\mu\phi) & 0 \end{bmatrix} \Big|\ \mu \in \mathbb{N}^+\} \cup \{b^{11}_{\mu,0}\ (\phi) = \begin{bmatrix} 0 & 0 \\ 0 & \cos(\mu\phi) \end{bmatrix} \Big|\ \mu \in \mathbb{N}\ \}.$$

Through the same change of basis applied in the first paragraph of Appendix B.2.2, we get the following equivalent basis for the same space:

$$\begin{aligned} &\left\{ b_{\mu,s}(\phi) = \begin{bmatrix} \cos(\mu\phi) & -\sin(\mu\phi) \\ \sin(\mu\phi) & \cos(\mu\phi) \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & s \end{bmatrix} \right\}_{\mu\in\mathbb{Z}, s\in\{\pm 1\}} \\ &= \left\{ b_{\mu,s}(\phi) = \psi(\mu\phi)\xi(s) \right\}_{\mu\in\mathbb{Z}, s\in\{\pm 1\}}. \end{aligned} \tag{B.21}$$

On the other hand, 2-dimensional $\mathrm{O}(2)$ representations restrict to the $\mathrm{SO}(2)$ irreps of the corresponding frequency, i.e.

$$\mathrm{Res}^{\mathrm{O}(2)}_{\mathrm{SO}(2)}\,\rho_{\text{in}} = \mathrm{Res}^{\mathrm{O}(2)}_{\mathrm{SO}(2)}\,\psi^{\mathrm{O}(2)}_{1,n}(r_\theta) = \psi^{\mathrm{SO}(2)}_n(r_\theta)$$

and

$$\mathrm{Res}^{\mathrm{O}(2)}_{\mathrm{SO}(2)}\,\rho_{\text{out}} = \mathrm{Res}^{\mathrm{O}(2)}_{\mathrm{SO}(2)}\,\psi^{\mathrm{O}(2)}_{1,m}(r_\theta) = \psi^{\mathrm{SO}(2)}_m(r_\theta).$$

In Appendix B.2.2, a basis for $\mathrm{SO}(2)$-equivariant kernels with respect to a $\psi^{\mathrm{SO}(2)}_n$ input field and $\psi^{\mathrm{SO}(2)}_m$ output field was derived starting from the basis in Eq. (B.9).

Notice that the basis of reflection-equivariant kernels in Eq. (B.21) contains exactly half of the elements in Eq. (B.9), indexed by $\gamma = 0$. A basis for $\mathrm{O}(2)$-equivariant kernels can be found by repeating the derivations in Appendix B.2.2 for $\mathrm{SO}(2)$-equivariant kernels using only the subset in Eq. (B.21) of reflection-equivariant kernels. The resulting two-dimensional $\mathrm{O}(2)$-equivariant basis, which includes the $\mathrm{SO}(2)$-equivariance conditions ($\mu = m - sn$) and the reflection-equivariance conditions ($\gamma = 0$), is given by

$$\mathcal{K}^{\mathrm{O}(2)}_{\psi_{i,m} \leftarrow \psi_{j,n}} = \left\{ b_{\mu,0,s}(\phi) = \psi(\mu\phi)\xi(s) \;\middle|\; \mu = m - sn, s \in \{\pm 1\} \right\}, \tag{B.22}$$

where $i = j = 1$ and $m, n > 0$. See the bottom right cell in Tab. B.2.

1-dimensional irreps:

$\mathrm{O}(2)$ has two 1-dimensional irreps $\psi^{\mathrm{O}(2)}_{0,0}$ and $\psi^{\mathrm{O}(2)}_{1,0}$ (see Sec. 2.7.2). Both are trivial under rotations and each of them corresponds to one of the two reflection group's irreps, i.e.

$$\mathrm{Res}^{\mathrm{O}(2)}_{(\{\pm 1\},*)} \psi^{\mathrm{O}(2)}_{i,0}(s) = \psi^{(\{\pm 1\},*)}_i(s) = s^i$$

and

$$\mathrm{Res}^{\mathrm{O}(2)}_{\mathrm{SO}(2)} \psi^{\mathrm{O}(2)}_{i,0}(r_\theta) = \psi^{\mathrm{SO}(2)}_0(r_\theta) = 1.$$

Considering output and input representations $\rho_{\text{out}} = \psi^{\mathrm{O}(2)}_{i,0}$ and $\rho_{\text{in}} = \psi^{\mathrm{O}(2)}_{j,0}$, it follows that:

$$\begin{aligned}
\mathrm{Res}^{\mathrm{O}(2)}_{(\{\pm 1\},*)} \rho_{\text{in}} &= \mathrm{Res}^{\mathrm{O}(2)}_{(\{\pm 1\},*)} \psi^{\mathrm{O}(2)}_{j,0} = \psi^{(\{\pm 1\},*)}_j \\
\mathrm{Res}^{\mathrm{O}(2)}_{(\{\pm 1\},*)} \rho_{\text{out}} &= \mathrm{Res}^{\mathrm{O}(2)}_{(\{\pm 1\},*)} \psi^{\mathrm{O}(2)}_{i,0} = \psi^{(\{\pm 1\},*)}_i \\
\mathrm{Res}^{\mathrm{O}(2)}_{\mathrm{SO}(2)} \rho_{\text{in}} &= \mathrm{Res}^{\mathrm{O}(2)}_{\mathrm{SO}(2)} \psi^{\mathrm{O}(2)}_{j,0} = \psi^{\mathrm{SO}(2)}_0 \\
\mathrm{Res}^{\mathrm{O}(2)}_{\mathrm{SO}(2)} \rho_{\text{out}} &= \mathrm{Res}^{\mathrm{O}(2)}_{\mathrm{SO}(2)} \psi^{\mathrm{O}(2)}_{i,0} = \psi^{\mathrm{SO}(2)}_0
\end{aligned}$$

In order to solve the $\mathrm{O}(2)$ kernel constraint consider again the reflectional constraint and the $\mathrm{SO}(2)$ constraint. Bases for reflection-equivariant kernels with above representations were derived in Appendix B.2.3 and are shown in Eq. (B.19). These bases form a subset of the Fourier basis in Eq. (B.13) which is being indexed by $\gamma = (i + j \mod 2)\frac{\pi}{2}$. On the other hand, the full Fourier basis was restricted by the

$\mathrm{SO}(2)$ constraint to satisfy $\mu = 0$ and therefore $\gamma = 0$, see Eq. (B.14). Intersecting both constraints therefore implies $i = j$, resulting in the $\mathrm{O}(2)$-equivariant basis

$$\mathcal{K}^{\mathrm{O}(2)}_{\psi_{i,m} \leftarrow \psi_{j,n}} = \begin{cases} \{b_{0,0}(\phi) = 1\} & \text{if } i = j, \\ \emptyset & \text{else} \end{cases} \tag{B.23}$$

for $m, n = 0$ which is shown in the top left cell in Tab. B.2.

1 and 2-dimensional irreps:

Now we consider the 2-dimensional output representation $\rho_{\text{out}} = \psi^{\mathrm{O}(2)}_{1,m}$ and the 1-dimensional input representation $\rho_{\text{in}} = \psi^{\mathrm{O}(2)}_{j,0}$.

Following the same strategy as before we find the reflectional constraints for these representations to be given by

$$\kappa(s.x) = \underbrace{\begin{pmatrix} \psi_0^{(\{\pm 1\},*)}(s) & \\ & \psi_1^{(\{\pm 1\},*)}(s) \end{pmatrix}}_{\mathrm{Res}^{\mathrm{O}(2)}_{(\{\pm 1\},*)} \rho_{\text{out}}(s)} \cdot \underbrace{\begin{pmatrix} \kappa^{00} \\ \kappa^{10} \end{pmatrix}}_{\kappa(x)} \cdot \underbrace{\left(\psi_j^{(\{\pm 1\},*)}(s)^{-1} \right)}_{\mathrm{Res}^{\mathrm{O}(2)}_{(\{\pm 1\},*)} \rho_{\text{in}}(s)},$$

and therefore to decompose into two independent constraints on the entries κ^{00} and κ^{10}. Solving for a basis for each entry and taking their union as before we get[2]

$$\left\{ b^{00}_\mu(\phi) = \begin{bmatrix} \cos(\mu\phi + j\frac{\pi}{2}) \\ 0 \end{bmatrix} \right\}_{\mu \in \mathbb{N}} \cup \left\{ b^{10}_\mu(\phi) = \begin{bmatrix} 0 \\ \sin(\mu\phi - j\frac{\pi}{2}) \end{bmatrix} \right\}_{\mu \in \mathbb{N}},$$

which, through a change of basis, can be rewritten as

$$\left\{ b_{\mu, j\frac{\pi}{2}}(\phi) = \begin{bmatrix} \cos(\mu\phi + j\frac{\pi}{2}) \\ \sin(\mu\phi + j\frac{\pi}{2}) \end{bmatrix} \right\}_{\mu \in \mathbb{Z}}. \tag{B.24}$$

We intersect this basis with the basis of $\mathrm{SO}(2)$ equivariant kernels with respect to a $\mathrm{Res}^{\mathrm{O}(2)}_{\mathrm{SO}(2)} \rho_{\text{in}} = \psi_0^{\mathrm{SO}(2)}$ input field and $\mathrm{Res}^{\mathrm{O}(2)}_{\mathrm{SO}(2)} \rho_{\text{out}} = \psi_m^{\mathrm{SO}(2)}$ output field as derived in Appendix B.2.2. Both constraints, that is, $\gamma = j\frac{\pi}{2}$ for the reflection group and $\mu = m$

[2] Notice that for $\mu = 0$ some of the elements of the set are zero and are therefore not part of the basis. We omit this detail to reduce clutter.

for $\mathrm{SO}(2)$ (see Eq. (B.15)), define the one-dimensional basis for $\mathrm{O}(2)$-equivariant kernels for $n = 0$, $m > 0$ and $i = 1$ as

$$\mathcal{K}^{\mathrm{O}(2)}_{\psi_{i,m} \leftarrow \psi_{j,n}} = \left\{ b_{\mu, j\frac{\pi}{2}}(\phi) = \begin{bmatrix} \cos(\mu\phi + j\frac{\pi}{2}) \\ \sin(\mu\phi + j\frac{\pi}{2}) \end{bmatrix} \;\middle|\; \mu = m \right\}, \tag{B.25}$$

see the bottom left cell in Tab. B.2.

2 and 1-dimensional irreps:

As already argued in the case for $\mathrm{SO}(2)$, the basis for 2-dimensional input representations $\rho_{\text{in}} = \psi^{\mathrm{O}(2)}_{1,n}$ and 1-dimensional output representations $\rho_{\text{out}} = \psi^{\mathrm{O}(2)}_{i,0}$ is identical to the previous basis up to a transpose, i.e. it is given by

$$\mathcal{K}^{\mathrm{O}(2)}_{\psi_{i,m} \leftarrow \psi_{j,n}} = \left\{ b_{\mu, i\frac{\pi}{2}}(\phi) = \begin{bmatrix} \cos(\mu\phi + i\frac{\pi}{2}) & \sin(\mu\phi + i\frac{\pi}{2}) \end{bmatrix} \;\middle|\; \mu = n \right\}, \tag{B.26}$$

where $j = 1$, $n > 0$ and $m = 0$. This case is visualized in the top right cell of Tab. B.2.

B.2.5 Cyclic Group C_N

The derivations for C_N coincide mostly with the derivations done for $\mathrm{SO}(2)$ with the difference that the projected constraints need to hold for discrete angles $\theta \in \{p\frac{2\pi}{N} \,|\, p = 0, \dots, N-1\}$ only. Furthermore, C_N has one additional 1-dimensional irrep of frequency$N/2$ if (and only if) N is even.

2-dimensional irreps:

During the derivation of the solutions for $\mathrm{SO}(2)$'s 2-dimensional irreps in Appendix B.2.2, we assumed continuous angles only in the very last step. The constraint in Eq. (B.11) therefore holds for C_N as well. Specifically, it demands that for each $\theta \in \{p\frac{2\pi}{N} \,|\, p = 0, \dots, N-1\}$ there exists a $t \in \mathbb{Z}$ such that:

$$\begin{aligned} & (\mu' - (m - ns'))\theta = 2t\pi \\ \Leftrightarrow \quad & (\mu' - (m - ns'))p\frac{2\pi}{N} = 2t\pi \\ \Leftrightarrow \quad & (\mu' - (m - ns'))p = tN \end{aligned}$$

The last result corresponds to a system of N linear congruence equations modulo N which require N to divide $(\mu' - (m - ns'))p$ for each non-negative integer p smaller than N. Note that solutions of the constraint for $p = 1$ already satisfy the constraints for $p \in 2, \ldots, N-1$ such that it is sufficient to consider

$$\begin{aligned} & (\mu' - (m - ns'))1 = tN \\ \Leftrightarrow \quad & \mu' = m - ns' + tN\,. \end{aligned}$$

The resulting basis

$$\boxed{\mathcal{K}^{\mathrm{C}_N}_{\psi_m \leftarrow \psi_n} = \left\{ b_{\mu,\gamma,s}(\phi) = \psi(\mu\phi + \gamma)\xi(s) \,\middle|\, \mu = m - sn + tN, \gamma \in \left\{0, \frac{\pi}{2}\right\} s \in \{\pm 1\} \right\}_{t \in \mathbb{Z}}} \tag{B.27}$$

for $m, n > 0$ thus coincides mostly with the basis B.12 for $\mathrm{SO}(2)$ but contains solutions for aliased frequencies, defined by adding tN. The bottom right cell in Tab. B.4 gives the explicit form of this basis.

1-dimensional irreps:

The same trick could be applied to solve the remaining three cases. However, since C_N has an additional one dimensional irrep of frequency $N/2$ for even N it is convenient to rederive all cases. We therefore consider $\rho_{\text{out}} = \psi^{\mathrm{C}_N}_m$ and $\rho_{\text{in}} = \psi^{\mathrm{C}_N}_n$, where $m, n \in \{0, N/2\}$. Note that $\psi^{\mathrm{C}_N}_m(\theta), \psi^{\mathrm{C}_N}_n(\theta) \in \{\pm 1\}$ for $\theta \in \{p\frac{2\pi}{N} \,|\, p = 0, \ldots, N-1\}$.

We use the same Fourier basis

$$\left\{ b_{\mu,\gamma}(\phi) = \cos(\mu\phi + \gamma) \;\middle|\; \mu \in \mathbb{N},\ \gamma \in \begin{cases} \{0\} & \text{if } \mu = 0 \\ \{0, \pi/2\} & \text{otherwise} \end{cases} \right\} \tag{B.28}$$

and the same projection operators as used for $\mathrm{SO}(2)$.

Since the *lhs* of the kernel constraint does not depend on the representations considered its projection $\langle b_{\mu',\gamma'}, R_\theta \kappa \rangle$ is the same found for $\mathrm{SO}(2)$:

$$\langle b_{\mu',\gamma'}, R_\theta \kappa \rangle = \frac{1}{2} \sum_\gamma w_{\mu',\gamma} \left(\cos((\gamma' - \gamma) - \mu'\theta) + \delta_{\mu',0} \cos(\gamma' + \gamma) \right)$$

For the *rhs* we find

$$\langle b_{\mu',\gamma'}, \psi_m^{\mathrm{C}_N}(r_\theta)\kappa\ \psi_n^{\mathrm{C}_N}(r_\theta)^{-1}\rangle$$
$$= \frac{1}{2\pi}\int d\phi\ b_{\mu',\gamma'}(\phi)\psi_m^{\mathrm{C}_N}(r_\theta)\kappa(\phi)\psi_n^{\mathrm{C}_N}(r_\theta)^{-1}\,,$$

which by expanding the kernel in the linear combination of the basis and writing the respresentations out yields:

$$= \sum_{\mu,\gamma} w_{\mu,\gamma}\frac{1}{2\pi}\int d\phi\ b_{\mu',\gamma'}(\phi)\cos(m\theta)b_{\mu,\gamma}(\phi)\cos(n\theta)^{-1}$$
$$= \sum_{\mu,\gamma} w_{\mu,\gamma}\frac{1}{2\pi}\int d\phi\ \cos(\mu'\phi+\gamma')\cos(m\theta)\cos(\mu\phi+\gamma)\cos(n\theta)^{-1}$$

Since $\cos(n\theta)\in\{\pm 1\}$ the inverses can be dropped and terms can be collected via trigonometric identities:

$$= \sum_{\mu,\gamma} w_{\mu,\gamma}\frac{1}{2\pi}\int d\phi\ \cos(\mu'\phi+\gamma')\cos(m\theta)\cos(\mu\phi+\gamma)\cos(n\theta)$$
$$= \sum_{\mu,\gamma} w_{\mu,\gamma}\cos(m\theta)\cos(n\theta)\frac{1}{2\pi}\int d\phi\ \cos(\mu'\phi+\gamma')\cos(\mu\phi+\gamma)$$
$$= \sum_{\mu,\gamma} w_{\mu,\gamma}\cos((\pm m\pm n)\theta)\frac{1}{4\pi}\int d\phi\Big(\cos((\mu'-\mu)\phi+\gamma'-\gamma)+\cos((\mu'+\mu)\phi+\gamma'+\gamma)\Big)$$
$$= \sum_{\mu,\gamma} w_{\mu,\gamma}\cos((\pm m\pm n)\theta)\frac{1}{2}\left(\delta_{\mu,\mu'}\cos(\gamma'-\gamma)+\delta_{\mu+\mu',0}\cos(\gamma'+\gamma)\right)$$
$$= \frac{1}{2}\sum_{\mu',\gamma} w_{\mu',\gamma}\cos((\pm m\pm n)\theta)\left(\cos(\gamma'-\gamma)+\delta_{\mu',0}\cos(\gamma'+\gamma)\right)$$

We require the projections to be equal for each $\theta = p\frac{2\pi}{N}$ with $p\in\{0,\dots,N-1\}$:

$$\langle b_{\mu',\gamma'}, R_\theta\kappa\rangle = \left\langle b_{\mu',\gamma'}, \psi_m^{\mathrm{C}_N}(r_\theta)\kappa\ \psi_n^{\mathrm{C}_N}(r_\theta)^{-1}\right\rangle$$
$$\Leftrightarrow \quad \sum_{\gamma} w_{\mu',\gamma}\left(\cos((\gamma'-\gamma)-\mu'\theta)+\delta_{\mu',0}\cos(\gamma'+\gamma)\right) =$$
$$= \sum_{\mu',\gamma} w_{\mu',\gamma}\cos((\pm m\pm n)\theta)\Big(\cos(\gamma'-\gamma)+\delta_{\mu',0}\cos(\gamma'+\gamma)\Big)$$

Again, we consider two cases for μ':

• $\mu' = 0$: The basis in Eq.(B.28) is restricted to the single case $\gamma' = 0$.

$$\sum_\gamma w_{0,\gamma}(\cos(-\gamma)+\cos(\gamma)) = \cos((\pm m \pm n)\theta) \sum_\gamma w_{0,\gamma}\Big(\cos(-\gamma)+\cos(\gamma)\Big)$$

$$\Leftrightarrow \quad w_{0,0}2\cos(0) + w_{0,\frac{\pi}{2}}0 = \cos((\pm m \pm n)\theta)\left(w_{0,0}2\cos(0) + w_{0,\frac{\pi}{2}}0\right)$$

$$\Leftrightarrow \quad w_{0,0} = \cos((\pm m \pm n)\theta) w_{0,0}$$

If $\cos((\pm m \pm n)\theta) \neq 1$, the coefficient $w_{0,0}$ is forced to 0. Conversely:

$$\cos((\pm m \pm n)\theta) = 1$$

$$\Leftrightarrow \quad \exists t \in \mathbb{Z} \text{ s.t.} \quad (\pm m \pm n)\theta = 2t\pi$$

Using $\theta = p\frac{2\pi}{N}$:

$$\Leftrightarrow \quad \exists t \in \mathbb{Z} \text{ s.t.} \quad (\pm m \pm n)p\frac{2\pi}{N} = 2t\pi$$

$$\Leftrightarrow \quad \exists t \in \mathbb{Z} \text{ s.t.} \quad (\pm m \pm n)p = tN$$

• $\mu' > 0$:

$$\sum_\gamma w_{\mu',\gamma}\cos(\gamma' - \gamma - \mu'\theta) = \cos((\pm m \pm n)\theta)\sum_\gamma w_{\mu',\gamma}\cos(\gamma' - \gamma)$$

$$\Leftrightarrow \quad w_{\mu',0}\cos(\gamma' - \mu'\theta) + w_{\mu',\frac{\pi}{2}}\sin(\gamma' - \mu'\theta) = \cos((\pm m \pm n)\theta)\left(w_{\mu',0}\cos(\gamma') + w_{\mu',\frac{\pi}{2}}\sin(\gamma')\right)$$

Since $(\pm m \pm n)\theta \in \{-\pi, 0, \pi\}$ we have $\cos((\pm m \pm n)\theta) = \pm 1$, therefore:

$$\Leftrightarrow \quad w_{\mu',0}\cos(\gamma' - \mu'\theta) + w_{\mu',\frac{\pi}{2}}\sin(\gamma' - \mu'\theta) = w_{\mu',0}\cos(\gamma' + (\pm m \pm n)\theta) + w_{\mu',\frac{\pi}{2}}\sin(\gamma' + (\pm m \pm n)\theta)$$

Using the property in Eq. (B.8):

$$\Leftrightarrow \quad \exists t \in \mathbb{Z} \text{ s.t.} \quad \gamma' - \mu'\theta = \gamma' + (\pm m \pm n)\theta + 2t\pi$$

$$\Leftrightarrow \quad \exists t \in \mathbb{Z} \text{ s.t.} \quad \mu'\theta = (\pm m \pm n)\theta + 2t\pi$$

Using $\theta = p\frac{2\pi}{N}$:

$$\Leftrightarrow \quad \exists t \in \mathbb{Z} \text{ s.t.} \quad \mu' p\frac{2\pi}{N} = (\pm m \pm n)p\frac{2\pi}{N} + 2t\pi$$

$$\Leftrightarrow \quad \exists t \in \mathbb{Z} \text{ s.t.} \quad \mu' p = (\pm m \pm n)p + tN$$

$$\Leftrightarrow \quad \exists t \in \mathbb{Z} \text{ s.t.} \quad (\pm m \pm n + \mu')p = tN$$

In both cases $\mu' = 0$ and $\mu' > 0$ we thus find the constraints

$$\forall p \in \{0, 1, \dots, N-1\} \;\; \exists t \in \mathbb{Z} \text{ s.t.} \quad (\pm m \pm n + \mu')p = tN \,.$$

It is again sufficient to consider the constraint for $p = 1$ which results in solutions with frequencies $\mu' = \pm m \pm n + tN$. As $(\pm m \pm n) \in \left\{0, \pm\frac{N}{2}, \pm N\right\}$, all valid solutions are captured by $\mu' = (m + n \mod N) + tN$, resulting in the basis

$$\mathcal{K}^{\mathrm{C}_N}_{\psi_m \leftarrow \psi_n} = \left\{ b_{\mu,\gamma}(\phi) = \cos(\mu\phi + \gamma) \,\middle|\, \mu = (m+n \mod N) + tN, \gamma \in \left\{0, \frac{\pi}{2}\right\}, \mu \neq 0 \vee \gamma = 0 \right\}_{t \in \mathbb{N}} \tag{B.29}$$

for $n, m \in \left\{0, \frac{N}{2}\right\}$. See the top left cells in Tab. B.4.

1 and 2-dimensional irreps Next consider a 1-dimensional irrep $\rho_{\text{in}} = \psi^{\mathrm{C}_N}_n$ with $n \in \{0, \frac{N}{2}\}$ in the input and a 2-dimensional irrep $\rho_{\text{out}} = \psi^{\mathrm{C}_N}_m$ in the output. We derive the solutions by projecting the kernel constraint on the basis introduced in Eq. (B.15).

For the *lhs* the projection coincides with the result found for $\mathrm{SO}(2)$ as before:

$$\langle b_{\mu',\gamma'}, R_\theta \kappa \rangle = \sum_\gamma w_{\mu',\gamma} \cos((\gamma' - \gamma) - \mu'\theta)$$

An expansion and projection of *rhs* gives:

$$\begin{aligned}
&\langle b_{\mu',\gamma'}, \psi^{\mathrm{C}_N}_m(r_\theta)\kappa(\cdot)\psi^{\mathrm{C}_N}_n(r_\theta)^{-1} \rangle \\
&\quad = \frac{1}{2\pi} \int d\phi \; b_{\mu',\gamma'}(\phi)^T \psi^{\mathrm{C}_N}_m(r_\theta)\kappa(\phi)\psi^{\mathrm{C}_N}_n(r_\theta)^{-1} \\
&\quad = \sum_{\mu,\gamma} w_{\mu,\gamma} \frac{1}{2\pi} \int d\phi \; b_{\mu',\gamma'}(\phi)^T \psi^{\mathrm{C}_N}_m(r_\theta) \begin{bmatrix} \cos(\mu\phi + \gamma) \\ \sin(\mu\phi + \gamma) \end{bmatrix} \psi^{\mathrm{C}_N}_n(r_\theta)^{-1} \\
&\quad = \sum_{\mu,\gamma} w_{\mu,\gamma} \frac{1}{2\pi} \int d\phi \begin{bmatrix} \cos(\mu'\phi + \gamma') & \sin(\mu'\phi + \gamma') \end{bmatrix} \psi^{\mathrm{C}_N}_m(r_\theta) \begin{bmatrix} \cos(\mu\phi + \gamma) \\ \sin(\mu\phi + \gamma) \end{bmatrix} \psi^{\mathrm{C}_N}_n(r_\theta)^{-1} \\
&\quad = \sum_{\mu,\gamma} w_{\mu,\gamma} \left(\frac{1}{2\pi} \int d\phi \; \cos(\mu'\phi + \gamma' - \mu\phi - \gamma - m\theta) \right) \psi^{\mathrm{C}_N}_n(r_\theta)^{-1} \,.
\end{aligned}$$

The integral is non-zero only if the frequency is 0, i.e. iff $\mu' = \mu$:

$$\begin{aligned}
&= \sum_\gamma w_{\mu',\gamma} \cos(\gamma' - \gamma - m\theta) \psi^{\mathrm{C}_N}_n(r_\theta)^{-1} \\
&= \sum_\gamma w_{\mu',\gamma} \cos(\gamma' - \gamma - m\theta) \cos(\pm n\theta)
\end{aligned}$$

Since $\pm n\theta = p\pi$ for some $p \in \mathbb{N}$ one has $\sin(\pm n\theta) = 0$ which allows to add the following zero summand and simplify:

$$\begin{aligned}
&= \sum_{\gamma} w_{\mu',\gamma} \left(\cos(\gamma' - \gamma - m\theta)\cos(\pm n\theta) - \sin(\gamma' - \gamma - m\theta)\sin(\pm n\theta)\right) \\
&= \sum_{\gamma} w_{\mu',\gamma} \cos(\gamma' - \gamma - (m \pm n)\theta)
\end{aligned}$$

Requiring the projections to be equal then yields:

$$\begin{aligned}
& & \langle b_{\mu',\gamma'}, R_\theta \kappa \rangle &= \langle b_{\mu',\gamma'}, \psi_m^{\mathrm{C}_N}(r_\theta)\kappa(\cdot)\psi_n^{\mathrm{C}_N}(r_\theta)^{-1} \rangle & \forall \theta \in \left\{ p\frac{2\pi}{N} \right\} \\
\Leftrightarrow & & \sum_{\gamma} w_{\mu',\gamma} \cos(\gamma' - \gamma - \mu'\theta) &= \sum_{\gamma} w_{\mu',\gamma} \cos(\gamma' - \gamma - (m \pm n)\theta) & \forall \theta \in \left\{ p\frac{2\pi}{N} \right\} \\
\Leftrightarrow & & w_{\mu',0}\cos(\gamma' - \mu'\theta) + w_{\mu',\frac{\pi}{2}}\sin(\gamma' - \mu'\theta) &= w_{\mu',0}\cos(\gamma' - (m\pm n)\theta) + w_{\mu',\frac{\pi}{2}}\sin(\gamma' - (m\pm n)\theta) \\
& & & & \forall \theta \in \left\{ p\frac{2\pi}{N} \right\}
\end{aligned}$$

Using the property in Eq. (B.8), this requires that for each θ there exists a $t \in \mathbb{Z}$ such that:

$$\begin{aligned}
\Leftrightarrow & & \gamma' - \mu'\theta &= \gamma' - (m \pm n)\theta + 2t\pi & \forall \theta \in \left\{ p\frac{2\pi}{N} \right\} \\
\Leftrightarrow & & \mu'\theta &= (m \pm n)\theta + 2t\pi & \forall \theta \in \left\{ p\frac{2\pi}{N} \right\}
\end{aligned}$$

Since $\theta = p\frac{2\pi}{N}$ with $p \in \{0, \ldots, N-1\}$ we find that

$$\begin{aligned}
\Leftrightarrow & & \mu' p\frac{2\pi}{N} &= (m \pm n)p\frac{2\pi}{N} + 2t\pi & \forall p \in \{0, \ldots, N\text{-}1\} \\
\Leftrightarrow & & \mu' p &= (m \pm n)p + tN & \forall p \in \{0, \ldots, N\text{-}1\} \\
\Leftrightarrow & & \mu' &= (m \pm n) + tN & \\
\Leftrightarrow & & \mu' - (m \pm n) &= tN\,, &
\end{aligned}$$

which implies that N needs to divide $\mu' - (m \pm n)$. It follows that the condition holds also for any other p. This gives the basis

$$\mathcal{K}^{\mathrm{C}_N}_{\psi_m \leftarrow \psi_n} = \left\{ b_{\mu,\gamma}(\phi) = \begin{bmatrix} \cos(\mu\phi + \gamma) \\ \sin(\mu\phi + \gamma) \end{bmatrix} \;\middle|\; \mu = (m \pm n) + tN,\ \gamma \in \left\{0, \frac{\pi}{2}\right\} \right\}_{t \in \mathbb{Z}} \tag{B.30}$$

for $m > 0$ and $n \in \left\{0, \frac{N}{2}\right\}$; see the bottom left cells in Tab. B.4.

2 and 1-dimensional irreps:

The basis for 2-dimensional input and 1-dimensional output representations, i.e. $\rho_{\text{in}} = \psi_n^{\mathrm{C}_N}$ and $\rho_{\text{out}} = \psi_m^{\mathrm{C}_N}$ with $n > 0$ and $m \in \{0, \frac{N}{2}\}$, is identical to the previous one up to a transpose:

$$\mathcal{K}^{\mathrm{C}_N}_{\psi_m \leftarrow \psi_n} = \left\{ b_{\mu,\gamma}(\phi) = \begin{bmatrix} \cos(\mu\phi + \gamma) & \sin(\mu\phi + \gamma) \end{bmatrix} \;\middle|\; \mu = (\pm m + n) + tN, \gamma \in \left\{0, \frac{\pi}{2}\right\} \right\}_{t \in \mathbb{Z}} \tag{B.31}$$

for $n > 0$ and $m \in \left\{0, \frac{N}{2}\right\}$. See the top right cells in Tab. B.4.

B.2.6 Dihedral Group D_N

A solution for D_N can easily be derived by repeating the process done for $\mathrm{O}(2)$ in Appendix B.2.4 but starting from the bases derived for C_N in Appendix B.2.5 instead of those for $\mathrm{SO}(2)$.

In contrast to the case of $\mathrm{O}(2)$-equivariant kernels, the choice of reflection axis β is not irrelevant since D_N does not act transitively on axes. More precisely, the action of D_N defines equivalence classes $\beta \cong \beta' \Leftrightarrow \exists\, 0 \leq n < N \;:\; \beta = \beta' + n\frac{2\pi}{N}$ of axes which can be labeled by representatives $\beta \in [0, \frac{2\pi}{N})$. For the same argument considered in Appendix B.2.3 we can without loss of generality consider reflections along the axis $\beta = 0$ in our derivations and retrieve kernels κ', equivariant to reflections along a general axis β, as $\kappa'(r, \phi) = \kappa(r, \phi - \beta)$.

2-dimensional irreps:

For 2-dimensional input and output representations $\rho_{\text{in}} = \psi_{1,n}^{\mathrm{D}_N}$ and $\rho_{\text{out}} = \psi_{1,m}^{\mathrm{D}_N}$, the final basis is

$$\mathcal{K}^{\mathrm{D}_N}_{\psi_{i,m} \leftarrow \psi_{j,n}} = \left\{ b_{\mu,0,s}(\phi) = \psi(\mu\phi)\xi(s) \;\middle|\; \mu = m - sn + tN, s \in \{\pm 1\} \right\}_{t \in \mathbb{Z}} \tag{B.32}$$

where $i = j = 1$ and $m, n > 0$. These solutions are written out explicitly in the bottom right of Tab. B.5.

1-dimensional irreps:

D_N has 1-dimensional representations $\rho_{\text{in}} = \psi^{\mathrm{D}_N}_{j,n}$ and $\rho_{\text{out}} = \psi^{\mathrm{D}_N}_{i,m}$ for $m, n \in \{0, \frac{N}{2}\}$. In these cases we find the bases

$$\mathcal{K}^{\mathrm{D}_N}_{\psi_{i,m} \leftarrow \psi_{j,n}} = \Big\{ b_{\mu,\gamma}(\phi) = \cos(\mu\phi + \gamma) \,\Big|\, \mu = (m + n \mod N) + tN, \quad \gamma = (i + j \mod 2)\frac{\pi}{2},\ \mu \neq 0 \vee \gamma = 0 \Big\}_{t \in \mathbb{N}} \tag{B.33}$$

which are shown in the top left cells of Tab. B.5.

1 and 2-dimensional irreps:

For 1-dimensional input and 2-dimensional output representations, that is, $\rho_{\text{in}} = \psi^{\mathrm{D}_N}_{j,n}$ and $\rho_{\text{out}} = \psi^{\mathrm{D}_N}_{1,m}$ with $i = 1$, $m > 0$ and $n \in \{0, \frac{N}{2}\}$, the kernel basis is given by:

$$\mathcal{K}^{\mathrm{D}_N}_{\psi_{i,m} \leftarrow \psi_{j,n}} = \left\{ b_{\mu,\gamma}(\phi) = \begin{bmatrix} \cos(\mu\phi + \gamma) \\ \sin(\mu\phi + \gamma) \end{bmatrix} \,\middle|\, \mu = (m \pm n) + tN,\ \gamma = j\frac{\pi}{2} \right\}_{t \in \mathbb{Z}} \tag{B.34}$$

See the bottom left of Tab. B.5.

2 and 1-dimensional irreps:

Similarly, for 2-dimensional input and 1-dimensional output representations $\rho_{\text{in}} = \psi^{\mathrm{D}_N}_{1,n}$ and $\rho_{\text{out}} = \psi^{\mathrm{D}_N}_{i,m}$ with $j = 1$, $n > 0$ and $m \in \{0, \frac{N}{2}\}$, we find:

$$\mathcal{K}^{\mathrm{D}_N}_{\psi_{i,m} \leftarrow \psi_{j,n}} = \Big\{ b_{\mu,\gamma}(\phi) = \begin{bmatrix} \cos(\mu\phi + \gamma) & \sin(\mu\phi + \gamma) \end{bmatrix} \,\Big|\, \mu = (\pm m + n) + tN, \gamma = i\frac{\pi}{2} \Big\}_{t \in \mathbb{Z}} \tag{B.35}$$

Tab. B.5 shows these solutions in its top right cells.

B.2.6.1. Kernel constraints at the origin

Our derivations rely on the fact that the kernel constraints restrict only the angular parts of the unconstrained kernel space $L^2(\mathbb{R}^2)^{c_{\text{out}} \times c_{\text{in}}}$ which suggests an independent solution for each radius $r \in \mathbb{R}^+ \cup \{0\}$. Particular attention is required for kernels defined at the origin, i.e. when $r = 0$. The reason for this is that we are using polar coordinates (r, ϕ) which are ambiguous at the origin where the angle is not defined. In order to stay consistent with the solutions for $r > 0$ we still define the kernel at the origin as an element of $L^2(S^1)^{c_{\text{out}} \times c_{\text{in}}}$. However, since the coordinates $(0, \phi)$ map to the same point for all $\phi \in [0, 2\pi)$, we need to demand the kernels to be angularly constant, that is, $\kappa(\phi) = \kappa(0)$. This additional constraint restricts the angular Fourier bases used in the previous derivations to zero frequencies only. Apart from this, the kernel constraints are the same for $r = 0$ and $r > 0$ which implies that the G-steerable kernel bases at $r = 0$ are given by restricting the bases derived in B.2.2, B.2.3, B.2.4, B.2.5 and B.2.6 to the elements indexed by frequencies $\mu = 0$.

B.3 Complex valued representations and Harmonic Networks

Instead of considering real (irreducible) representations we could have derived all results using complex representations, acting on complex feature maps. For the case of $\mathrm{O}(2)$ and D_N this would essentially not affect the derivations since their complex and real irreps are equivalent, that is, they coincide up to a change of basis. Conversely, all complex irreps of $\mathrm{SO}(2)$ and C_N are 1-dimensional which simplifies the derivations in complex space. However, the solution spaces of complex G-steerable kernels need to be translated back to a real valued implementation. This translation has some not immediately obvious pitfalls which can lead to an underparameterized implementation in real space. In particular, Harmonic Networks [48] were derived with a complete solution in complex space; however, their real valued implementation is using a H-steerable kernel space of half the dimensionality as ours. We will in the following explain why this is the case.

In the complex field, the irreps of $\mathrm{SO}(2)$ are given by $\psi_k^{\mathbb{C}}(\theta) = e^{ik\theta} \in \mathbb{C}$ with frequencies $k \in \mathbb{Z}$. Notice that these complex irreps are indexed by positive and negative frequencies while their real counterparts, defined in Sec. 2.7.2, only involve non-negative frequencies. As in [48] we consider complex feature fields $f^{\mathbb{C}} : \mathbb{R}^2 \to \mathbb{C}$ which are transforming according to complex irreps of $\mathrm{SO}(2)$. A complex input field

$f_{\text{in}}^{\mathbb{C}} : \mathbb{R}^2 \to \mathbb{C}$ of type $\psi_n^{\mathbb{C}}$ is mapped to a complex output field $f_{\text{out}}^{\mathbb{C}} : \mathbb{R}^2 \to \mathbb{C}$ of type $\psi_m^{\mathbb{C}}$ via the cross-correlation

$$f_{\text{out}}^{\mathbb{C}} = k^{\mathbb{C}} \star f_{\text{in}}^{\mathbb{C}} \,. \tag{B.36}$$

with a complex filter $k^{\mathbb{C}} : \mathbb{R}^2 \to \mathbb{C}$. The (angular part of the) complete space of equivariant kernels between $f_{\text{in}}^{\mathbb{C}}$ and $f_{\text{out}}^{\mathbb{C}}$ was in [48] proven to be parameterized by

$$k^{\mathbb{C}}(\phi) \;=\; w\, e^{i(m-n)\phi},$$

where $w \in \mathbb{C}$ is a complex weight which scales and phase-shifts the complex exponential. We want to point out that an equivalent parametrization is given in terms of the real and imaginary parts w^{Re} and w^{Im} of the weight w, i.e.

$$\begin{aligned} k^{\mathbb{C}}(\phi) \;&=\; w^{\text{Re}} e^{i(m-n)\phi} + i\, w^{\text{Im}} e^{i(m-n)\phi} \\ &=\; w^{\text{Re}} e^{i(m-n)\phi} + w^{\text{Im}} e^{i((m-n)\phi+\pi/2)} \,. \end{aligned} \tag{B.37}$$

The real valued implementation of Harmonic Networks models the complex feature fields $f^{\mathbb{C}}$ of type $\psi_k^{\mathbb{C}}(\theta)$ by splitting them in two real valued channels $f^{\mathbb{R}} := (f^{\text{Re}}, f^{\text{Im}})^T$ which contain their real and imaginary part. The action of the complex irrep $\psi_k^{\mathbb{C}}(\theta)$ is modeled accordingly by a rotation matrix of the same, potentially negative[3] frequency. A real valued implementation of the cross-correlation (B.36) is built using a real kernel $k : \mathbb{R}^2 \to \mathbb{R}^{2\times 2}$ as specified by

$$\begin{bmatrix} f_{\text{out}}^{Re} \\ f_{\text{out}}^{Im} \end{bmatrix} = \begin{bmatrix} k^{Re} & -k^{Im} \\ k^{Im} & k^{Re} \end{bmatrix} \star \begin{bmatrix} f_{\text{in}}^{Re} \\ f_{\text{in}}^{Im} \end{bmatrix}.$$

The complex steerable kernel (B.37) is then given by

$$\begin{aligned} k(\phi) &= w^{\text{Re}} \begin{bmatrix} \cos\left((m\text{ - }n)\phi\right) & \text{- }\sin\left((m\text{ - }n)\phi\right) \\ \sin\left((m\text{ - }n)\phi\right) & \cos\left((m\text{ - }n)\phi\right) \end{bmatrix} + w^{\text{Im}} \begin{bmatrix} \text{- }\sin\left((m\text{ - }n)\phi\right) & \text{- }\cos\left((m\text{ - }n)\phi\right) \\ \cos\left((m\text{ - }n)\phi\right) & \text{- }\sin\left((m\text{ - }n)\phi\right) \end{bmatrix} \\ &= w^{\text{Re}} \qquad \psi\left((m-n)\phi\right) \qquad + w^{\text{Im}} \qquad \psi\left((m-n)\phi + \frac{\pi}{2}\right) \end{aligned} \tag{B.38}$$

While this implementation models the *complex* Harmonic Networks faithfully in real space, it does not utilize the complete $\mathrm{SO}(2)$-steerable kernel space when the real feature fields are interpreted as fields transforming under the real irreps $\psi_k^{\mathbb{R}}$ as done in our work. More specifically, the kernel space used in (B.38) is only 2-dimensional while our basis (B.12) for the same case is 4-dimensional. The additional solutions with frequency $m + n$ are missing.

[3] This establishes an isomorphism between $\psi_k^{\mathbb{C}}(\theta)$ and $\psi_{|k|}^{\mathbb{R}}(\theta)$ depending on the sign of k.

The lower dimensionality of the complex solution space can be understood by analyzing the relationship between $\mathrm{SO}(2)$'s real and complex irreps. On the complex field, the real irreps become reducible and decomposes into two 1-dimensional complex irreps with opposite frequencies:

$$\psi_k^{\mathbb{R}}(\theta) = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & -i \\ -i & 1 \end{bmatrix} \begin{bmatrix} e^{ik\theta} & 0 \\ 0 & e^{-ik\theta} \end{bmatrix} \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & i \\ i & 1 \end{bmatrix}$$

Indeed, $\mathrm{SO}(2)$ has only half as many real irreps as complex ones since positive and negative frequencies are conjugated to each other, i.e. they are equivalent up to a change of basis: $\psi_k^{\mathbb{R}}(\theta) = \xi(-1)\psi_{-k}^{\mathbb{R}}(\theta)\xi(-1)$. It follows that a real valued implementation of a complex $\psi_k^{\mathbb{C}}$ fields as a 2-dimensional $\psi_k^{\mathbb{R}}$ fields implicitly adds a complex $\psi_{-k}^{\mathbb{C}}$ field. The intertwiners between two real fields of type $\psi_n^{\mathbb{R}}$ and $\psi_m^{\mathbb{R}}$ therefore do not only include the single complex intertwiner between complex fields of type $\psi_n^{\mathbb{C}}$ and $\psi_m^{\mathbb{C}}$, but four complex intertwiners between fields of type $\psi_{\pm n}^{\mathbb{C}}$ and $\psi_{\pm m}^{\mathbb{C}}$. The real parts of these intertwiners correspond to our four dimensional solution space.

In conclusion, [48] indeed found the complete solution on the complex field. However, by implementing the network on the real field, negative frequencies are implicitly added to the feature fields which allows for our larger basis (B.12) of steerable kernels to be used without adding an overhead.

C

Alternative approaches to compute kernel bases and their complexities

The main challenge of building steerable CNNs is to find the space of solutions of the kernel space constraint in Eq. 3.9. Several recent works tackle this problem for the very specific case of features which transform under *irreducible* representations of $\mathrm{SE}(3) \cong (\mathbb{R}^3, +) \rtimes \mathrm{SO}(3)$. The strategy followed in [42, 25, 26, 1] is based on well known analytical solutions and does not generalize to arbitrary representations. In contrast, [44] present a numerical algorithm to solve the kernel space constraint. While this algorithm was only applied to solve the constraints for irreps, it generalizes to arbitrary representations. However, the computational complexity of the algorithm scales unfavorably in comparison to the approach proposed in this work. We will in the following review the kernel space solution algorithm of [44] for general representations and discuss its complexity in comparison to our approach.

The algorithm proposed in [44] is considering the same kernel space constraint

$$k(h\boldsymbol{x}) = \rho_{\text{out}}(h)k(\boldsymbol{x})\rho_{\text{in}}^{-1}(h) \quad \forall h \in H$$

as in this work. By vectorizing the kernel the constraint can be brought in the form

$$\begin{aligned}\operatorname{vec}(k)\,(h\boldsymbol{x}) &= \left(\rho_{\text{out}} \otimes \left(\rho_{\text{in}}^{-1}\right)^T\right)(h)\operatorname{vec}(k)\,(\boldsymbol{x}) \\ &= \left(\rho_{\text{out}} \otimes \rho_{\text{in}}\right)(h)\operatorname{vec}(k)\,(\boldsymbol{x})\,,\end{aligned}$$

where the second step assumes the input representation to be unitary, that is, to satisfy $\rho_{\text{in}}^{-1} = \rho_{\text{in}}^T$. A Clebsch-Gordan decomposition, i.e. a decomposition of the tensor product representation into a direct sum of irreps ψ_j of H, then yields[1]

$$\operatorname{vec}(k)\,(h\boldsymbol{x}) = Q^{-1}\left(\bigoplus\nolimits_{J\in\mathcal{J}}\psi_J\right)(h)\,Q\operatorname{vec}(k)\,(\boldsymbol{x})$$

Through a change of variables $\eta(\boldsymbol{x}) := Q\operatorname{vec}(k)(\boldsymbol{x})$ this simplifies to

$$\eta(h\boldsymbol{x}) = \left(\bigoplus\nolimits_{J\in\mathcal{J}}\psi_J\right)(h)\eta(\boldsymbol{x})$$

[1] For the irreps of $\mathrm{SO}(3)$ it is well known that $\mathcal{J} = \{|j-l|, \ldots, j+l\}$ and $|\mathcal{J}| = 2\min(j,l)+1$.

which, in turn, decomposes into $|\mathcal{J}|$ independent constraints

$$\eta_J(h\boldsymbol{x}) = \psi_J(h)\eta_J(\boldsymbol{x}) \; .$$

Each of these constraints can be solved independently to find a basis for each η_J. The kernel basis is then found by inverting the change of basis and the vectorization, i.e. by computing $k(\boldsymbol{x}) = \mathrm{unvec}(Q^{-1}\eta(\boldsymbol{x}))$.

For the case that $\rho_{\text{in}} = \psi_j$ and $\rho_{\text{out}} = \psi_l$ are Wigner D-matrices, i.e. irreps of $\mathrm{SO}(3)$, the change of basis Q is given by the Clebsch-Gordan coefficients of $\mathrm{SO}(3)$. These well known solutions were used in [42, 25, 26, 1] to build the basis of steerable kernels. Conversely, the authors of [44] solve for the change of basis Q numerically. Given *arbitrary* unitary representations ρ_{in} and ρ_{out} the numerical algorithm solves for the change of basis in

$$\begin{aligned} (\rho_{\text{in}} \otimes \rho_{\text{out}})(h) \;&=\; Q^{-1} \left(\bigoplus_{J\in\mathcal{J}} \psi_J(h) \right) Q \qquad && \forall h \in H \\ \Leftrightarrow \qquad 0 \;&=\; Q(\rho_{\text{in}} \otimes \rho_{\text{out}})(h) \;-\; \left(\bigoplus_{J\in\mathcal{J}} \psi_J(h) \right) Q \qquad && \forall h \in H \; . \end{aligned}$$

This linear constraint on Q, which is a specific instance of the Sylvester equation, can be solved by vectorizing Q, i.e.

$$\left[I \otimes (\rho_{\text{in}} \otimes \rho_{\text{out}})(h) \;-\; \left(\bigoplus\nolimits_{J\in\mathcal{J}} \psi_J \right)(h) \otimes I \right] \mathrm{vec}(Q) \;=\; 0 \qquad \forall h \in H \, ,$$

where I is the identity matrix on $\mathbb{R}^{\dim(\rho_{\text{in}}\otimes\rho_{\text{out}})} = \mathbb{R}^{\dim(\rho_{\text{in}})\dim(\rho_{\text{out}})}$ and $\mathrm{vec}(Q) \in \mathbb{R}^{\dim(\rho_{\text{in}})^2 \dim(\rho_{\text{out}})^2}$. In principle there is one Sylvester equation for each group element $h \in H$, however, it is sufficient to consider the *generators* of H only, since the solutions found for the generators will automatically hold for all group elements. One can therefore stack the matrices $\left[I \otimes (\rho_{\text{in}} \otimes \rho_{\text{out}})(h) - \left(\bigoplus_{J\in\mathcal{J}} \psi_J \right)(h) \otimes I \right]$ for the generators of H into a bigger matrix and solve for Q as the null space of this stacked matrix. The linearly independent solutions Q^J in the null space correspond to the Clebsch-Gordan coefficients for $J \in \mathcal{J}$.

This approach does not rely on the analytical Clebsch-Gordan coefficients, which are only known for specific groups and representations, and therefore works for any choice of representations. However, applying it naively to large representations can be extremely expensive. Specifically, computing the null space to solve the (stacked) Sylvester equation for $\mathcal{H}$ generators of h via a *SVD*, as done in [44], scales as $\mathcal{O}(\dim(\rho_{\text{in}})^6 \dim(\rho_{\text{out}})^6 \mathcal{H})$. This is the case since the matrix which is multiplying $\mathrm{vec}(Q)$ is of shape $\dim(\rho_{\text{in}})^2 \dim(\rho_{\text{out}})^2 \mathcal{H} \times \dim(\rho_{\text{in}})^2 \dim(\rho_{\text{out}})^2$. Moreover, the change of basis matrix Q itself has shape $\dim(\rho_{\text{in}})\dim(\rho_{\text{out}}) \times \dim(\rho_{\text{in}})\dim(\rho_{\text{out}})$

which implies that the change of variables[2] from η to k has complexity $\mathcal{O}(\dim(\rho_{\text{in}})^2 \dim(\rho_{\text{out}})^2)$. In [44], the authors only use irreducible representations which are relatively small such that the bad complexity of the algorithm is negligible.

In comparison, the algorithm proposed in this work is based on an *individual* decomposition of the representations ρ_{in} and ρ_{out} into irreps and leverages the analytically derived kernel space solutions between the irreps of $H \leq \mathrm{O}(2)$. The independent decomposition of the input and output representations leads to a complexity of only $\mathcal{O}((\dim(\rho_{\text{in}})^6 + \dim(\rho_{\text{in}})^6)\mathcal{H})$. We further apply the input and output changes of basis Q_{in} and Q_{out} independently to the irreps kernel solutions κ^{ij} which leads to a complexity of $\mathcal{O}(\dim(\rho_{\text{in}}) \dim(\rho_{\text{out}})^2 + \dim(\rho_{\text{out}}) \dim(\rho_{\text{in}})^2)$. The improved complexity of our implementation makes working with large representations as used in this work, for instance $\dim(\rho_{\text{reg}}^{\mathrm{D}_{20}}) = 40$, possible.

[2] No inversion from Q to Q^{-1} is necessary if the Sylvester equation is solved directly for Q^{-1}.

D

An intuition for quotient representations

Recall the definition of *quotient representation* $\rho_{\text{quot}}^{H/K}$ in Def. 24. A vector $\boldsymbol{v}$ in its representation space can be expressed as $\boldsymbol{v} = \sum_{hK \in H/K} v_{hK} \boldsymbol{e}_{hK}$. Then, its coefficients can be interpreted as a scalar function over H/K as

$$\boldsymbol{v} : H/K \to \mathbb{R},\ hK \mapsto v_{hK}$$

or, equivalently, as a scalar function over H by extension over K:

$$\boldsymbol{v} : H \to \mathbb{R},\ h \mapsto v_{hK} \quad .$$

Therefore, a feature vector transforming according to $\rho_{\text{quot}}^{H/K}$ can be interpreted as a feature map over H, constrained to be constant along its K component. In the special case $K = \{e\}$, the feature vector becomes an unconstrained feature map over H.

In our experiments, we used quotient representations of the form $\rho_{\text{quot}}^{\mathrm{C}_N/\mathrm{C}_M}$ with $\mathrm{C}_M \leq \mathrm{C}_N$ to define the feature types of the models in *rows 11-15* of Tab. 5.1 and in Tab. 5.3. Such field types can encode features which are simultaneously *invariant* with respect to the subgroup C_M but equivariant to C_N. In particular, each of the N/M coefficients in a $\rho_{\text{quot}}^{\mathrm{C}_N/\mathrm{C}_M}$-field detects a different orientation of a C_M-invariant pattern. For instance, if $\mathrm{C}_M = \mathrm{C}_2$ (and therefore N even), a $\rho_{\text{quot}}^{\mathrm{C}_N/\mathrm{C}_2}$-field contains $N/2$ coefficients which are invariant to rotations by π but which permute under rotations by $p\frac{2\pi}{N}$, with $p \in \{0, \dots, \frac{N}{2} - 1\} \cong \mathrm{C}_N \,/\, \mathrm{C}_2$. Such feature field can therefore learn patterns which are symmetric to rotations by π like lines, e.g. for $N = 16$

$$-,\ \backslant,\ \diagup,\ /,\ |,\ \diagdown,\ \diagdown \text{ and } \backslant .$$

As another example, consider $\mathrm{C}_M = \mathrm{C}_4$; in this case, the feature field encodes patterns which are invariant to $\frac{\pi}{2}$ rotations. For instance, for $N = 8$, the features can detect

$$+ \text{ and } \prec,$$

or, for $N = 16$:

$$+,\ \prec,\ \times \text{ and } \prec .$$

Finally, in the special case $\mathrm{C}_M = \mathrm{C}_1 = \{e\}$ is the trivial group, one recovers the *regular representation* $\rho_{\text{reg}}^{\mathrm{C}_N}$ of C_N. A $\rho_{\text{reg}}^{\mathrm{C}_N}$-field can indeed encode arbitrary patterns in N orientations, e.g. for $N = 8$:

$$\rightthreetimes, \perp, \bar{\rightthreetimes}, \perp\!\!\!\perp, \ltimes, \perp\!\!\!\perp, \barwedge \text{ and } \perp\!\!\!\perp .$$

Therefore, a $\rho_{\text{red}}^{\mathrm{C}_N}$-field is always more expressive than any $\rho_{\text{quot}}^{\mathrm{C}_N/\mathrm{C}_M}$-field and a $\rho_{\text{quot}}^{\mathrm{C}_N/\mathrm{C}_M}$-field can always be embedded in a $\rho_{\text{red}}^{\mathrm{C}_N}$-field. However, if a pattern intrinsically has C_M symmetries, a $\rho_{\text{red}}^{\mathrm{C}_N}$-field stores many redundant coefficients. For instance, if $\mathrm{C}_M = \mathrm{C}_4$ and $\mathrm{C}_N = \mathrm{C}_{16}$, a $\rho_{\text{red}}^{\mathrm{C}_N}$-field uses a different coefficient for each of the following patterns:

$$\times, +, \times, +, \times, +, \times \text{ and } +\ .$$

In general, a $\rho_{\text{red}}^{\mathrm{C}_N}$-field uses M times more coefficients to store a C_M-symmetric pattern than a $\rho_{\text{quot}}^{\mathrm{C}_N/\mathrm{C}_M}$-field.

This suggests that the use of quotient representations can reduce the number of channels in a feature field. This can be used to both reduce the memory and computational cost of a model or to introduce more independent fields in the feature type of a layer without increasing its number of channels, potentially increasing its expressiveness. However, notice that a $\rho_{\text{quot}}^{\mathrm{C}_N/\mathrm{C}_M}$-field reduces the number of coefficient stored by enforcing a particular symmetry in the features. Such assumption corresponds to a strong inductive bias in the model which can be useful if it matches the actual data but could harm performance by reducing the expressiveness of the model if non C_M-symmetric patterns are important to solve the task.

In the experiments, we mainly utilized quotient representation invariant to C_2 and C_4, thereby assuming the symmetric patterns like $|$ or $+$, which we believed being the most frequent in MNIST. In order to avoid imposing a too strong restriction, we combined multiple different quotient representations, including regular representation and trivial representations, in the feature types. The model with quotient fields $5\rho_{\text{reg}} \oplus 2\rho_{\text{quot}}^{\mathrm{C}_{16}/\mathrm{C}_2} \oplus 2\rho_{\text{quot}}^{\mathrm{C}_{16}/\mathrm{C}_4} \oplus 4\psi_0$ of C_{16} obtains marginally better accuracy than the purely regular model with the same number of parameters on MNIST rot, see Tab. 5.3. However, in the experiments conducted with the smaller architecture in Tab. 5.1, we did not find improvements by using quotient fields. It is possible that this is a consequence of a non-optimal design of the field type. Indeed, the space of possible quotient representations and their multiplicities is considerably large and may require a more extensive search to find optimal combinations. The use of *neural architecture search* methods to explore these combinations is left as future work. For this reason, we suggest to use regular features as a default design.

We want to draw attention to the fact that the case described above is a very special case as $K = \mathrm{C}_M \triangleleft H = \mathrm{C}_N$ is a *normal subgroup*. In general, if $K \triangleleft H$,

the left and right cosets are identical i.e. $hK = Kh \ \forall h \in H$ and, therefore, $k(hK) = K(kh) = Kh = hK \ \forall k \in K$. Therefore, the action of $\rho_{\text{quot}}^{H/K}$ on a vector $\boldsymbol{v} = \sum_{hK} v_{hK} \boldsymbol{e}_{hK}$ simplifies to

$$\begin{aligned}\rho_{\text{quot}}^{H/K}(k)\ \boldsymbol{v} &= \sum_{hK} v_{hK} \rho_{\text{quot}}^{H/K}(k)\ \boldsymbol{e}_{hK} \\ &= \sum_{hK} v_{hK} \boldsymbol{e}_{khK} \\ &= \sum_{hK} v_{hK} \boldsymbol{e}_{hK} = \boldsymbol{v}\end{aligned}$$

which means $\boldsymbol{v}$ is invariant to K. This does not generally hold if K is not a *normal subgroup*, where the action of $k \in K$ can still induce a permutation of the $|H : K|$ axes. Consider for example quotient representation $\rho_{\text{quot}}^{\mathrm{D}_N/(\{\pm 1\},*)}$, where $(\{\pm 1\}, *)$ is not a normal subgroup of D_N. Then, one has the following action of an element $s \in (\{\pm 1\}, *)$:

$$\rho_{\text{quot}}^{\mathrm{D}_N/(\{\pm 1\},*)}(s)\ e_{r(\{\pm 1\},*)} = \boldsymbol{e}_{sr(\{\pm 1\},*)} = \begin{cases} \boldsymbol{e}_{r(\{\pm 1\},*)} & \text{for } \ s = +1 \\ \boldsymbol{e}_{r^{-1}s(\{\pm 1\},*)} = \boldsymbol{e}_{r^{-1}(\{\pm 1\},*)} & \text{for } \ s = -1 \end{cases}$$

where $r \in \mathrm{C}_N$ is a representative of the coset $r(\{\pm 1\}, *) \in \mathrm{D}_N / (\{\pm 1\}, *)$.

Nevertheless, $\rho_{\text{quot}}^{H/K}$ still enforces a certain symmetry with respect to K in the model. For instance, a ρ_{triv}^{H} to $\rho_{\text{quot}}^{H/K}$ steerable convolution consists of $|H : K|$ kernels $\{\mathcal{R}(hK).\kappa \mid hK \in H/K\}$ obtained transforming a K-invariant kernel κ.

E Additional information on the training setup

layer	output fields
conv block 7×7 (pad 1)	16
conv block 5×5 (pad 2)	24
max pooling 2×2	24
conv block 5×5 (pad 2)	32
conv block 5×5 (pad 2)	32
max pooling 2×2	32
conv block 5×5 (pad 2)	48
conv block 5×5	64
invariant projection	64
global average pooling	64
fully connected	64
fully connected + softmax	10

Tab. E.1.: Basic model architecture from which all models for the MNIST benchmarks in Tab. 5.1 and 5.2 are being derived. Each convolution block includes a convolution layer, batch-normalization and a nonlinearity. The first fully connected layer is followed by batch-normalization and ELU. The width of each layer is expressed as the number of fields of a regular C_{16} model with approximately the same number of parameters.

layer	output fields
conv block 9×9	24
conv block 7×7 (pad 3)	32
max pooling 2×2	32
conv block 7×7 (pad 3)	36
conv block 7×7 (pad 3)	36
max pooling 2×2	36
conv block 7×7 (pad 3)	64
conv block 5×5	96
invariant projection	96
global average pooling	96
fully connected	96
fully connected	96
fully connected + softmax	10

Tab. E.2.: Model architecture for the final MNIST-rot experiments (replicated from [45]). Each fully connected layer follows a dropout layer with $p = 0.3$; the first two fully connected layers are followed by batch normalization and ELU. The width of each layer is expressed in terms of regular feature fields of a C_{16} model.

E.1 Benchmarking on transformed MNIST datasets

Each model reported in Sec. 5.1, 5.2 and 5.3 is derived from the architecture reported in Tab. E.1. The width of each model's layers is thereby scaled such that the total number of parameters is matched and the relative width of layers coincides with that reported in Tab. E.1. Training is performed with a batch size of 64 samples,

using the *Adam* optimizer [24]. The learning rate is initialized to 10^{-3} and decayed exponentially by a factor of 0.8 per epoch, starting after a burn in phase of 10 epochs. We train each model for 30 epochs and test the model which performed best on the validation set. A weight decay of 10^{-7} is being used for all convolution layers and the first fully connected layer. In all experiments, we build steerable bases with Gaussian radial profiles of width $\sigma = 0.6$ for all except the outermost ring where we use $\sigma = 0.4$. We apply a strong band-limiting policy which permits frequencies up to $0, 2, 2$ for radii $0, 1, 2$ in a 5×5 kernel and up to $0, 2, 3, 2$ for radii $0, 1, 2, 3$ in a 7×7 kernel. The strong cutoff in the rings of maximal radius is motivated by our empirical observation that these rings introduce a relatively high equivariance error for higher frequencies. This is the case since the outermost ring ranges out of the sampled kernel support. During training, data augmentation with continuous rotations and reflections is performed (if these are present in the dataset) to not disadvantage non-equivariant models. In the models using group restriction, the restriction operation is applied after the convolution layers but before batch normalization and non-linearities.

E.2 Competitive runs on MNIST rot

In Tab. 5.3 we report the performances of some of our best models. Our experiments are based on the best performing, C_{16}-equivariant model of [45] which defined the state of the art on rotated MNIST at the time of writing. We replicate their model architecture, summarized in Tab. E.2, though our models have a different frequency band-limit and width σ for the Gaussian radial profiles as discussed in the previous subsection. As before, the table reports the width of each layer in terms of number of fields in the C_{16} regular model.

As commonly done, we train our final models on the 10000 + 2000 training and validation samples. Training is performed for 40 epochs with an initial learning rate 0.015, which is being decayed by a factor of 0.8, starting after 15 epochs. As before, we use the *Adam* optimizer with a batch size of 64, this time using L1 and L2 regularization with a weight of 10^{-7}. The fully connected layers are additionally regularized using dropout with a probability of $p = 0.3$. We are again using train time augmentation.

E.3 CIFAR experiments

The equivariant models used in the experiments on CIFAR-10 and CIFAR-100 are adapted from the original WideResNet models by replacing conventional with G-

steerable convolutions and scaling the number of feature fields such that the total number of parameters is preserved. For blocks which are equivariant under D_8 or C_8 we use 5×5 kernels instead of 3×3 kernels to allow for higher frequencies. All models use regular feature fields in all but the final convolution layer, which maps to a scalar field (conv2triv) to produce invariant predictions. We use a frequency cut-off of 3 times the ring's radius, e.g. $0, 3, 6$ for rings of radii $0, 1, 2$. These higher bandlimits in comparison to the MNIST experiments are motivated by the fact that the corresponding bases introduce small discretization errors, which is no problem for the classification of natural images. In the contrary, this leads to the models having a strong bias towards being equivariant, but might allow them to break equivariance if necessary. The widths of the bases' rings is chosen to be $\sigma = 0.45$ in all rings.

The training process is the same as used for WideResNets: we train for 200 epochs with a batch size of 128. We optimize the model with SGD, using an initial learning rate of 0.1, momentum 0.9 and a weight decay of $5 \cdot 10^{-4}$. The learning rate is decayed by a factor of 0.2 every 60 epochs. We perform a standard data augmentation with random crops, horizontal flips and normalization. No CutOut is done during the normal experiments but it is used in the AutoAugment policies.

E.4 STL-10 experiments

The models for our STL-10 experiments are adapted from [16]. However, according to an issue[1] in the authors' GitHub repository, the publication states some model parameters and the training setup wrongly. Our adaptations are therefore based on the setting reported on GitHub. Specifically, we use patches of 60×60 pixels for cutout and the stride of the first convolution layer in the first block is 2 instead of 1. Moreover, we normalize input features using CIFAR-10 statistics. Though these statistics are very close to the statistics of STL-10, they might, as the authors of [16] suggest, cause non-negligible changes in performance because of the small training set size of STL-10.

As before, regular feature fields are used throughout the whole model except for the last convolution layer which maps to trivial fields. In the small model, which does not preserve the number of parameters but the number of channels, we still scale up the number of output channels of the very first convolution layer (before the first residual block). As the first convolution layer originally has 16 output channels and our model is initially equivariant to D_8 (whose regular representation spans 16 channels), the first convolution layer would only be able to learn 1 single

[1] https://github.com/uoguelph-mlrg/Cutout/issues/2

independent filter (repeated 16 times, rotated and reflected). Hence, we increase the number of output channels of the first convolution layer by the square root of the group size ($\sqrt{16} = 4$) leading to $4 \cdot 16 = 64$ channels, i.e. $64/16 = 4$ regular fields. We use a ring width of $\sigma = 0.6$ for the kernel basis except for the outermost ring where we use $\sigma = 0.4$ and use a frequency cut-off factor of 3 for the rings' radii, i.e. cutoffs of $0, 3, 6, \ldots$.

We are again exactly replicating the training process as reported in the publication [16]. Only the labeled subset of the training set is used, that is, the 100000 unlabeled training images are discarded. Training is performed for 1000 epochs with a batch size of 128, using SGD with Nesterov momentum of 0.9 and weight decay of $5 \cdot 10^{-4}$. The learning rate is initialized to 0.1 and decayed by a factor of 5 at 300, 400, 600 and 800 epochs. During training, we perform data augmentation by zero-padding with 12 pixels and randomly cropping patches of 96×96 pixels, mirroring them horizontally and applying CutOut.

In the data ablation study, reported in Figure 5.4, we use the same models and training procedure as in the main experiment on the full STL-10 dataset. For every single run, we generate new datasets by mixing the original training, validation and test set and sample reduced datasets such that all classes are balanced. The results are averaged over 4 runs on each of the considered training set sizes of 250, 500, 1000, 2000 or 4000. The validation and test sets contain 1000 and 8000 images, which are re-sampled in each run as well.

F

Additional information on the irrep models

SO(2) models We experiment with some variants (rows 37-44) of the Harmonic Network model in row 30 of Tab. 5.1, varying in either the non-linearity or the invariant map applied. All of these models are therefore to be analyzed relative to this baseline. First, we try to use *squashing* nonlinearities [21] (row 37) instead of norm-ReLUs on each non-trivial irrep. This variant performs consistently worse than the original model. In the baseline variant, we generate invariant features via a convolution to scalar fields in the last layer (*conv2triv*). This, however, reduces the utilization of high frequency irrep fields in the penultimate layer. The reason for this is that the kernel space for mappings from high frequency- to scalar fields consists of kernels of a high angular frequency, which will be cut off by our bandlimiting. To overcome this problem, we propose to instead compute the norms of all non-trivial fields to produce invariant features. This enables us to use all irreps in the output of the last convolution layer. However, we find that combining invariant norm mappings with norm-ReLUs does not improve on the baseline model, see row 38. In row 39 we consider a variant which applies norm-ReLUs on the direct sum of multiple non-trivial irrep fields, each with multiplicity 1, together (*shared norm-ReLU*), while the scalar fields are still being acted on by ELUs. This is legitimate since the direct sum of unitary representations is itself unitary. After the last convolution layer, the invariant projection preserves the trivial fields but computes the norm of each composed field. This model significantly outperforms all previous variants on all datasets. The model in row 40 additionally merges the scalar fields to such combined fields instead of treating them independently. This architecture performs significantly worse than the previous variants.

We further explore four different variations which are applying *gated nonlinearities* (rows 41-44). These models distinguish from each other by 1) their mapping to invariant features and 2) whether the gate is being applied to each non-trivial field independently or being shared between multiple non-trivial fields. We find that the second choice, i.e. sharing gates, does not significantly affect the performances (row 41 vs. 42 and 43 vs. 44). However, mapping to invariant features by taking the norm of all non-trivial fields performs consistently better than applying *conv2triv*. Overall, gated nonlinearities perform significantly better than any other choice of nonlinearity on the tested $\mathrm{SO}(2)$ irrep models.

O(2) models Here we will give more details on the $\mathrm{O}(2)$-specific operations which we introduce to improve the performance of the $\mathrm{O}(2)$-equivariant models, reported in rows 45-57 of Tab. 5.1.

- $\mathrm{O}(2)$-*conv2triv:* As invariant map of the $\mathrm{O}(2)$ irrep models in rows 46-49 and 54 we are designing a last convolution layer which is mapping to an output representation $\rho_{\text{out}} = \psi_{0,0}^{\mathrm{O}(2)} \oplus \psi_{1,0}^{\mathrm{O}(2)}$, that is, to scalar fields $f_{0,0}$ and sign-flip fields $f_{1,0}$ in equal proportions. Since the latter are not invariant under reflections, we are in addition taking their absolute value. The resulting, invariant output features are then multiple fields $f_{0,0} \oplus |f_{1,0}|$. The motivation for not convolving to trivial representations of $\mathrm{O}(2)$ directly via *conv2triv* is that the steerable kernel space for mappings between irreps of $\mathrm{O}(2)$ does not allow for mapping between $\psi_{0,0}^{\mathrm{O}(2)}$ and $\psi_{1,0}^{\mathrm{O}(2)}$ (see Tab. B.2), which would lead to dead neurons.

The models in rows 50-53, 56 and 57 operate on $\mathrm{Ind}_{\mathrm{SO}(2)}^{\mathrm{O}(2)} \psi_k^{\mathrm{SO}(2)}$-fields whose representations are induced from the irreps of $\mathrm{SO}(2)$. Per definition, this representation acts on feature vectors f in $\mathbb{R}^{\dim(\psi_k^{\mathrm{SO}(2)})} \otimes \mathbb{R}^{|\,\mathrm{O}(2):\mathrm{SO}(2)|}$, which we treat in the following as functions $f : \mathrm{O}(2)/\,\mathrm{SO}(2) \to \mathbb{R}^{\dim(\psi_k^{\mathrm{SO}(2)})}$. We further identify the coset $s\,\mathrm{SO}(2)$ in the quotient space $\mathrm{O}(2)/\,\mathrm{SO}(2)$ by its representative $\mathcal{R}(s\,\mathrm{SO}(2)) := s \in (\{\pm 1\}, *)$ in the reflection group. Eq. 2.5 defines the action of the induced representation on a feature vector by

$$
\begin{aligned}
\Big([\,\mathrm{Ind}_{\mathrm{SO}(2)}^{\mathrm{O}(2)} \psi_k^{\mathrm{SO}(2)}](\tilde{r}\tilde{s})\, f\Big)(s\,\mathrm{SO}(2)) \;&:= \psi_k^{\mathrm{SO}(2)}\big(\,\mathrm{h}\,(\tilde{r}\tilde{s}\mathcal{R}((\tilde{r}\tilde{s})^{-1} s\,\mathrm{SO}(2)))\big)\, f\big((\tilde{r}\tilde{s})^{-1} s\,\mathrm{SO}(2)\big) \\
&= \psi_k^{\mathrm{SO}(2)}\big(\,\mathrm{h}(\tilde{r}s)\big)\, f\big(\tilde{s}s\,\mathrm{SO}(2)\big) \\
&= \begin{cases} \psi_k^{\mathrm{SO}(2)}(\tilde{r})\, f\big(\tilde{s}s\,\mathrm{SO}(2)\big) & \text{for } \; s = +1 \\ \psi_k^{\mathrm{SO}(2)}(\tilde{r}^{-1})\, f\big(\tilde{s}s\,\mathrm{SO}(2)\big) & \text{for } \; s = -1\,, \end{cases}
\end{aligned}
$$

where we used Eq. 2.3 to compute

$$
\mathrm{h}(\tilde{r}s) \;:=\; \mathcal{R}\big(\tilde{r}s\,\mathrm{SO}(2)\big)^{-1}\tilde{r}s \;=\; s^{-1}\tilde{r}s \;=\; \begin{cases} \tilde{r} & \text{for } \; s = +1 \\ \tilde{r}^{-1} & \text{for } \; s = -1\,. \end{cases}
$$

Intuitively, this action describes a permutation of the subfields (indexed by s) via the reflection $\tilde{s}$ and a rotation of the subfields by $\tilde{r}$ and $\tilde{r}^{-1}$, respectively. Specifically, for $k = 0$, the induced representation is for all $\tilde{r}$ instantiated by

$$[\operatorname{Ind}_{\mathrm{SO}(2)}^{\mathrm{O}(2)} \psi_0^{\mathrm{SO}(2)}](\tilde{r}\tilde{s}) \;=\; \begin{cases} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} & \text{for } \; \tilde{s} = +1 \\ \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} & \text{for } \; \tilde{s} = -1 \;\; , \end{cases} \tag{F.1}$$

that is, it coincides with the regular representation of the reflection group. Similarly, for $k > 0$, it is for all $\tilde{r}$ given by the 4×4 matrices

$$[\operatorname{Ind}_{\mathrm{SO}(2)}^{\mathrm{O}(2)} \psi_{k>0}^{\mathrm{SO}(2)}](\tilde{r}\tilde{s}) \;=\; \begin{cases} \left[\begin{array}{c|c} \psi_{k>0}^{\mathrm{SO}(2)}(\tilde{r}) & 0 \\ \hline 0 & \psi_{k>0}^{\mathrm{SO}(2)}(-\tilde{r}) \end{array}\right] & \text{for } \; \tilde{s} = +1 \\ \left[\begin{array}{c|c} 0 & \psi_{k>0}^{\mathrm{SO}(2)}(\tilde{r}) \\ \hline \psi_{k>0}^{\mathrm{SO}(2)}(-\tilde{r}) & 0 \end{array}\right] & \text{for } \; \tilde{s} = -1 \;\; . \end{cases}$$

We adapt the conv2triv and norm invariant maps, as well as the norm-ReLU and the gated nonlinearities to operate on $\operatorname{Ind}_{\mathrm{SO}(2)}^{\mathrm{O}(2)}$-fields as follows:

- Ind-*conv2triv:* Instead of applying *O(2)-conv2triv* to compute invariant features, we apply convolutions to $\operatorname{Ind}_{\mathrm{SO}(2)}^{\mathrm{O}(2)} \psi_0^{\mathrm{SO}(2)}$-fields which are invariant under rotations but behave like regular $(\{\pm 1\}, *)$-fields under reflections. These fields are subsequently mapped to a scalar field via G-pooling, i.e. by taking the maximal response over the two subfields.

- Ind-*norm:* An alternative invariant map is defined by computing the norms of the subfields of each final $\operatorname{Ind}_{\mathrm{SO}(2)}^{\mathrm{O}(2)} \psi_k^{\mathrm{SO}(2)}$-field and applying G-pooling over the result.

- Ind *norm-ReLU:* It would be possible to apply a norm-ReLU to a $\operatorname{Ind}_{\mathrm{SO}(2)}^{\mathrm{O}(2)} \psi_k^{\mathrm{SO}(2)}$-field for $k > 0$ as a whole, that is, to compute the norm of both subfields together. Instead, we apply two individual norm-ReLUs to the subfields. Since the fields permute under reflections, we need to choose the bias parameter of the two norm-ReLUs to be equal.

- Ind *gate:* Similarly, we could apply a single gate to each $\mathrm{Ind}_{\mathrm{SO}(2)}^{\mathrm{O}(2)} \psi_k^{\mathrm{SO}(2)}$-field. However, we apply an individual gate to each subfield. In this case it is necessary that the gates permute together with the $\mathrm{Ind}_{\mathrm{SO}(2)}^{\mathrm{O}(2)} \psi_k^{\mathrm{SO}(2)}$-fields to ensure equivariance. This is achieved by computing the gates from $\mathrm{Ind}_{\mathrm{SO}(2)}^{\mathrm{O}(2)} \psi_0^{\mathrm{SO}(2)}$-fields, which contain two permuting scalar fields.

Empirically we find that $\mathrm{Ind}_{\mathrm{SO}(2)}^{\mathrm{O}(2)}$ models perform much better than pure irrep models, despite both of them being equivalent up to a change of basis. In particular, the induced representations decompose for some change of basis matrices Q_0 and $Q_{>0}$ into:

$$\begin{aligned}
\mathrm{Ind}_{\mathrm{SO}(2)}^{\mathrm{O}(2)} \psi_0^{\mathrm{SO}(2)} &= Q_0 \left[\psi_{0,0}^{\mathrm{O}(2)} \oplus \psi_{1,0}^{\mathrm{O}(2)}\right] Q_0^{-1} \\
\mathrm{Ind}_{\mathrm{SO}(2)}^{\mathrm{O}(2)} \psi_{k>0}^{\mathrm{SO}(2)} &= Q_{>0} \left[\psi_{1,k>0}^{\mathrm{O}(2)} \oplus \psi_{1,k>0}^{\mathrm{O}(2)}\right] Q_{>0}^{-1}
\end{aligned}$$

The difference between both bases is that the induced representations disentangle the action of reflections into a permutation, while the direct sum of irreps is modeling reflections in each of its sub-vectorfields independently as an inversion of the vector direction and rotation orientation. Note the analogy to the better performance of regular representations in comparison to a direct sum of the respective irreps.

G

Efficient Decomposition of Induced Representations

For a finite group G, if the G-representation to be decomposed is the induced representation from an *irreducible* representation ψ of a subgroup $H \leq G$, i.e. $\mathrm{Ind}_H^G \psi$, a more efficient algorithm can be used. [36] uses induced representations to derive an efficient algorithm to compute the inverse Fourier transform for complex functions over finite groups and includes a method to build the change of basis matrix of an induced representation. Though [36] assumes complex-valued representations, a very similar algorithm can be derived on the real field. Here, we will first repeat the derivations in [36] to obtain an abstract description of the change of basis matrix and, then, we will discuss how this result can be adapted to real representations and implemented in practice.

Notations In this section, we assume a generic finite group G and a subgroup $H < G$. We will denote an irrep of G with ρ and an irrep of H with ψ. Without loss of generality, we assume all irreps are unitary representations, i.e. $\rho(g)^{-1} = \rho(g)^\dagger$, where † is the conjugate transpose. Given an irrep ψ of H, our goal is to decompose the G-representation $\mathrm{Ind}_H^G \psi$ into a direct sum of irreducible representations of G, i.e.:

$$\mathrm{Ind}_H^G \psi = D^{-1} \left(\bigoplus_{i \in I} \rho_i \right) D \quad ,$$

where $\{\rho_i\}_i$ are (possibly equal) irreducible representations of G.

The irreducible representation $\psi : H \to \mathrm{GL}(W)$ acts on a vector space W of size d_ψ. Therefore, we can choose a basis

$$\{e_1, e_2, \dots e_k, \dots, e_{d_\psi}\}$$

of W and express the matrix representations of H on this basis. We call $\mathrm{Ind}_H^G W$, or shortly $\mathrm{Ind}\, W$, the vector space the representation $\mathrm{Ind}_H^G \psi$ acts on. Assuming a set of representative elements

$$R = \{r_1, \dots, r_l, \dots, r_L\}$$

of the $L = [G : H]$ cosets in G/H, the matrix representations of G can be naturally expressed with respect to the following basis of $\mathrm{Ind}_H^G W$:

$$\{r_1 \otimes e_1, r_1 \otimes e_2, \dots, r_1 \otimes e_{d_\psi}, \dots, r_L \otimes e_1, \dots, r_K \otimes e_{d_\psi}\} \ .$$

In such basis, the matrix representations of G are in the form presented in Sec. 2.4, i.e. they are block matrices with $L \times L$ blocks, each of size $d_\psi \times d_\psi$, with each row and column contains only one non-zero block.

As before, we can use the *orthogonality of the characters of the irreducible representations* in Thm. 5 to find the multiplicities of the irreps of G in $\mathrm{Ind}_H^G \psi$. We can assume we already know $\mathrm{Ind}_H^G \psi \cong \bigoplus_i \rho_i$. Denoting the direct sum representation as $P = \bigoplus_i \rho_i$, we only need to find the matrix D such that $P = D \,\mathrm{Ind}_H^G \psi D^{-1}$. We call V the vector space P acts on. Note that $V \cong \mathrm{Ind}\, W$, but V is associated with a basis such that the matrix representations of $\mathrm{Ind}\, \psi$ are the block diagonal ($P = D \,\mathrm{Ind}_H^G \psi D^{-1}$). Therefore, the problem is that of finding a linear map which commutes with the two representations (an *intertwiner*), i.e. a matrix D such that $D\left(\mathrm{Ind}_H^G \psi\right) = PD$.

Given two representations ρ_1 and ρ_2 of a group G on the vector spaces V_1 and V_2, the set of all linear maps between V_1 and V_2 which commute with respect to their respective G-actions (Def. 28) is

$$\mathrm{Hom}_G(V_1, V_2) = \{A \mid \rho_2(g)A = A\rho_1(g) \quad \forall g \in G\} \ .$$

An element $\phi \in \mathrm{Hom}_G(V_1, V_2)$ is a linear map $\phi : V_1 \to V_2$. Once the bases of V_1 and V_2 are chosen, this set corresponds to the set of all matrices which commute with the two representations. Then, $D \in \mathrm{Hom}_G(\mathrm{Ind}\, W, V)$; therefore, we will now study this set.

Because $P = \bigoplus_i \rho_i$, its vector space (and its basis) can also be decomposed in a direct sum $V = \bigoplus_{i \in I} V_i$. It follows that:

$$\mathrm{Hom}_G(\mathrm{Ind}\, W, V) = \mathrm{Hom}_G\left(\mathrm{Ind}\, W, \bigoplus_{i \in I} V_i\right) \cong \bigoplus_{i \in I} \mathrm{Hom}_G(\mathrm{Ind}\, W, V_i) \tag{G.1}$$

which means that the matrices of the linear maps between V and $\mathrm{Ind}\, W$ can be decomposed in $|I|$ blocks $\{D_i \mid i \in I\}$, where the i-th block is an intertwiner between $\mathrm{Ind}\, \psi$ and ρ_i. The block D_i contains the coefficient of a map $\Phi_i \in \mathrm{Hom}_G(\mathrm{Ind}\, W, V_i)$. This enables us to focus only on one of these blocks, i.e. on one of the irreps ρ_i.

We can now use the following theorem [15]:

Theorem 10: Frobenius Reciprocity

Let G be a finite group, $H \leq G$ a subgroup and $\{r_1, \dots, r_K\}$ (with $r_1 = e$) a complete set of coset representatives for G/H. Let ψ be an H-representation in W and ρ be a G-representation in V.
Then, there is a *canonical isomorphism* such that:

$$\mathrm{Hom}_H\left(W, \mathrm{Res}\, V\right) \cong \mathrm{Hom}_G\left(\mathrm{Ind}\, W, V\right)$$

where $\mathrm{Res}\, V$ is the vector space V considered as an H-space, i.e. as the vector space of the restricted representation $\mathrm{Res}^G_H \psi$.
This canonical isomorphism maps an element $\Phi \in \mathrm{Hom}_G\left(\mathrm{Ind}\, W, V\right)$ to an element $\phi \in \mathrm{Hom}_H\left(W, \mathrm{Res}\, V\right)$ defined as:

$$\phi : W \to \mathrm{Res}\, V, \quad \phi(w) := \Phi(e \otimes w) \quad .$$

Similarly, an element $\phi \in \mathrm{Hom}_H\left(W, \mathrm{Res}\, V\right)$ is mapped to an element $\Phi \in \mathrm{Hom}_G\left(\mathrm{Ind}\, W, V\right)$ as:

$$\Phi : \mathrm{Ind}\, W \to V, \quad \Phi(r_l \otimes w) := \rho(r_l)\, \phi(w) \quad .$$

This allows us to study the set $\mathrm{Hom}_H\left(W, \mathrm{Res}\, V_i\right)$ instead of $\mathrm{Hom}_G\left(\mathrm{Ind}\, W, V_i\right)$. Splitting the block D_i horizontally into L $d_\psi \times d_\psi$ blocks $D_i(r_1 = e), \dots, D_i(r_L)$, $\mathrm{Hom}_H\left(W, \mathrm{Res}\, V_i\right)$ is the set of all possible matrices $D_i(e)$ in the first d_ψ columns of D_i, i.e. the block of D which maps the coset associated with the identity $r_1 = e$ to the irrep ρ_i. Once the maps in $\mathrm{Hom}_H\left(W, \mathrm{Res}\, V_i\right)$ are found, we can use the *canonical isomorphism* in Thm. 10 to compute all the maps in $\mathrm{Hom}_G\left(\mathrm{Ind}\, W, V_i\right)$, i.e. we can build the other blocks $\{D_i(r)\}_{r \in R}$ from $D_i(e)$.

The result obtained in the last paragraph allows us to reduce the study to $\mathrm{Res}^G_H \rho_i$, the restriction to H of the irreducible representation ρ_i of G. We assume that we already know its decomposition $\mathrm{Res}^G_H \rho_i = A_i \bigoplus_{j \in J_i} \psi^i_j A_i^{-1}$ and that $A_i^{-1} = A^\dagger$ is unitary. Without loss of generality, we can momentarily ignore the change of basis A_i and consider the space $\tilde{V}_i$ associated with the representation $A_i^{-1} \rho_i A_i$. Note that the vector spaces $\tilde{V}_i \cong V_i$ are isomorphic and only differ for the change of basis A_i. Similarly, $\mathrm{Hom}_H\left(W, \mathrm{Res}\, V_i\right) \cong \mathrm{Hom}_H\left(W, \mathrm{Res}\, \tilde{V}_i\right)$, with isomorphism

$$\mathrm{Hom}_H\left(W, \mathrm{Res}\, \tilde{V}_i\right) \to \mathrm{Hom}_H\left(W, \mathrm{Res}\, V_i\right), \quad \phi \mapsto A_i \phi$$

and

$$\mathrm{Hom}_H\left(W, \mathrm{Res}\, V_i\right) \to \mathrm{Hom}_H\left(W, \mathrm{Res}\, \tilde{V}_i\right), \quad \phi \mapsto A_i^{-1} \phi \quad .$$

If $D_i(e) \in \mathrm{Hom}_H\left(W, \mathrm{Res}\, V_i\right)$, using the isomorphism above, we define the block $\tilde{D}_i(e) = A_i^{-1} D_i(e) \in \mathrm{Hom}_H\left(W, \mathrm{Res}\, \tilde{V}_i\right)$. We will re-introduce the change of basis

A_i later in the final change of basis D. Then, the vector space $\operatorname{Res}\tilde{V}_i$ (and its basis) decomposes into a direct sum: $\operatorname{Res}\tilde{V}_i = \bigoplus_{j=1}^{J_i} W_j^i$. Similarly, the set of homomorphisms can be decomposed as:

$$\operatorname{Hom}_H\left(W, \operatorname{Res}\tilde{V}_i\right) = \operatorname{Hom}_H\left(W, \bigoplus_{j=1}^{J_i} W_j^i\right) = \bigoplus_{j=1}^{J_i} \operatorname{Hom}_H\left(W, W_j^i\right) \quad .$$

This means that the block $\tilde{D}_i(e)$ is itself composed by sub-blocks $\{D_j^i(e)\}_j$. We can now use Schur's Representation Lemma, which we introduced in Thm. 3. The theorem implies that $\operatorname{Hom}_H\left(W, W_j^i\right)$ contains only the null matrix (containing only zeros) if ψ_j^i differs from ψ. We can therefore consider only those W_j^i associated to the irrep $\psi_j^i = \psi$. We will use the set $\bar{J}_i = \{j \in J_i | \psi_j^i = \psi\}$ to index them.

Because of the block-diagonal structure of $\operatorname{Res}\tilde{V}_i$, for $j \in \bar{J}_i$ there is a subset of the basis of $\operatorname{Res}\tilde{V}_i$ which forms a basis for W_j^i. If $\{f_1^i, \dots, f_{d_{\rho_i}}^i\}$ is the basis of $\operatorname{Res}\tilde{V}_i$, we denote as $\{f_{j,1}^i, \dots, f_{j,k}^i, \dots, f_{j,d_\psi}^i\}$ its subset which is a basis for W_j^i. Given any isomorphism $\phi_j^i \in \operatorname{Hom}_H\left(W, W_j^i\right)$ and the basis $\{e_1, e_2, \dots e_k, \dots, e_{d_\psi}\}$ of W, the vectors $\{\phi_j^i(e_1), \phi_j^i(e_2), \dots \phi_j^i(e_k), \dots, \phi_j^i(e_{d_\psi})\}$ are expressed in terms of the basis $\{f_{j,k}^i\}_{k=1}^{d_\psi}$ and their coefficients form the columns of the block $D_j^i(e)$. Note that these are d_ψ-dimensional vectors padded to d_{ρ_i}-dimensional vectors.

Then, the block $\tilde{D}_i(e)$ is the stack of the blocks $\{D_j^i(e)\}_j$ and is the matrix representation of $\tilde{\phi}_i \in \operatorname{Hom}_H\left(W, \operatorname{Res}\tilde{V}_i\right)$ which sends

$$e_k \to \tilde{\phi}_i(e_k) := \sum_{j \in \bar{J}_i} \phi_j^i(e_k) \quad .$$

We can now reintroduce the change of basis A_i so the block $D_i(e) = A_i\tilde{D}_i(e)$ is the matrix representation of $\phi_i \in \operatorname{Hom}_H(W, \operatorname{Res} V_i)$ which sends

$$e_k \to \phi_i(e_k) := A_i\tilde{\phi}_i(e_k) = A_i \sum_{j \in \bar{J}_i} \phi_j^i(e_k) = \sum_{j \in \bar{J}_i} A_i\phi_j^i(e_k) \quad .$$

Then, using the *canonical isomorphism* described in Thm 10, the corresponding element $\Phi_i \in \operatorname{Hom}_G(\operatorname{Ind} W, V_i)$ sends

$$r_l \otimes e_k \to \rho_i(r_l)\phi_i(e_k) = \rho_i(r_l) \sum_{j \in \bar{J}_i} A_i\phi_j^i(e_k) \quad .$$

This means that the other blocks $\{D_i(r_l)\}_l$ in D_i can be built as

$$D_i(r_l) = \rho_i(r_l)D_i(e) \quad .$$

Finally, the whole matrix D can be built by stacking the blocks $\{D_i\}_{i \in I}$.

G.1 Orthogonality

The previous construction completely defines the space $\mathrm{Hom}_G(\mathrm{Ind}\, W, V)$ of matrices which commute with the induced representation, i.e. the set $\left\{D \mid D\left(\mathrm{Ind}_H^G \psi\right) = PD\right\}$. However, this does not guarantee the invertibility of D. Furthermore, we want to find a matrix which is not only invertible but also unitary (or orthonormal for real representations). This can be done by choosing the blocks $\{D_j^i(e)\}_{i,j}$ appropriately.

Recall that a matrix D is invertible if and only if its rows are linearly independent. However, if an irrep ρ of G has multiplicity 2 in $\mathrm{Ind}\, \psi$ (i.e. $\exists i_1, i_2 \in I$ s.t. $\rho_{i_1} = \rho_{i_2} = \rho$) and we choose the same isomorphisms in both cases (i.e. $D_{i_1}(e) = D_{i_2}(e)$), the matrix D will contain two identical rows (i.e. $D_{i_1}(r_l) = D_{i_2}(r_l)\ \forall r_l$). We will now show that it is always possible to choose two different isomorphisms ϕ_{i_1} and ϕ_{i_2} such that the rows of the resulting matrix will be orthogonal too. First of all, we can make use of the following variant of Thm. 10:

Theorem 11: Frobenius Reciprocity (Character Theory)

Let G be a finite group, $H \leq G$ a subgroup. Let ψ be an H-representation in W and ρ be a G-representation in V. Then:

$$< \mathrm{Ind}_H^G \psi, \rho >_G = < \psi, \mathrm{Res}_H^G \rho >_H$$

where $< \cdot, \cdot >$. denotes the inner product of the characters of the two representations.

Complex representations For simplicity, we first assume representations over the *complex* field $\mathbb{C}$. Without loss of generality, we assume all representations to be *unitary*, i.e. $\rho(g^{-1}) = \rho(g)^{-1} = \rho(g)^\dagger$, where † is the conjugate transpose. In this case, Thm. 11 together with Thm. 5 imply that the multiplicity of a G-irrep ρ in the induced representation $\mathrm{Ind}_H^G \psi$ is equal to the multiplicity of the H-irrep ψ in the restricted representation $\mathrm{Res}_H^G \rho$. Moreover, in the complex field, a stronger version of *Schur's Representation Lemma* than the one in Thm. 3 holds. The complex version of the theorem was stated in Thm. 4. In our case, it implies that $\mathrm{Hom}_H\left(W, W_j^i\right)$ is a one-dimensional space and

$$\forall j \in \bar{J}_i,\ \exists \lambda_j^i \in \mathbb{C} \text{ s.t. } \quad \phi_j^i = \lambda_j^i I \quad .$$

Denoting the multiplicity of the irrep ρ in $\mathrm{Ind}\, \psi$ (or, equivalently, the multiplicity of ψ in $\mathrm{Res}\, \rho$) as $M = |\bar{J}_i|$, $\mathrm{Hom}_H(W, \mathrm{Res}\, V_i) = \bigoplus_{j \in \bar{J}_i} \mathrm{Hom}_H\left(W, W_j^i\right)$ is an M-dimensional space and it occurs M times in $\mathrm{Hom}_H(W, \mathrm{Res}\, V)$. Therefore, we can

always choose different isomorphisms in $\mathrm{Hom}_H(W, \mathrm{Res}\, V)$ by setting all $\lambda^i_j = 0$ except for the m-th index in $\bar{J}_i$ in the m-th occurrence of $\mathrm{Hom}_H(W, \mathrm{Res}\, V_i)$.

We now show the orthogonality of the rows of D. Consider two different blocks $D_i(e), D_j(e) \in \{D_i(e)\}_{i\in I}$ associated to the i-th and the j-th irreps ρ_i and ρ_j in the irreps decomposition of $\mathrm{Ind}\,\psi = D \bigoplus_{i\in I} \rho_i D^{-1}$. ρ_i and ρ_j are not necessarily equivalent. Let's now evaluated the inner product between all pairs of rows of D associated with the i-th and j-th irreps i.e. between the rows of D_i and those of D_j. Recall that a block $D_i : \mathrm{Ind}\, W \to V_i$ is composed by L blocks $\{D_i(r) = \rho_i(r)D_i(e) : W \to V_i\}_{r\in R}$ and that $D_i(e)\psi(h) = \rho_i(h)D_i(e) \;\; \forall h \in H$ by definition. Then, we can write:

$$\begin{aligned} O =& D_i D_j^\dagger \\ =& \sum_{r\in R} \rho_i(r) D_i(e) D_j(e)^\dagger \rho_j(r)^\dagger \\ =& \frac{1}{|H|} \sum_{h\in H} \sum_{r\in R} \rho_i(r) D_i(e) D_j(e)^\dagger \rho_j(r)^\dagger \\ =& \frac{1}{|H|} \sum_{h\in H} \sum_{r\in R} \rho_i(r) D_i(e)\psi(h)\psi(h)^\dagger D_j(e)^\dagger \rho_j(r)^\dagger \\ =& \frac{1}{|H|} \sum_{h\in H} \sum_{r\in R} \rho_i(r)\rho_i(h) D_i(e) D_j(e)^\dagger \rho_j(h)^\dagger \rho_j(r)^\dagger \\ =& \frac{1}{|H|} \sum_{g\in G} \rho_i(g) D_i(e) D_j(e)^\dagger \rho_j(g)^\dagger \end{aligned} \tag{G.2}$$

Note that O commutes with $\rho_i(g)$ and $\rho_j(g)$ for any $g \in G$ by construction, i.e.

$$O\rho_j(g) = \rho_i(g)O \quad \forall g \in G$$

and, therefore, it is an intertwiner between ρ_j and ρ_i. Using Thm. 4, this implies that O is the null matrix if $\rho_i \ncong \rho_j$. Conversely, if $\rho_i = \rho_j =: \rho$, then it guarantees that

$$\exists \lambda \in \mathbb{C} \text{ s.t.} \quad O = \lambda I$$

Now, because the trace operator is linear and invariant under conjugation

$$\begin{aligned} \mathrm{Tr}(O) =& \mathrm{Tr}\left(\frac{1}{|H|} \sum_{g\in G} \rho(g) D_i(e) D_j(e)^\dagger \rho(g)^\dagger \right) \\ =& \frac{1}{|H|} \sum_{g\in G} \mathrm{Tr}\left(\rho(g) D_i(e) D_j(e)^\dagger \rho(g)^\dagger \right) \end{aligned}$$

recalling $D_i(e) = A_i\tilde{D}_i(e)$ and defining $A := A_i = A_j$:

$$
\begin{aligned}
&= \frac{1}{|H|}\sum_{g\in G} \mathrm{Tr}\left(\rho(g)A\tilde{D}_i(e)\tilde{D}_j(e)^\dagger A^\dagger \rho(g)^\dagger\right) \\
&= \frac{1}{|H|}\sum_{g\in G} \mathrm{Tr}(\tilde{D}_i(e)\tilde{D}_j(e)^\dagger)
\end{aligned}
$$

The matrix $\tilde{D}_i\tilde{D}_j^\dagger$ is a $d_\rho \times d_\rho$ zero matrix with a block structure (according to the decomposition of ρ into irreps of H) containing the $d_\psi \times d_\psi$ identity matrix in the (i,j)-th block. If $i \neq j$, the identity is not appearing on the main diagonal and, therefore, $\mathrm{Tr}(\tilde{D}_i\tilde{D}_j^\dagger) = 0$. It follows that $\mathrm{Tr}(O) = 0$ and, because $O = \lambda I$, $\lambda = 0$, i.e. O is the null matrix. Hence, the inner product between any pair of rows belonging to different blocks is always zero and, so, different blocks are orthogonal. Conversely, if $i = j$, $\tilde{D}_i\tilde{D}_j^\dagger$ contains the $d_\psi \times d_\psi$ identity matrix in the main diagonal and, therefore, $\mathrm{Tr}(O) > 0$, i.e. $O = \lambda I$ with $\lambda > 0$. This implies that the rows within the same block D_i are orthogonal to each other Finally, by properly scaling the blocks $D_i(e)$, one can achieve orthonormality.

Real field On the *real* field $\mathbb{R}$ the inner product of the characters can not be directly interpreted as the multiplicity of the irreps and the space of isomorphisms might be larger. Nevertheless, it turns out that these two concepts are very related. Using Thm. 6, it is possible to show that $< \mathrm{Ind}_H^G \psi, \rho >_G = \delta_\rho M$, where M is the multiplicity of ρ in $\mathrm{Ind}_H^G \psi$ while $\delta_\rho =< \rho, \rho >_G \in \mathbb{N}^+$. Similarly, the following relation holds: $< \psi, \mathrm{Res}\, \rho >_H = \delta_\psi N$, where N is the multiplicity of ψ in $\mathrm{Res}_H^G \rho$ while $\delta_\psi =< \psi, \psi >_H \in \mathbb{N}^+$. Using Thm. 11:

$$\delta_\rho M = \delta_\psi N$$

By decomposing a real irrep ψ in complex irreps, one can also show that $< \psi, \psi >_H$ is equal to the dimensionality of the space $\mathrm{Hom}_H(\psi, \psi)$. It follows that the dimensionality of $\mathrm{Hom}_H\left(W, W_j^i\right)$ is equal to δ_ψ.

Unfortunately, deriving an orthogonal change of basis is less straightforward than in the complex case. We first need to introduce some properties of real representations. It can be shown[1] that any complex irreducible representation σ of a compact group G occurs in one and only one real irreducible representation ρ, when the last is interpreted as a complex representation. Moreover, every real irrep ρ of G can be classified in one of three categories (types):

- *real type*: if $\exists\sigma$ s.t. $\rho \cong \sigma \cong \overline{\sigma}$, i.e. if it is isomorphic to a complex irrep σ

[1] http://www-math.mit.edu/~poonen/715/real_representations.pdf

- *complex type*: if $\exists \sigma \ncong \overline{\sigma}$ s.t. $\rho \cong \sigma \oplus \overline{\sigma}$, i.e. if it decomposes as the direct sum of two (non-equivalent) complex irreps σ and $\overline{\sigma}$, which are the conjugated of each other

- *quaternionic type*: if $\exists \sigma \cong \overline{\sigma}$ s.t. $\rho \cong \sigma \oplus \overline{\sigma} \cong \sigma \oplus \sigma$, i.e. if decomposes as the direct sum of two copies of complex irreps σ, which is equivalent to its conjugation $\overline{\sigma}$

where $\bar{\cdot}$ is the complex conjugate. For brevity, because no irrep of the groups $H \leq \mathrm{O}(2)$ have quaternionic type, we will ignore the last case. Without loss of generality, we assume that real-type real irreps are expressed on a basis such that their matrix representations have real coefficients and that $\sigma = \rho$ (and, therefore, $\sigma = \overline{\sigma}$). Similarly, for any complex-type real irrep ρ, we can choose a basis such that

$$\rho(g) = \begin{bmatrix} Re\left(\sigma(g)\right) & -Im\left(\sigma(g)\right) \\ Im\left(\sigma(g)\right) & Re\left(\sigma(g)\right) \end{bmatrix} = C_{d_\sigma} \begin{bmatrix} \sigma(g) & 0 \\ 0 & \overline{\sigma}(g) \end{bmatrix} C_{d_\sigma}^{\dagger} \tag{G.3}$$

with

$$C_d = \frac{1}{\sqrt{2}} \begin{bmatrix} iI_d & -iI_d \\ I_d & I_d \end{bmatrix} \tag{G.4}$$

where i is the imaginary unit, I_d is the identity matrix of size d, † is the conjugate transpose while $Re\left(\cdot\right)$ and $Im\left(\cdot\right)$ are the real and imaginary parts of their arguments. We will often write C instead of C_d when the dimension d is clear from the other matrices in the expression. As before, we study the space $\mathrm{Hom}_H\left(\psi, \mathrm{Res}\, \rho_i\right)$ where the block $D_i(e)$ lives. As in the complex case, if $\rho_i \neq \rho_j$, the space $\mathrm{Hom}\left(\rho_i, \rho_j\right)$ contains only the null matrix and, therefore, using similar steps to those in Eq. (G.2), one can show that the rows of D_i and D_j are orthogonal. Here, however, when $\rho_i = \rho_j =: \rho$, we need to consider four different cases, depending on whether ρ and ψ are of real or complex types.

As earlier, we assume the restricted ρ_i decomposes as $\mathrm{Res}\, \rho = A_i \left(\bigoplus_{j \in J_i} \psi_j^i\right) A_i^{-1}$ and, so, $\mathrm{Hom}_H\left(\psi, \mathrm{Res}\, \rho_i\right) \cong \bigoplus_{j \in \bar{J}_i} \mathrm{Hom}_H\left(\psi, \psi_j^i\right)$, where $\bar{J}_i \subseteq J_i$ contains the indexes of the irreps $\psi_j^i \cong \psi$.

ρ_i real, ψ real This case is identical to the case with complex irreps, so no further analysis is necessary.

ρ_i real, ψ complex Assume $\psi = C\left(\eta \oplus \overline{\eta}\right) C^\dagger$, where η is a complex irrep of H. Note that here $\delta_\rho = 1$ and $\delta_\psi = 2$, so $M = 2N$. Since ψ is a complex-type irrep, $\mathrm{Hom}_H\left(\psi, \psi\right) \cong \mathrm{Hom}_H\left(\eta, \eta\right) \oplus \mathrm{Hom}_H\left(\overline{\eta}, \overline{\eta}\right)$ in the complex field. Considering only

homomorphisms whose matrix coefficients are real and expressing ψ as in Eq. (G.3), $\mathrm{Hom}_H(\psi,\psi)$ contains matrices in the form $\lambda I_{d_\psi} + \omega S$ ($\lambda, \omega \in \mathbb{R}$), where

$$S = \begin{bmatrix} 0 & I_{d_\eta} \\ -I_{d_\eta} & 0 \end{bmatrix}$$

and I_d is the $d \times d$ identity matrix. Note that I_{d_ψ} and S together form an orthogonal basis for $\mathrm{Hom}_H(\psi,\psi)$. We denote this basis with $\mathcal{X} = \{I_{d_\psi}, S\}$ and we will use X to refer to one of its elements. In addition, if $X_1, X_2 \in \mathcal{X}$, one can also show that $\mathrm{Tr}(X_1 X_2^T) = \delta_{X_1,X_2}$. Then, for each of the M occurrences of ρ in $\mathrm{Ind}\,\psi$, we can choose one of the N occurrences of ψ in $\mathrm{Res}\,\rho$ and one of the two orthogonal basis elements of $\mathrm{Hom}_H(\psi,\psi)$: $\tilde{D}_{i_m}(e)$ is a zero $d_\rho \times d_\psi$ matrix containing $X_{m \mod 2} \in \mathcal{X}$ in the rows corresponding to the $\lfloor m/2 \rfloor$-th occurrence of ψ in the irreps decomposition of $\mathrm{Res}\,\rho$. We now need to show that this construction leads to orthonormal rows in D. As before, we compare the rows associated with the m-th and the n-th occurrence of ρ_i, i.e. the blocks D_{i_m} and D_{i_n}. Again, we assume the decomposition $\mathrm{Res}\,\rho_i = A_i \left(\bigoplus_j \psi_j\right) A_i^{-1}$, with $A_i^{-1} = A_i^T$ real-valued orthogonal matrix. Then, $D_i(r) = \rho_i(r) A_i \tilde{D}_i(e)$. With similar steps as in the complex field:

$$\begin{aligned} O &= D_{i_m} D_{i_n}^T \\ &= \sum_{r \in R} \rho(r) A_i \tilde{D}_{i_m}(e) \tilde{D}_{i_n}(e)^T A_i^T \rho(r)^T \\ &= \frac{1}{|H|} \sum_g \rho(g) A_i \tilde{D}_{i_m}(e) \tilde{D}_{i_n}(e)^T A_i^T \rho(g)^T \end{aligned}$$

Because O commutes with $\rho(g)$, $O \in \mathrm{Hom}_G(\rho,\rho)$ and, therefore, $\exists \lambda \in \mathbb{R}$ s.t. $O = \lambda I$. If $\lfloor m/2 \rfloor \neq \lfloor n/2 \rfloor$, $\mathrm{Tr}(O) = \mathrm{Tr}(\tilde{D}_{i_m}(e)\tilde{D}_{i_n}(e)^T) = 0$ and, so, O is the null matrix. Otherwise, $\mathrm{Tr}(\tilde{D}_{i_m}(e)\tilde{D}_{i_n}(e)^T) = \mathrm{Tr}(X_{m \mod 2} X_{n \mod 2}^T) = \delta_{n \mod 2, n \mod 2} = \delta_{n,m}$. It follows that in general $O = \delta_{n,m}\lambda I$. Thus, D_{i_m} and D_{i_n} always contain orthogonal rows.

ρ_i complex, ψ real Assume $\rho_i = C(\sigma_i \oplus \overline{\sigma_i}) C^\dagger$ and $\mathrm{Res}\,\sigma_i = B_i \left(\bigoplus_{j \in J_i} \eta_j\right) B_i^\dagger$, where $\sigma_i \ncong \overline{\sigma_i}$ are complex irreps of G while η_j are complex irreps of H. Note that it necessarily holds that $\mathrm{Res}\,\overline{\sigma_i} = \overline{\mathrm{Res}\,\sigma_i} = \overline{B}_i \left(\bigoplus_{j \in J_i} \overline{\eta_j}\right) B_i^T$. Because ψ is of real-type, there exists a complex η such that $\psi = \eta = \overline{\eta}$. Here, $\delta_\rho = 2$ and $\delta_\psi = 1$, so $2M = N$. Indeed, using the decomposition of $\mathrm{Res}\,\rho$ in $\sigma_i \oplus \overline{\sigma_i}$ and their decompositions in terms of complex H-irreps, it follows that ψ occurs M times in $\overline{\sigma_i}$ and M times in σ_i. Here, instead of decomposing $\mathrm{Res}\,\rho_i$ in terms of real irreps of H as $\mathrm{Res}\,\rho_i = A_i \left(\bigoplus_j \psi_j\right) A_i^T$, we consider its decomposition

$$\mathrm{Res}\,\rho_i = C \begin{bmatrix} B_i & \\ & \overline{B}_i \end{bmatrix} \begin{bmatrix} \bigoplus_{j \in J_i} \eta_j & \\ & \bigoplus_{j \in J_i} \overline{\eta_j} \end{bmatrix} \begin{bmatrix} B_i & \\ & \overline{B}_i \end{bmatrix}^\dagger C^\dagger$$

in terms of the complex irreps of H. Analogously, we parametrize a block $D_i(e)$ as $D_i(e) = C(B \oplus \overline{B})\tilde{D}_i(e)$ instead of $D_i(e) = A_i\tilde{D}_i(e)$. For the m-th occurrence of ρ_i in $\operatorname{Ind}\psi$, we set $\tilde{D}_{i_m}(e) = \begin{bmatrix} X_m \\ X_m \end{bmatrix} \in \mathbb{R}^{2d_{\sigma_i} \times d_\psi}$, where X_m is a $d_{\sigma_i} \times d_\psi$ zero matrix containing the $d_\psi \times d_\psi$ identity in the rows corresponding to the m-th occurrence of $\psi = \eta$ in the complex irreps decomposition of $\operatorname{Res}\sigma_i$. It follows that $\operatorname{Tr}(X_m X_n^T) = \delta_{m,n}$. We can also verify that $D_{i_m}(e)$ is always real valued; indeed

$$C(B_i \oplus \overline{B}_i) = \frac{1}{\sqrt{2}}\begin{bmatrix} iI & -iI \\ I & I \end{bmatrix}\begin{bmatrix} B_i & \\ & \overline{B}_i \end{bmatrix} = \underbrace{\begin{bmatrix} Re\,(B_i) & -Im\,(B_i) \\ Im\,(B_i) & Re\,(B_i) \end{bmatrix}}_{B_i^{\mathbb{R}}} C$$

and, therefore:

$$D_{i_m}(e) = C(B_i \oplus \overline{B}_i)\tilde{D}_{i_m}(e) = B_i^{\mathbb{R}} C \begin{bmatrix} X_m \\ X_m \end{bmatrix} = B_i^{\mathbb{R}} \frac{2}{\sqrt{2}} \begin{bmatrix} 0 \\ X_m \end{bmatrix}$$

We can now verify orthogonality as before.

$$\begin{aligned}
O = D_{i_m} D_{i_n}^T = D_{i_m} D_{i_n}^\dagger &= \frac{1}{|H|}\sum_g \rho(g) D_{i_m}(e) D_{i_n}(e)^\dagger \rho(g)^\dagger \\
&= \frac{1}{|H|}\sum_g C \begin{bmatrix} \sigma_i(g) & \\ & \overline{\sigma_i}(g) \end{bmatrix} C^\dagger D_{i_m}(e) D_{i_n}(e)^T C^\dagger \begin{bmatrix} \sigma_i(g) & \\ & \overline{\sigma_i}(g) \end{bmatrix}^\dagger C^\dagger \\
&= \frac{1}{|H|}\sum_g C \begin{bmatrix} \sigma_i(g) & \\ & \overline{\sigma_i}(g) \end{bmatrix} \begin{bmatrix} B_i & \\ & \overline{B}_i \end{bmatrix} \begin{bmatrix} X_m \\ X_m \end{bmatrix} \begin{bmatrix} X_n^T & X_n^T \end{bmatrix} \begin{bmatrix} B_i & \\ & \overline{B}_i \end{bmatrix}^\dagger \begin{bmatrix} \sigma_i(g) & \\ & \overline{\sigma_i}(g) \end{bmatrix}^\dagger C^\dagger \\
&= \frac{1}{|H|} C \sum_g \begin{bmatrix} \sigma_i(g) B_i X_m X_n^T B_i^\dagger \sigma_i(g)^\dagger & \sigma_i(g) B_i X_m X_n^T \overline{B}_i^\dagger \overline{\sigma}_i(g)^\dagger \\ \overline{\sigma}_i(g)\overline{B}_i X_m X_n^T B_i^\dagger \sigma_i(g)^\dagger & \overline{\sigma}_i(g)\overline{B}_i X_m X_n^T \overline{B}_i^\dagger \overline{\sigma}_i(g)^\dagger \end{bmatrix} C^\dagger \\
&= C \begin{bmatrix} \lambda I & 0 \\ 0 & \overline{\lambda} I \end{bmatrix} C^\dagger
\end{aligned}$$

Now, note that $\operatorname{Tr}(\lambda I) = \operatorname{Tr}(\overline{\lambda} I) \propto \operatorname{Tr}(X_m X_n^T) = \delta_{m,n}$. It follows that $\lambda = \overline{\lambda} \in \mathbb{R}$ and $O = \lambda\delta_{m,n} I$. Thus, the rows in D_{i_m} and D_{i_n} are orthogonal to each other.

ρ_i complex, ψ complex Again, assume $\rho_i = C_{d_\sigma}\left(\sigma_i \oplus \overline{\sigma_i}\right) C_{d_\sigma}^\dagger$, $\operatorname{Res}\sigma_i = B_i\left(\bigoplus_{j\in J_i}\eta_j\right)B_i^\dagger$ and $\psi = C_{d_\eta}(\eta \oplus \overline{\eta})C_{d_\eta}^\dagger$, with $\sigma_i \ncong \overline{\sigma_i}$ complex irreps of G and $\eta \ncong \overline{\eta}$ complex irreps of H. Here, $\delta_\rho = \delta_\psi = 2$, so $2M = 2N$. Hence, for the m-th occurrence of ρ_i, we parametrize the block D_{i_m} as

$$D_{i_m} = C_{d_\sigma}(B \oplus \overline{B})\tilde{D}_{i_m} C_{d_\eta}^\dagger$$

with

$$\tilde{D}_{i_m} = \begin{bmatrix} X_m & 0 \\ 0 & X_m \end{bmatrix} \in \mathbb{R}^{d_\rho \times d_\psi}$$

where X_m is a $d_\sigma \times d_\eta$ zero matrix containing the identity in the rows correspoding to the m-th occurrence of η in σ. Then

$$\begin{aligned}
D_{i_m} &= C_{d_\sigma}(B \oplus \overline{B})\tilde{D}_{i_m}C_{d_\eta}^\dagger \\
&= \underbrace{\begin{bmatrix} Re\,(B) & -Im\,(B) \\ Im\,(B) & Re\,(B) \end{bmatrix}}_{B^{\mathbb{R}}} C_{d_\sigma}\tilde{D}_{i_m}C_{d_\eta}^\dagger \\
&= B^{\mathbb{R}}\frac{1}{2}\begin{bmatrix} iI_{d_\sigma} & -iI_{d_\sigma} \\ I_{d_\sigma} & I_{d_\sigma} \end{bmatrix}\begin{bmatrix} X_m & 0 \\ 0 & X_m \end{bmatrix}\begin{bmatrix} iI_{d_\eta} & -iI_{d_\eta} \\ I_{d_\eta} & I_{d_\eta} \end{bmatrix} \\
&= B^{\mathbb{R}}\begin{bmatrix} X_m & 0 \\ 0 & X_m \end{bmatrix} = B^{\mathbb{R}}\tilde{D}_{i_m}
\end{aligned}$$

which is a real-valued matrix. Note that $\mathrm{Tr}(X_m X_n^T) = d_\eta \delta_{m,n}$. Then:

$$\begin{aligned}
O &= D_{i_m}D_{i_n}^T = D_{i_m}D_{i_n}^\dagger \\
&= \frac{1}{|H|}\sum_g \rho(g)D_{i_m}(e)D_{i_n}(e)^\dagger\rho(g)^\dagger \\
&= \frac{1}{|H|}\sum_g \rho(g)C_{d_\sigma}(B \oplus \overline{B})\tilde{D}_{i_m}C_{d_\eta}^\dagger C_{d_\eta}\tilde{D}_{i_n}^T(B \oplus \overline{B})^\dagger C_{d_\sigma}^\dagger\rho(g)^\dagger \\
&= \frac{1}{|H|}C_{d_\sigma}\sum_g \begin{bmatrix} \sigma(g) & \\ & \overline{\sigma}(g) \end{bmatrix}(B \oplus \overline{B})\begin{bmatrix} X_m X_n^T & \\ & X_m X_n^T \end{bmatrix}(B \oplus \overline{B})^\dagger \begin{bmatrix} \sigma(g) & \\ & \overline{\sigma}(g) \end{bmatrix}^\dagger C_{d_\sigma}^\dagger \\
&= \frac{1}{|H|}C_{d_\sigma}\sum_g \begin{bmatrix} \sigma(g)BX_m X_n^T B^\dagger\sigma(g)^\dagger & \\ & \overline{\sigma}(g)\overline{B}X_m X_n^T\overline{B}^\dagger\overline{\sigma}(g)^\dagger \end{bmatrix} C_{d_\sigma}^\dagger \\
&= C_{d_\sigma}\begin{bmatrix} \lambda I_{d_\sigma} & \\ & \overline{\lambda}I_{d_\sigma} \end{bmatrix} C_{d_\sigma}^\dagger
\end{aligned}$$

Using the properties of the trace, the following equalities need to hold $\mathrm{Tr}(\lambda I_{d_\sigma}) = \mathrm{Tr}(\overline{\lambda}I_{d_\sigma}) = \mathrm{Tr}(X_m X_n^T)$. Hence, $\lambda = \overline{\lambda} \in \mathbb{R}$ and $\lambda \propto \delta_{m,n}$. It follows that $\exists\lambda \in \mathbb{R}$ s.t. $O = \delta_{m,n}\lambda I$ and that the rows of D_{i_m} and D_{i_n} are othogonal.

Finally, by scaling each block $\{D_i\}_i$, one can normalize all the columns of the matrix D, generating an orthonormal change of basis.

To validate our implementation, we perform an extensive numerical test, inducing from and to a large number of subgroups of D_{32} and from $\mathrm{SO}(2)$ to $\mathrm{O}(2)$.

Nomenclature

Groups

C_N	Cyclic Group of order N
D_N	Dihedral Group of order $2N$
$\mathrm{E}(2)$	Euclidean Group
$(\{\pm 1\}, *)$	Reflection Group
$\mathrm{GL}(2)$	General Linear Group of 2×2 invertible matrices
$\mathrm{GL}(n)$	General Linear Group of $n \times n$ invertible matrices. Often also written as $\mathrm{GL}(\mathbb{R}^n)$
$\mathrm{O}(2)$	Orthogonal Group
$\mathrm{SO}(2)$	Special Orthogonal Group
$\mathrm{U}(1)$	Unitary Group of complex numbers with norm 1

Notation

$\oplus$	Direct Sum
$\otimes$	Direct Product
Cov	Covariance
$\mathbb{E}$	Expectation
$\Pr$	Probability function
$\boldsymbol{v}$	a vector
$\boldsymbol{x}$	a point in $\mathbb{R}^2$

Operators

$\mathrm{Ind}_H^G \rho$	Induced Representation of ρ from H to G, with $H \leq G$
$\mathrm{Res}_H^G \rho$	Restricted Representation of ρ from G to H, with $H \leq G$
$\mathrm{vec}(M)$	Vectorization of a matrix M.

Other Symbols

$\psi(\theta)$	2×2 rotation matrix associated with the angle θ	$\begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}$
$\xi(s)$	2×2 matrix associated with the element $s \in (\{\pm 1\}, *)$	$\begin{bmatrix} 1 & 0 \\ 0 & s \end{bmatrix}$

List of Definitions

List of Theorems

List of Figures

List of Tables