$\mathrm{SU}(1, 1)$ Equivariant Neural Networks and Application to Robust Toeplitz Hermitian Positive Definite Matrix Classification

Pierre-Yves Lagrave[1(✉)], Yann Cabanes[2,3], and Frédéric Barbaresco[2]

[1] Thales Research and Technology, Palaiseau, France
pierre-yves.lagrave@thalesgroup.com
[2] Thales Land and Air Systems, Limours, France
frederic.barbaresco@thalesgroup.com
[3] Institut de Mathématiques de Bordeaux, Bordeaux, France

Abstract. In this paper, we propose a practical approach for building $\mathrm{SU}(1, 1)$ equivariant neural networks to process data with support within the Poincaré disk $\mathbb{D}$. By leveraging on the transitive action of $\mathrm{SU}(1, 1)$ on $\mathbb{D}$, we define an equivariant convolution operator on $\mathbb{D}$ and introduce a Helgason-Fourier analysis approach for its computation, that we compare with a conditional Monte-Carlo method. Finally, we illustrate the performance of such neural networks from both accuracy and robustness standpoints through the example of Toeplitz Hermitian Positive Definite (THPD) matrix classification in the context of radar clutter identification from the corresponding Doppler signals.

Keywords: Equivariant neural networks · Hyperbolic embedding · Homogeneous spaces · Burg algorithm · Radar clutter.

1 Introduction

Group-Convolutional Neural Networks (G-CNN) are becoming more and more popular thanks to their conceptual soundness and to their ability to reach state-of-the-art accuracies for a wide range of applications [3,9,12]. In this paper, we propose a scalable approach for building $\mathrm{SU}(1, 1)$ equivariant neural networks and provide numerical results for their application to Toeplitz Hermitian Positive Definite (THPD) matrix classification.

Dealing with THPD matrices is indeed of particular interest for Doppler signal processing tasks, as the input data can be represented by auto-correlation matrices assuming some stationarity assumptions [5], which can themselves be embedded into a product space of Poincaré disks after proper rescaling [6] by leveraging on the Trench-Verblunsky theorem. We have instanciated a $\mathrm{SU}(1, 1)$ equivariant neural network in this context and have provided accuracy and robustness results by using simulated data according to some realistic radar clutter modeling [6].

F. Nielsen and F. Barbaresco (Eds.): GSI 2021, LNCS 12829, pp. 577–584, 2021.
https://doi.org/10.1007/978-3-030-80209-7_62

1.1 Related Work and Contribution

G-CNN have initially been introduced in the seminal work [8] as a generalization of usual CNN [15] by introducing group-based convolution kernels to achieve equivariance with respect to the action of finite groups and have further been generalized to more generic actions [9,14]. In particular, [10] proposed a sound theory for the case of the transitive action of a compact group on its homogeneous space by leveraging on group representation theory, a topic also covered in [7].

In [2], G-CNN on Lie groups have been introduced for homogeneous spaces by decomposing the kernel function on a B-splines basis. Anchoring in similar ideas, [12] introduced a very generic approach providing equivariance to the action of any Lie group with a surjective exponential map and which is applicable to any input data representable by a function defined on a smooth manifold and valued in a vectorial space.

Practical applications of G-CNN mainly relate to groups of transforms in Euclidean spaces such as SE(2), SO(3), $\mathbb{R}_+^*$, or the semi-product of those with the translation group. In the present paper, we work with the group of isometric transforms $\mathrm{SU}(1,1)$ acting on the hyperbolic space represented as the Poincaré disk $\mathbb{D}$.

Although existing works provide tools for specifying $\mathrm{SU}(1,1)$ based G-CNN on which we will anchor, working with non-compact groups is quite challenging from a numerical stanpoint. For instance, the Monte-Carlo approach of [12] faces scalability issue with the number of layers of the considered network due to the large sampling requirement for $\mathrm{SU}(1,1)$. As a remediation for our specific $\mathrm{SU}(1,1)$ instanciation, we propose specifying the convolution operator on the corresponding homogeneous space $\mathbb{D}$ rather than on the group itself in order to leverage on Helgason-Fourier analysis, following the ideas introduced for compact group actions in [14] and generalized in [7].

Finally, we have applied our approach to the problem of classifying THPD matrices arising in the context of Doppler signals processing. We propose here an alternative approach to [5] by using a $\mathrm{SU}(1,1)$ equivariant neural network operating on the hyperbolic representation of the THPD matrices manifold.

1.2 Preliminaries

We denote $\mathbb{D}$ the Poincaré unit disk $\mathbb{D} = \{z = x + iy \in \mathbb{C} / |z| < 1\}$ and by $\mathbb{T}$ the unit circle $\mathbb{T} = \{z = x + iy \in \mathbb{C} / |z| = 1\}$. We further introduce the following Lie groups:

$$\mathrm{SU}(1,1) = \left\{ g_{\alpha,\beta} = \begin{bmatrix} \alpha & \beta \\ \bar{\beta} & \bar{\alpha} \end{bmatrix}, |\alpha|^2 - |\beta|^2 = 1, \alpha, \beta \in \mathbb{C} \right\} \tag{1}$$

$$\mathrm{U}(1) = \left\{ \begin{bmatrix} \alpha / |\alpha| & 0 \\ 0 & \bar{\alpha} / |\alpha| \end{bmatrix}, \alpha \in \mathbb{C} \right\} \tag{2}$$

We can endow $\mathbb{D}$ with a transitive action $\circ$ of $\mathrm{SU}(1,1)$ defined as it follows

$$\forall g_{\alpha,\beta} \in \mathrm{SU}(1,1),\ \forall z \in \mathbb{D},\ g_{\alpha,\beta} \circ z = \frac{\alpha z + \beta}{\bar{\beta} z + \bar{\alpha}} \tag{3}$$

The Cartan decomposition associated with $\mathrm{SU}(1,1)$ allows building a diffeomorphism between $\mathrm{SU}(1,1)$ (seen as a topological space) and $\mathbb{T}\times\mathbb{D}$ [11]. As $\mathbb{T}\simeq U(1)$ as a group, we have $\mathbb{D}\simeq \mathrm{SU}(1,1)/\,\mathrm{U}(1)$ and we can therefore associate $z\in\mathbb{D}$ with an element of the quotient space $\mathrm{SU}(1,1)/\,\mathrm{U}(1)$.

We consider in the following a supervised learning set-up in which the input data space $\mathscr{X}$ corresponds to functionals defined on $\mathbb{D}$ and valued in a vectorial space V. Hence, a data point $x_i\in\mathscr{X}$ will be represented by the graph of a function $f_{x_i}:\mathbb{D}\to V$, so that $x_i=\{(z,f_{x_i}(z)),\ z\in\mathbb{D}\}$. We then extend the action $\circ$ of $\mathrm{SU}(1,1)$ on $\mathbb{D}$ to the considered input data space by writing, $\forall g\in\mathrm{SU}(1,1)$ and $\forall x\in\mathscr{X}$, $\mathrm{g}\circ x=\{(z,f_x(g\circ z)),\ z\in\mathbb{D}\}$.

2 Equivariant Convolution Operator

We introduce in this section a convolution operator defined on the Poincaré disk $\mathbb{D}$ and which is equivariant with respect to the action of $\mathrm{SU}(1,1)$. Corresponding feature maps can then be embedded within usual neural network topologies. After introducing local convolution kernels, we discuss the numerical aspects associated with practical computations.

2.1 Convolution on $\mathbb{D}$

Following [13], we will denote by μ^G the Haar measure of $G=\mathrm{SU}(1,1)$ that is normalized according to $\int_G F(g\circ 0_{\mathbb{D}})\,d\mu^G(g)=\int_{\mathbb{D}}F(z)\,dm(z)$ for all $F\in L^1(\mathbb{D},dm)$, where $L^1(X,d\mu)$ refers to the set of functions from X to $\mathbb{C}$ which are integrable with respect to the measure μ, with $0_{\mathbb{D}}$ the center of $\mathbb{D}$ and where the measure dm is given for $z=z_1+iz_2$ by $dm(z)=\frac{dz_1dz_2}{\left(1-|z|^2\right)^2}$. We can then build feature maps $\psi:\mathbb{D}\to\mathbb{C}$ by leveraging on the convolution operator on $\mathbb{D}$.

More precisely, for a kernel function $k_\theta:\mathbb{D}\to\mathbb{C}$ parameterized by $\theta\in\mathbb{R}^\ell$, an input feature map $f:\mathbb{D}\to\mathbb{C}$ and an element $z\in\mathbb{D}$, we will consider the following operator ψ^θ

$$\psi^\theta(z)=(k_\theta\star_{\mathbb{D}}f)(z)=\int_G k_\theta\left(g^{-1}\circ z\right)f(g\circ 0_{\mathbb{D}})\,d\mu^G(g) \tag{4}$$

as long as the right handside integral is well defined. The operator ψ^θ can actually be seen as a specific group-convolution operator which is constant over each coset so that it only requires to be evaluated on $\mathbb{D}$.

Denoting L_g the left shift operator such that $\forall z\in\mathbb{D}$, $\forall g\in G$ and $\forall\phi:\mathbb{D}\to\mathbb{C}$, $L_g\phi(z)=\phi\left(g^{-1}\circ z\right)$, we show that ψ^θ is equivariant with respect to the action of G by writing:

$$\begin{aligned}L_h\psi^\theta(z)&=\int_G k_\theta\left(g^{-1}\circ\left(h^{-1}\circ z\right)\right)f(g\circ 0_{\mathbb{D}})\,d\mu^G(g)\\&\overset{\text{action}}{=}\int_G k_\theta\left((hg)^{-1}\circ z\right)f(g\circ 0_{\mathbb{D}})\,d\mu^G(g)\\&\overset{\text{Haar}}{=}\int_G k_\theta\left(g^{-1}\circ z\right)L_hf(g\circ 0_{\mathbb{D}})\,d\mu^G(g)=k_\theta\star_{\mathbb{D}}(L_hf)\end{aligned}$$

2.2 Kernel Modeling and Locality

To accomodate with the Euclidean nature of a wide range of kernel functionals, we propose projecting the input data to k_θ to a vectorial space with the logarithm map $\log_{\mathbb{D}}$ from $\mathbb{D}$ to its tangent space $(\mathcal{T}\mathbb{D})_{0_{\mathbb{D}}}$ at its center. Following [12], we will generically model the kernel functional as a simple neural network. However, for the sake of computational efficiency, specific parameterizations can also be envisioned such as the use of a B-splines expansion as in [2], or more generically, the use of a well suited functional expansion in $(\mathcal{T}\mathbb{D})_{0_{\mathbb{D}}}$.

To localize the kernel action, we leverage on the geodesic distance in the Poincaré disk $\rho_{\mathbb{D}}$ and we then restrict the integration in the convolution operator to the points "close" to z according to the metric $\rho_{\mathbb{D}}$, leading to

$$\psi^{\theta,M}(z) = \int_{\rho_{\mathbb{D}}(z, g\circ 0_{\mathbb{D}}) \leq M} k_\theta \left(\log_{\mathbb{D}}\left(g^{-1}\circ z\right)\right) f\left(g \circ 0_{\mathbb{D}}\right) d\mu^G(g) \tag{5}$$

where M is a threshold to be considered as a model hyperparameter and which could be assimilated as the filter bandwidth in standard CNN. It should also be noted that the localization of the kernel function ensures the definition of the integral in the right handside of (5).

2.3 Numerical Aspects

In practice, the integral (5) cannot be computed exactly and a numerical scheme is therefore required for evaluating the convolution maps on a discrete grid of points in $\mathbb{D}$. In this section, we tackle the scalability challenges associated with $\mathrm{SU}(1,1)$ equivariant networks operating on the hyperbolic space $\mathbb{D}$.

Monte-Carlo Method. Following [12], a first approach to the estimation of (5) is to compute, $\forall z \in \mathbb{D}$, the following Monte-Carlo estimator,

$$\psi_N^{\theta,M}(z) = \frac{1}{N}\sum_{i=1}^{N} k_\theta\left(\log_{\mathbb{D}}\left(g_i^{-1}\circ z\right)\right) f\left(g_i \circ 0_{\mathbb{D}}\right) \tag{6}$$

where the $g_i \in \mathrm{SU}(1,1)$ are drawn according to the conditional Haar measure $\mu^G\big|_{\mathcal{B}(z,M)}$, where $\mathcal{B}(z,M)$ is the ball centered in z and of radius M measured according to the metric $\rho_{\mathbb{D}}$. To do so efficiently, we first sample elements in $\mathcal{B}(0_{\mathbb{D}},M)$ and then move to $\mathcal{B}(z,M)$ through the group action. The obtained coset elements are then lifted to $\mathrm{SU}(1,1)$ by sampling random elements of $\mathrm{U}(1)$.

Although appealing from a an implementation standpoint, this approach is severely limited by its exponential complexity in the number of layers of the considered neural network. To illustrate this point, let's consider a simple architecture with ℓ convolutional layers of one kernel each and let's assume that we are targeting an evaluation of the last convolution map of the layer ℓ on m points. By denoting n_i the number of evaluation points required for the evaluation of $\psi_i^{\theta_i}$ and by using a simple induction, we show that $n_i = mN^{\ell-i+1}$, for $1 \leq i \leq \ell$.

Approaches such as [17], further generalized in [12], have been introduced to reduce the computational cost by using a clever reordering of the computations. However, the corresponding complexity remains exponential and therefore prevents from building deep networks with this Monte-Carlo approach, especially when working with groups such as $\mathrm{SU}(1,1)$ which require significant sampling.

Helgason-Fourier Analysis. The use of Fourier analysis within G-CNN architectures to evaluate convolution operators defined on Euclidean spaces has been previously suggested [9,14]. By leveraging on similar ideas, we here propose an alternative approach to the Monte-Carlo sampling on $\mathrm{SU}(1,1)$ by using Helgason-Fourier (HF) transforms [13] of functions defined on the hyperbolic space $\mathbb{D}$.

For points $b \in \partial\mathbb{D}$ and $z \in \mathbb{D}$, we denote $\langle z, b\rangle$ the algebraic distance to the center of the horocyle based at b through z. Let's then consider $f : \mathbb{D} \to \mathbb{C}$ such that $z \to f(z)e^{(1-i\rho)\langle z,b\rangle} \in L^1(\mathbb{D}, dm)$, $\forall \rho \in \mathbb{R}$. The HF transform of f is then defined on $\mathbb{R} \times \partial\mathbb{D}$ by

$$\mathcal{HF}(f)(\rho, b) = \int_{\mathbb{D}} f(z)\, e^{(1-i\rho)\langle z,b\rangle} dm(z) \tag{7}$$

[13] gives the formal expression for the horocyclic wave $e^{\nu\langle z,b\rangle}$, for $\nu \in \mathbb{C}$. Assuming corresponding integrability conditions, the HF transform of the convolution of two functions $F_1, F_2 : \mathbb{D} \to \mathbb{C}$ is given by

$$\mathcal{HF}(F_1 \star_{\mathbb{D}} F_2)(\rho, b) = \mathcal{HF}(F_1)(\rho, b)\, \mathcal{HF}(F_2)(\rho, b) \tag{8}$$

As for the Euclidean case, an inversion formula also exists and is given by

$$f(z) = \frac{1}{4\pi} \int_{\mathbb{R}} \int_{\partial\mathbb{D}} \mathcal{HF}(f)(\rho, b)\, e^{(1+i\rho)\langle z,b\rangle} \rho \tanh\left(\frac{\pi\rho}{2}\right) d\rho db \tag{9}$$

Using HF transforms allows working with independent discretization grids for the convolution maps computation across the network structure, but the pointwise evaluation of each map requires the computation of three integrals, instead of one in the Monte-Carlo approach. Also, the numerical estimation of the integrals (7) and (9) is quite challenging and leads to convergence issues with classical schemes. We have obtained satisfactory results by using the Quasi-Monte-Carlo approach [4] and the corresponding GPU-compatible implementation provided by the authors. A differentiable estimation of $\mathcal{HF}\left(z \to k_{\theta_i}(z)\, \mathbf{1}_{\rho_{\mathbb{D}}(0_{\mathbb{D}}, z) \leq M}\right)$ can therefore be obtained efficiently by using a well chosen functional expansion of the kernel (e.g., basis of the Fock-Bargmann Hilbert space [11], B-splines [3], eigenfunctions of the Laplacian on $\mathbb{D}$ [13]) and by pre-computing the HF transforms of the corresponding basis elements.

3 Application to THPD Matrix Classification

Let's denote by $\mathcal{T}_n^+$ the set of Toeplitz Hermitian Positive Definite (THPD) matrices of size n. As described in [6], the regularized Burg algorithm allows transforming a given matrix $\Gamma \in \mathcal{T}_n^+$ into a power factor in $\mathbb{R}_+^*$ and reflection coefficients μ_i in a $(n-1)$ dimensional product of Poincaré spaces $\mathbb{D}^{n-1} = \mathbb{D} \times ... \times \mathbb{D}$. Combining this with the formalism introduced in Sect. 1.2, we can therefore see an element $\Gamma \in \mathcal{T}_n^+$ as a power factor in $\mathbb{R}_+^*$ and $n-1$ coset elements in $\mathrm{SU}(1,1)/\mathrm{U}(1)$.

In the following, we will focus on radar clutter classification and consider in this context rescaled THPD matrices which can be represented by the reflection coefficients only. We further assume that the ordering of these coefficients is not material for our considered classification task, so that we will represent a given element $\Gamma \in \mathcal{T}_n^+$ by folding into $\mathbb{D}$ with the function $f_\Gamma : \mathbb{D} \to \mathbb{C}$, with $f_\Gamma(\mu) = \sum_{i=1}^{n-1} \mu^{\mathbb{C}} \mathbf{1}_{\{\mu=\mu_i\}}$ for $\mu \in \mathbb{D}$ and where $\mu \to \mu^{\mathbb{C}}$ is the canonical embedding from $\mathbb{D}$ to $\mathbb{C}$.

3.1 Classification Problem

We have considered the problem of pathological radar clutter classification as formulated in [6]. More precisely, we represent each cell by its THPD auto-correlation matrix Γ, our goal being to predict the corresponding clutter $c \in \{1, ..., n_c\}$ from the observation of Γ. Within our formalism, the training samples are of the form (f_{Γ_i}, c_i) and the input data have been obtained by simulating the auto-correlation matrix of a given cell according to

$$Z = \sqrt{\tau} R^{1/2} x + b_{radar} \tag{10}$$

where τ is a positive random variable corresponding to the clutter texture, R a THPD matrix associated with a given clutter, $x \sim \mathcal{N}_{\mathbb{C}}(0, \sigma^x)$ and $b_{radar} \sim \mathcal{N}_{\mathbb{C}}(0, \sigma)$, with $\mathcal{N}_{\mathbb{C}}(0, t)$ referring to the complex gaussian distribution with mean 0 and standard deviation t. In the following, b_{radar} will be considered as a source of thermal noise inherent to the sensor.

3.2 Numerical Results

We have instanciated a simple neural network constituted of one $\mathrm{SU}(1,1)$ convolutional layer with two filters and ReLu activation functions, followed by one fully connected layer and one softmax layer operating on the complex numbers represented as 2-dimensional tensors. The kernel functions are modeled as a neural networks with one layer of 16 neurons with swish activation functions. The two convolution maps have been evaluated on the same grid constituted of 100 elements of $\mathbb{D}$ sampled according to the corresponding volume measure.

To appreciate the improvement provided by our approach, we will compare the obtained results with those corresponding to the use of a conventional neural network with roughly the same numbers of trainable parameters and operating on the complex reflection coefficients. In the following, we will denote $\mathcal{N}_\sigma^G$ (resp. $\mathcal{N}_\sigma^{FC}$) the neural network with $\mathrm{SU}(1,1)$ equivariant convolutional (resp. fully connected) layers and trained on 400 THPD matrices of dimension 10 corresponding to 4 different classes (100 samples in each class) which have been simulated according to (10) with a thermal noise standard deviation σ.

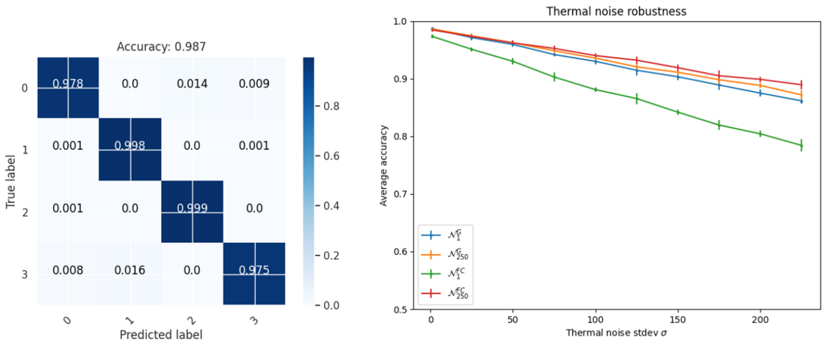

Fig. 1. Left handside: confusion matrix corresponding to the evaluation of $\mathcal{N}_1^G$ on the testing set T_1, averaged over 10 realizations. Right handside: average accuracy results of the algorithms $\mathcal{N}_1^G$, $\mathcal{N}_1^{FC}$, $\mathcal{N}_{250}^G$ and $\mathcal{N}_{250}^{FC}$ on the testing sets T_σ shown as a function of σ, together with the corresponding standard deviation as error bars

In order to evaluate the algorithms $\mathcal{N}_\sigma^G$ and $\mathcal{N}_\sigma^{FC}$, we have considered several testing sets T_σ consisting in 2000 THPD matrices of dimension 10 (500 samples in each of the 4 classes) simulated according to (10) with a thermal noise standard deviation σ. The obtained results are shown on Fig. 1 where it can in particular be seen that $\mathcal{N}_1^G$ reaches similar performances as $\mathcal{N}_{250}^G$ and $\mathcal{N}_{250}^{FC}$ while significantly outperforming $\mathcal{N}_1^{FC}$ as σ increases, meaning that our approach allows to achieve some degree of robustness with respect to the variation of σ through the use of equivariant layers.

4 Conclusions and Further Work

Following recent work on G-CNN specification, we have introduced an approach for building $\mathrm{SU}(1,1)$ equivariant neural networks and have discussed how to deal with corresponding numerical challenges by leveraging on Helgason-Fourier analysis. We have instanciated the proposed algorithm for the problem of THPD matrix classification arising in the context of Doppler signals processing and have obtained satisfactory accuracy and robustness results. Further work will include studying $\mathrm{SU}(n,n)$ equivariant neural networks and their applicability to the problem of classifying Positive Definite Block-Toeplitz matrices by leveraging on their embedding into Siegel spaces [1]. We will also investigate the link with the coadjoint representation theory which may allow generalizing our approach while transferring convolution computation techniques to coadjoint orbits [16].

References

1. Barbaresco, F.: Radar micro-doppler signal encoding in Siegel unit poly-disk for machine learning in Fisher metric space. In: 2018 19th International Radar Symposium (IRS), pp. 1–10 (2018). https://doi.org/10.23919/IRS.2018.8448021

2. Bekkers, E.J.: B-spline CNNs on lie groups. In: International Conference on Learning Representations (2020). https://openreview.net/forum?id=H1gBhkBFDH
3. Bekkers, E.J., Lafarge, M.W., Veta, M., Eppenhof, K.A., Pluim, J.P., Duits, R.: Roto-translation covariant convolutional networks for medical image analysis (2018)
4. Borowka, S., Heinrich, G., Jahn, S., Jones, S., Kerner, M., Schlenk, J.: A GPU compatible Quasi-Monte Carlo integrator interfaced to pySecDec. Comput. Phys. Commun. **240**, 120–137 (2019). https://doi.org/10.1016/j.cpc.2019.02.015
5. Brooks, D., Schwander, O., Barbaresco, F., Schneider, J., Cord, M.: A Hermitian positive definite neural network for micro-doppler complex covariance processing. In: 2019 International Radar Conference (RADAR), pp. 1–6 (2019). https://doi.org/10.1109/RADAR41533.2019.171277
6. Cabanes, Y., Barbaresco, F., Arnaudon, M., Bigot, J.: Toeplitz Hermitian positive definite matrix machine learning based on Fisher metric. In: Nielsen, F., Barbaresco, F. (eds.) GSI 2019. LNCS, vol. 11712, pp. 261–270. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-26980-7_27
7. Chakraborty, R., Banerjee, M., Vemuri, B.C.: H-CNNs: convolutional neural networks for Riemannian homogeneous spaces. arXiv preprint arXiv:1805.05487 1 (2018)
8. Cohen, T., Welling, M.: Group equivariant convolutional networks. In: Balcan, M.F., Weinberger, K.Q. (eds.) Proceedings of The 33rd International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 48, pp. 2990–2999. PMLR, New York, 20–22 June 2016. http://proceedings.mlr.press/v48/cohenc16.html
9. Cohen, T.S., Geiger, M., Köhler, J., Welling, M.: Spherical CNNs. CoRR abs/1801.10130 (2018)
10. Cohen, T.S., Geiger, M., Weiler, M.: A general theory of equivariant CNNs on homogeneous spaces. In: Wallach, H., Larochelle, H., Beygelzimer, A., Alché-Buc, F., Fox, E., Garnett, R. (eds.) Advances in Neural Information Processing Systems, vol. 32, pp. 9145–9156. Curran Associates, Inc. (2019)
11. del Olmo, M.A., Gazeau, J.P.: Covariant integral quantization of the unit disk. J. Math. Phys. **61**(2) (2020). https://doi.org/10.1063/1.5128066
12. Finzi, M., Stanton, S., Izmailov, P., Wilson, A.G.: Generalizing convolutional neural networks for equivariance to lie groups on arbitrary continuous data (2020)
13. Helgason, S.: Groups and geometric analysis (1984)
14. Kondor, R., Trivedi, S.: On the generalization of equivariance and convolution in neural networks to the action of compact groups. In: Dy, J., Krause, A. (eds.) Proceedings of the 35th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 80, pp. 2747–2755. PMLR, Stockholmsmässan, 10–15 July 2018. http://proceedings.mlr.press/v80/kondor18a.html
15. Lecun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proc. IEEE **86**(11), 2278–2324 (1998). https://doi.org/10.1109/5.726791
16. Rieffel, M.A.: Lie group convolution algebras as deformation quantizations of linear poisson structures. Am. J. Math. **112**(4), 657–685 (1990)
17. Wu, W., Qi, Z., Fuxin, L.: PointConv: deep convolutional networks on 3d point clouds. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 2019