

## Spherical Convolutional Neural Networks

In the last 5 years, the field of Machine Learning has been revolutionized by the success of Deep Learning. Thanks to the increasing availability of data and computations, we now are able to train very complex and deep models to solve challenging tasks better than we ever did.

Nevertheless, Deep Learning is successful when the network architecture exploits properties of the data, allowing efficient and principled learning. For example, convolutional neural networks (CNNs) revolutionized computer vision because the network architecture has been specifically designed to deal with images. The main characteristic of CNNs is to be equivariant to translation: if the input is translated, so is the output. The translation equivariance property is extremely valuable in dense tasks such as segmentation. For global tasks such as object recognition, translation invariance is sought: a translation of the input image should not result in a change of class. This property of images and the adapted CNN architecture enables the spatial sharing of weights that dramatically reduces the number of parameters to be learned. By exploiting translation equivariance, CNNs exhibit lower computational and learning complexities on data that satisfies this property.

Beyond images, we need architectures adapted to other kinds of data, encoding both domain specific knowledge and data specific characteristics. For instance, spherical data is very common in (i) climate science with data on the Earth, (ii) in cosmology, where most observations are made from the Earth (see Figure 1), and (iii) in virtual reality, where one often works with user-centered  $360^\circ$  images. Spherical data is represented by pixels that live on the sphere. They are like curved images, but without borders and a potentially arbitrary orientation. Similarly to images, we'd like our architectures to exploit properties of the spherical domain. Instead of translation, the spherical domain naturally suggests equivariance to the rotation group  $SO(3)$ : a rotation of the input implies the same rotation of the output.

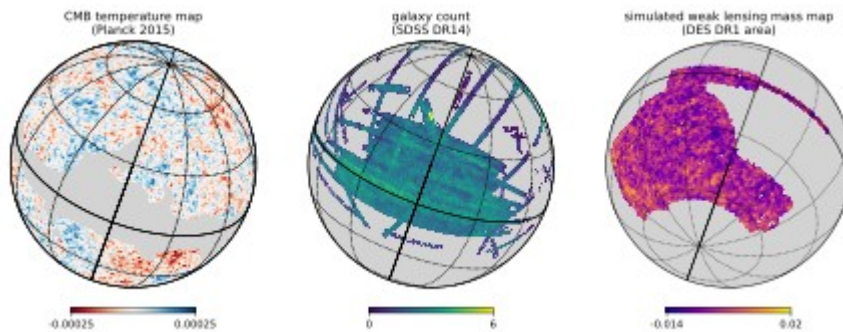


Figure 1: Example maps on the sphere: (left) the cosmic microwave background (CMB) temperature map from Plank, (middle) map of galaxy number counts, and (right) simulated weak lensing convergence map.

So far, two approaches have been followed. In the first, the data is transformed using a planar projection and a modified CNN is applied (see for example [1]). This strategy has the advantage to be built on top of a traditional CNN and hence to be efficient. Nevertheless, the distortions induced by the projection make the translation equivariance property of CNNs different from the desired rotation equivariance. In simple words, we are destroying the spherical structure of the data. The second approach [2, 3] leverages the convolution on the

SO(3) rotation group. This convolution is a generalization of the planar convolution for the sphere and similarly it can be performed by a multiplication in the spectral/Fourier domain. In this case, rotation equivariance is naturally obtained. However, the computational cost of the spectral projections (Fourier transforms) is important, limiting the size and the depth of these architectures.

In this project, you will work with an architecture that is almost rotation equivariant while remaining computationally inexpensive. [4, 5] The idea is to perform the convolution on a graph that approximates the sphere. The graph is a discrete model of the continuous 2D manifold. Similar to the traditional convolution, the graph convolution can be performed with a weighted average of neighboring pixels. Thanks to this property, we avoid computing Fourier transforms and obtain an operation with complexity linear in the data size.

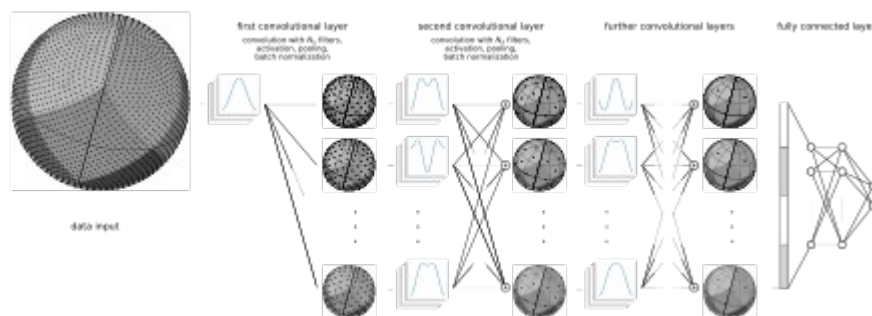


Figure 2: DeepSphere [5] overall architecture, showing here two convolutional layers acting as feature extractors followed by a fully connected layer with softmax acting as the classifier.

**Project Goal.** We aim here at making an extensive theoretical and empirical characterization of this new technique and benchmark it against the other spherical CNN architectures. In particular, the student will (i) collect and build some meaningful datasets, (ii) deal with the sphere pixelization as different architectures require different samplings, (iii) produce a reproducible research pipeline. This master thesis could potentially lead to a publication in a Machine Learning conference.

**Prerequisites.** Good knowledge of (Deep) Machine Learning and Python programming.

### Contact

Nathanaël Perraudin [nathanael.perraudin@sdsc.ethz.ch](mailto:nathanael.perraudin@sdsc.ethz.ch), Swiss Data Science Center, ETHZ

Michaël Defferrard [michael.defferrard@epfl.ch](mailto:michael.defferrard@epfl.ch), LTS2, EPFL

At their convenience, the student can work at EPFL or ETHZ.

### References

- [1] Boomsma, W., & Frellsen, J. (2017). Spherical convolutions and their application in molecular modelling.
- [2] Cohen, T. S., Geiger, M., Köhler, J., & Welling, M. (2018). Spherical CNNs.
- [3] Esteves, C., Allen-Blanchette, C., Makadia, A., & Daniilidis K. (2017). Learning SO(3) equivariant representations with spherical cnns.
- [4] Khasanova, R., & Frossard, P. (2017). Graph-based classification of omnidirectional images.
- [5] Perraudin, N., Defferrard, M., Kacprzak, T., & Sgier, R. (2018). DeepSphere: Efficient spherical Convolutional Neural Network with HEALPix sampling for cosmological applications.