ICLR spotlight talk ~~Addis Ababa~~ virtual, 2020-04-30

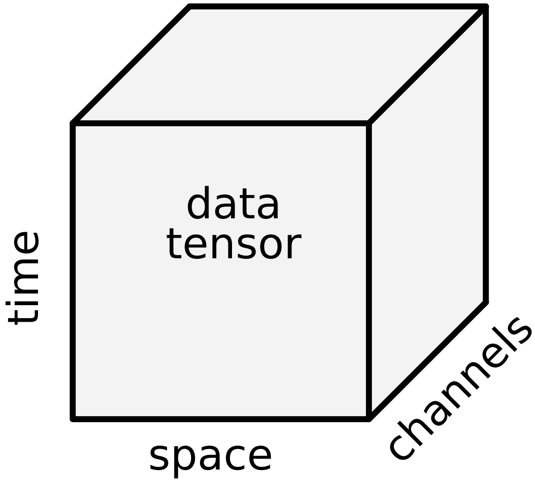
DeepSphere: a graph-based spherical CNN

Michaël Defferrard

Joint work with Martino Milani, Frédérick Gusset, Nathanaël Perraudin.

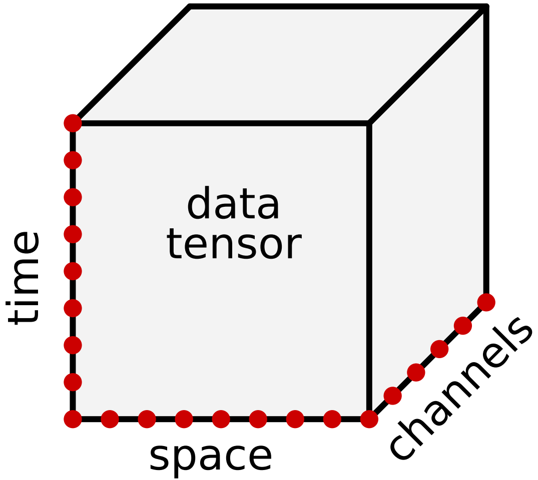
EPFL

Structured data

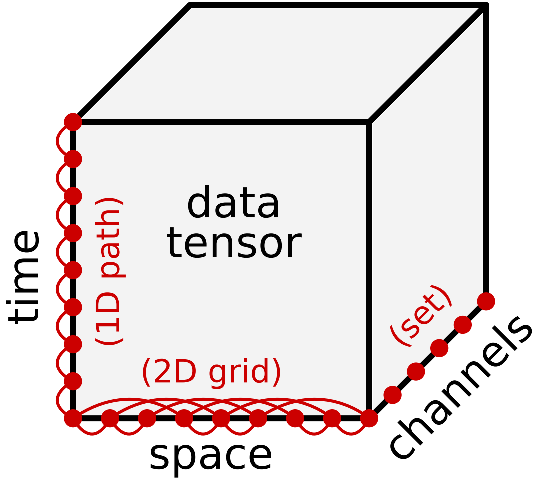


- data is multi-dimensional

Structured data



- data is multi-dimensional
- measurements are discrete

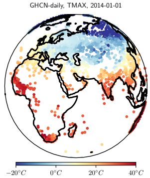


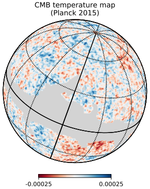
- data is multi-dimensional
- measurements are discrete
- dimensions are structured

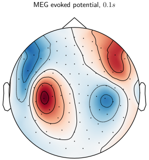
Problem: learning from spherical data

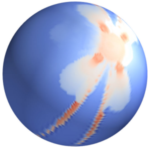
input $x : S^2 \times \cdots \to \mathbb{R}^d$

intrinsic	projection

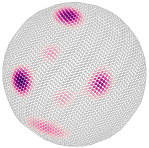






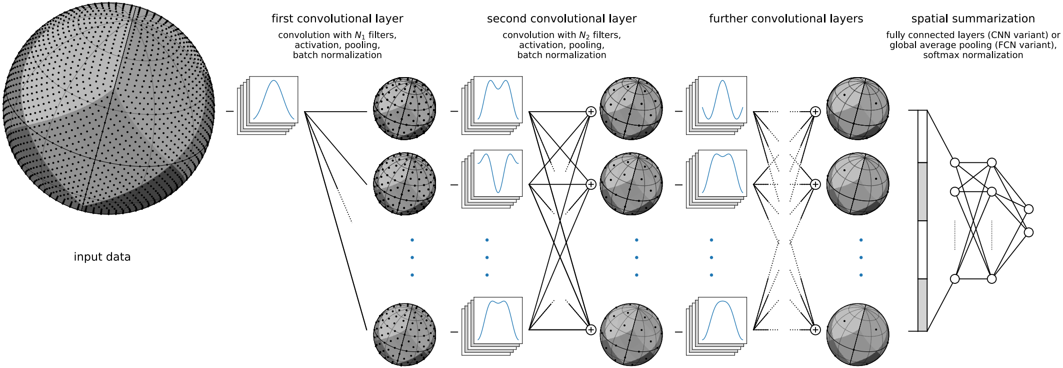


output $f(x)$

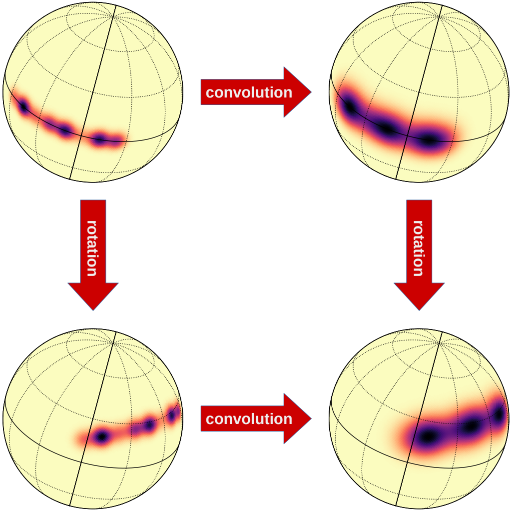
	classification	regression
global	"sleepy" / "alert"	$\sigma_8, \ \Omega_m$
dense		

Acoustic field from Simeoni et al. 2019. 3D shape from Esteves et al. 2018.

Solution: spherical neural networks



Desideratum 1: equivariant to rotations



- *Equivariance* for dense tasks:
  $f(Rx) = Rf(x)\ \forall R \in SO(3)$
- *Invariance* for global tasks:
  $f(Rx) = f(x)\ \forall R \in SO(3)$

Why exploit symmetries?

- reduced sample complexity
- generalization guarantee

$\Rightarrow$ principled convolution (weight sharing)

Desideratum 2: scalable

- Many inferences needed for training.
- Increasingly larger maps. ($n = 10^7$ pixels is customary in cosmology.)

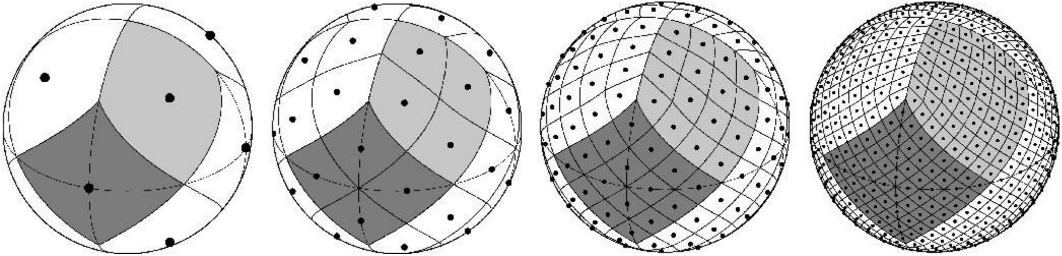
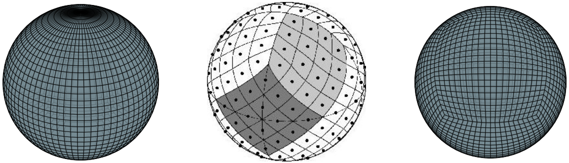
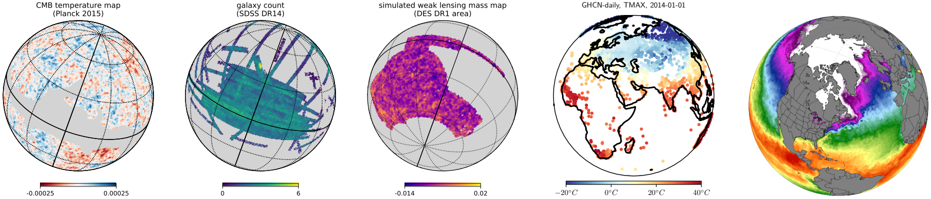


Figure from `https://healpix.sourceforge.io`.

Desideratum 3: flexible sampling



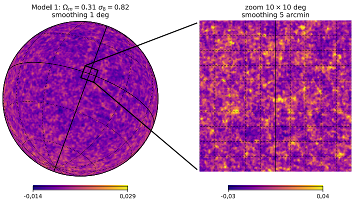
Sampling schemes: equiangular, HEALPix, cubed-sphere, icosahedral, Gauss-Legendre, etc.

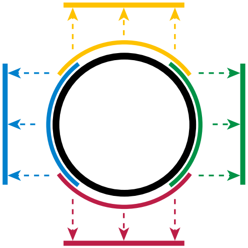


Partial and irregular sampling.

Some figures from Boomsma and Frellsen 2017 and `https://climatereanalyzer.org`.

Method 1: 2D projections





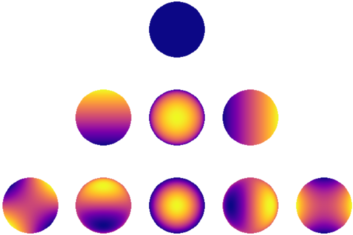
Manifold is locally Euclidean!
Project on 2D tangent planes.

Desiderata

- ⊖ Rotation equivariance: hard to properly glue planes together.
- ⊕ Scalability: well developed NN architectures and implementations. Some wastes at boundaries.
- ⊖ Flexibility: only handle compact subspaces.

Charting figure from `https://en.wikipedia.org/wiki/manifold`.

Method 2: discretization of continuous domain



Spectral decomposition.

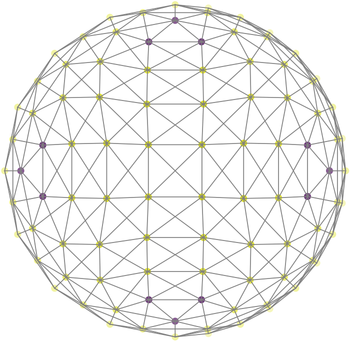
Discretize the continuous problem!
Compute the spherical harmonic transform (SHT), filter in the spectrum.

Desiderata

- ⊕ Rotation equivariance: well understood theory.
- ⊖ SHT is expensive. Fast transforms exist for some samplings.
- ⊖ Flexibility: unused pixels are mostly wasted.

Figure from `https://rodluger.github.io/starry`.

Our proposition: discrete domain



graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$ with
$A_{ij} = \exp(-d(z_i, z_j)/\sigma)$

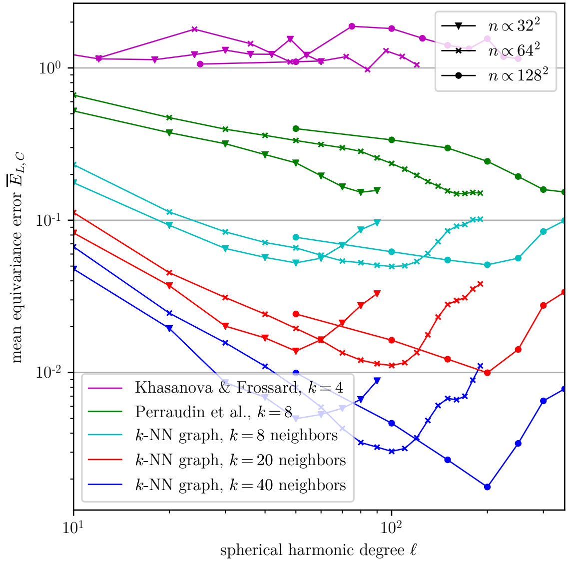
Domain set of pixels $\mathcal{V}$
topology given by geodesic distances

Data function $x : \mathcal{V} \to \mathbb{R}$
seen as $x \in \mathbb{R}^n$

Method in a nutshell (Defferrard et al. 2016)

1. Model the topology by a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$.
2. From it stems a Laplacian, e.g. $L = D - A$.
3. The Fourier basis diagonalizes the Laplacian.
4. Convolution is a multiplication in Fourier.
5. Spatial implementation for speed,
   e.g. $g_\alpha(L)x = \sum_k \alpha_k L^k x$.

Desideratum 1: equivariant to rotations



- Difficulty: set the edge weights.
- Equivariance error:

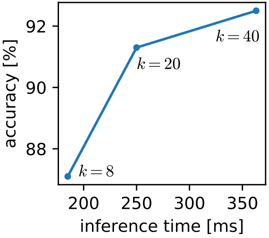
$$\mathbb{E}_{R,x} \left(\frac{\|RLx - LRx\|}{\|Lx\|} \right)^2$$

- Tradeoff between equivariance and cost (number of vertices n and edges kn)!

Desideratum 1: it matters!

	accuracy	time
Perraudin et al. 2019, 2D CNN baseline	54.2	104 ms
Perraudin et al. 2019, CNN variant, $k = 8$	62.1	185 ms
Perraudin et al. 2019, FCN variant, $k = 8$	83.8	185 ms
$k = 8$ neighbors, optimal t	87.1	185 ms
$k = 20$ neighbors, optimal t	91.3	250 ms
$k = 40$ neighbors, optimal t	92.5	363 ms

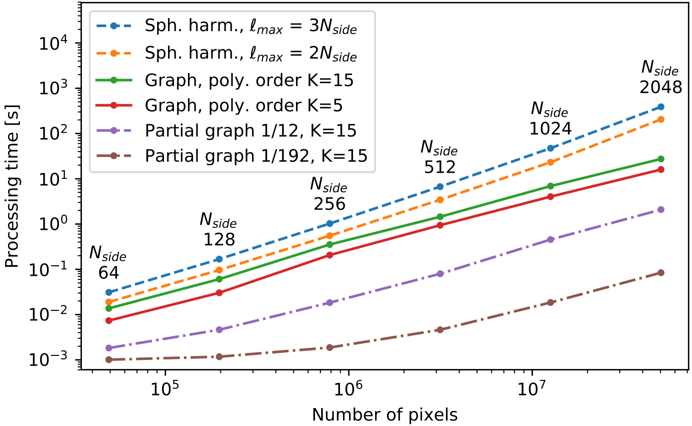
Lower equivariance error translates to higher performance.



Tradeoff between cost and accuracy.

Desideratum 2: scalable

- Graph convolutions cost $\mathcal{O}(n)$.
- Spherical convolutions cost $\mathcal{O}(n^2)$ in general, $\mathcal{O}(n^{3/2})$ for some samplings.

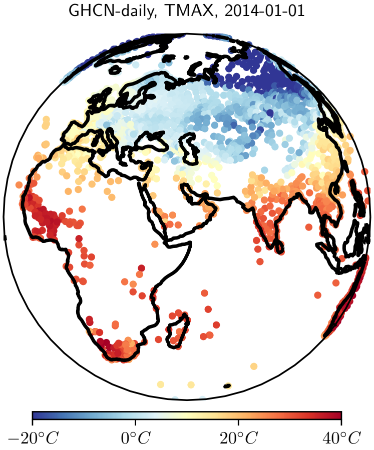


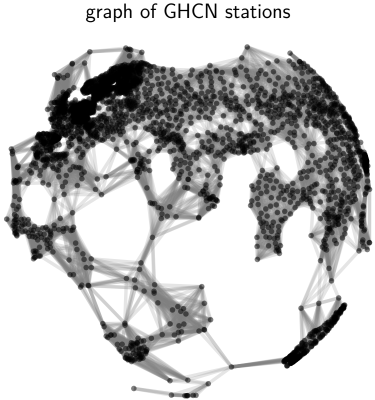
Desideratum 2: it matters!

	performance		size	speed	
	F1	mAP	params	inference	training
Cohen et al. 2018 ($b = 128$)	-	67.6	1400 k	38.0 ms	50 h
Cohen et al. 2018 (simplified, $b = 64$)	78.9	66.5	400 k	12.0 ms	32 h
Esteves et al. 2018 ($b = 64$)	79.4	68.5	500 k	9.8 ms	3 h
DeepSphere (equiangular, $b = 64$)	79.4	66.5	190 k	0.9 ms	50 m
DeepSphere (HEALPix, $N_{side} = 32$)	80.7	68.6	190 k	0.9 ms	50 m

Classification of 3D shapes (SHREC'17): anisotropy is an unnecessary price to pay.

Desideratum 3: flexible sampling

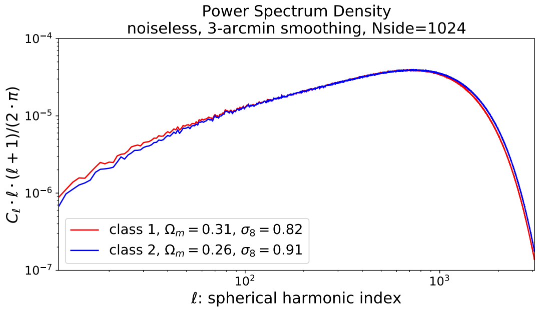




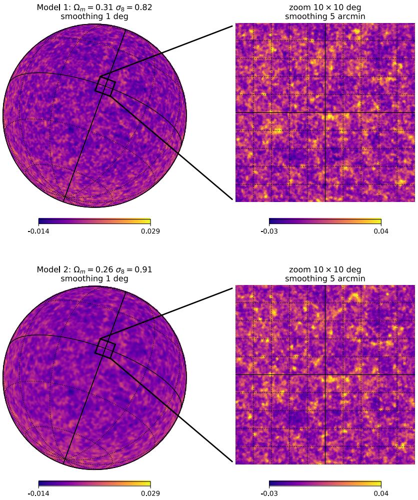
Application: discrimination of cosmological models

Classification of convergence maps created from two sets of cosmological parameters.

$$(\Omega_m, \sigma_8) = (0.31, 0.82) \text{ or } (0.26, 0.91)$$



Ω_m, σ_8, smoothing chosen to get identical PS.

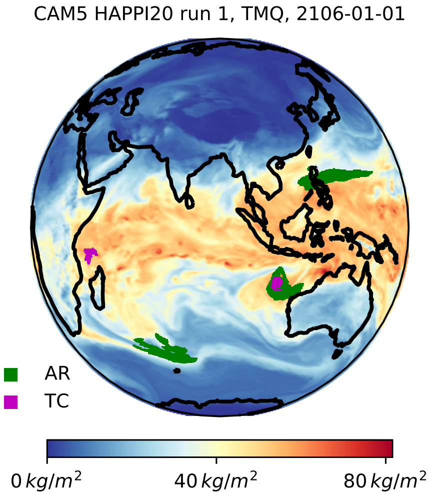


Maps with identical initial conditions.

Application: climate event segmentation

Segment extreme climate events: tropical cyclones (TC) and atmospheric rivers (AR).

- >1M spherical maps
- down-sampled to 10k pixels (original 900k)
- 0.1% TC, 2.2% AR, 97.7% background
- 16 channels (e.g., temperature, wind, humidity, pressure)



DeepSphere, a spherical CNN that strikes a controllable balance between desiderata.

Poster `https://iclr.cc/virtual/poster_B1e3OlStPB.html`

Slides `https://doi.org/10.5281/zenodo.3777976`

Papers Defferrard, Milani, Gusset, Perraudin, DeepSphere: a graph-based spherical CNN, ICLR, 2020.

Defferrard, Perraudin, Kacprzak, Sgier, DeepSphere: towards an equivariant graph-based spherical CNN, RLGM workshop at ICLR, 2019.

Perraudin, Defferrard, Kacprzak, Sgier, DeepSphere: Efficient spherical Convolutional Neural Network with HEALPix sampling for cosmological applications, Astronomy and Computing, 2019.

Code `https://github.com/deepsphere`
`https://github.com/epfl-lts2/pygsp`