Méthode d'apprentissage opérant sur la sphère: Application à la compression d'images omnidirectionnelles

Navid MAHMOUDIAN BIDGOLI[1], Roberto G. DE A. AZEVEDO[2], Thomas MAUGEY[1], Aline ROUMY[1], Pascal FROSSARD[3]

[1]Inria, Campus de Bealieu, 35042, Rennes Cedex, France

[2]ETH Zürich, Computer Graphics Laboratory, CH-8092 Zürich, Suisse

[2]EPFL, Laboratoire de Traitement des Signaux 4, CH-1015 Lausanne, Suisse

navid.mahmoudian-bidgoli@inria.fr, roberto.azevedo@inf.ethz.ch
thomas.maugey@inria.fr, aline.roumy@inria.fr, pascal.frossard@epfl.ch

Résumé – La popularité croissante pour les images 360° implique un besoin fort en outils avancés pour leur représentation et leur traitement. Pour contourner les difficultés liées à la topologie sphérique, la plupart des méthodes existantes utilisent des représentations planes de la sphère, occasionnant malheureusement des irrégularités dans la distribution spatiale du signal visuel. Dans cet article, nous proposons une boîte à outils opérant directement sur la sphère, permettant ainsi le développement de méthodes avancées d'apprentissage. Nous illustrons les bénéfices de l'approche proposée dans une application de compression.

Abstract – The growing popularity of 360° images implies a need of advanced tools for their representation and processing. In order to circumvent the problems due to the spherical topology, most of the existing methods use planar representation of the sphere, unfortunately providing an irregular pixel distribution. In this paper, we propose a set of tools working directly on the sphere, enabling the development of advanced learning methods. We illustrate the benefits of the proposed approach in a compression application.

1 Introduction

Les images et vidéos omnidirectionnelles, appelées également sphériques ou 360°, sont des signaux visuels dont le domaine de définition est une sphère. Les pixels disposés sur cette sphère décrivent l'ensemble de l'information visuelle convergeant en un point. Ce type d'images ou de vidéos permet ainsi une visualisation de la scène par l'utilisateur selon trois degrés de liberté (les trois types de rotation de la tête) et ainsi de donner une sensation d'immersion à celui-ci. Ce type de modalité est utilisé dans de nombreuses applications comme la réalité virtuelle ou la robotique. Dans le cadre de celles-ci, ces données sont l'objet de traitements complexes (*e.g.,* compression, segmentation, classification) généralement fondés sur des techniques récentes d'apprentissage profond. Cependant, la topologie sphérique, et donc non-euclidienne, du domaine sur lequel repose ces images, rende le déploiement de ces outils difficile. En effet, les opérations élémentaires composant une architecture de réseaux convolutifs (CNN) telles que la translation, le filtrage, l'échantillonnage, sont complexes à concevoir sur la sphère. Déployer des méthodes d'apprentissage capables d'opérer sur une surface sphérique constitue donc un enjeu scientifique majeur.

Actuellement, il existe trois approches pour développer une architecture CNN sur la sphère. La première consiste à projeter la sphère sur un ou plusieurs plans, se ramenant ainsi à un domaine de définition euclidien, et donc sur lequel les outils connus et performants de traitement d'images peuvent être déployés [9, 11, 14]. Cette approche a néanmoins le désavantage de déformer la topologie locale (les distances entre pixels), et de créer des coupures artificielles, ce qui diminue l'efficacité des méthodes d'apprentissage. La deuxième approche utilise la théorie des harmoniques sphériques afin de définir des outils opérant directement sur la sphère [2–4, 13]. Ainsi, il est possible d'effectuer des traitements comme le filtrage, la convolution, la translation et ainsi de concevoir des architectures CNN. En revanche, celles-ci restent prohibitives en temps de calculs de part la complexité des calculs liés aux harmoniques sphériques. Celles-ci nécessitent de plus de travailler sur la sphère entière (et non sur des fenêtres localisées) ce qui rend le traitement des images ou vidéos actuelles impossible de part leur grande résolution (> 10k). Dans la dernière approche, les avancées récentes en traitement du signal sur graphes sont utilisées afin de définir des convolutions directement dans le domaine spatial (sans calcul complexe de transformées) et sur les pixels directement (sans passer par le domaine continu) [7, 12, 15]. Modéliser la sphère par un graphe a ainsi permis de développer des architectures CNN très efficaces pour certaines tâches. Le défaut majeur de ce type d'approche est la faible expressivité des filtres. En effet, le graphe ne capturant pas la direction des voisins par rapport à un noeud, ceux-ci ne sont identifiés que par leur distance (sauf en utilisant plusieurs graphes accroissant en même temps la complexité [8]). Les filtres déve-

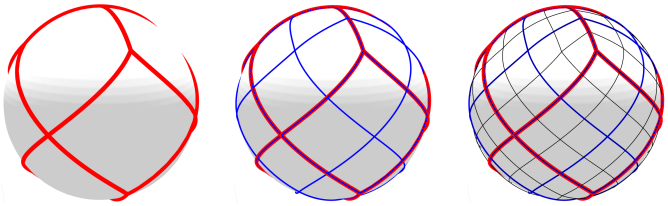

FIGURE 1 – Echantillonnage hiérarchique de la sphère selon la méthode HEALPix [6].

loppés sont ainsi isotropes. Si cela peut être avantageux dans certaines applications, il est indéniable que lorsque les données présentent des structures directionnelles particulières ou que les traitements deviennent plus complexes (*e.g.*, compression), il devient crucial de permettre le développement de filtres anisotropes.

Dans cet article, nous proposons justement de définir des convolutions qui sont à la fois expressives tout en restant peu complexes car calculées sur la sphère dans le domaine spatial discret. En utilisant l'échantillonnage HEALPix [6] possédant des propriété intéressantes, nous définissons une convolution comme une directe combinaison linéaire des voisins, tout en garantissant une cohérence du voisinage à tout endroit de la sphère. D'autres outils classiques des CNN sont également proposées afin d'élaborer une boîte à outils d'apprentissage (appelée OSLO) opérant directement sur la sphère. Nous illustrons l'efficacité de cette boîte à outils dans une application de compression d'images 360°.

2 Convolution proposée

Notre objectif est de définir une convolution qui puisse être calculée comme une simple combinaison linéaire de voisins, où chaque voisin serait affecté par un poids différent de celui des autres voisins. Il faut donc que le voisinage de chaque pixel comporte le même nombre d'éléments. De plus, il faut que chaque voisin soit identifiable (*e.g.,* le voisin au nord du pixel courant) pour qu'un poids spécifique puisse lui être assigné. Enfin, afin d'avoir des statistiques stationnaires à tout endroit de la sphère, il faut que la position relative du voisin (orientation et distance) par rapport au pixel central soit cohérente partout sur la sphère.

2.1 Echantillonnage HEALPix

Afin d'obtenir les propriétés énumérées ci-dessus, nous proposons d'utiliser l'échantillonnage HEALPix proposé dans [6] et illustré dans la Figure 1. La méthode d'échantillonnage repose sur une technique de pavage de la sphère à partir d'un dodécahèdre inscrit dans celle-ci. Plus précisément, les douze faces du dodécahèdre définissent 12 grande surfaces d'aires égales sur la sphère. Chacune d'elle est ensuite divisée en quatre sous-surfaces d'aires égales. Chacune de ces sous-surfaces peut ensuite être subdivisée en quatre sous-surfaces, et ce, jusqu'à obtenir la résolution désirée. Cette méthode présente des propriétés très utiles pour définir notre convolution. D'abord, chaque pixel sur la sphère a 8 voisins, sauf 24 d'entre eux qui n'en ont que 7, ce qui est négligeable pour de grandes résolutions. Ensuite, ces voisins ont une position relative stable en distance et direction, conséquence que l'échantillonnage produit des pixels de même aire et placés sur des cercles de même latitude (propriété isolatitude). Enfin, sa structure hiérarchique permet d'identifier très facilement les voisins ainsi que de concevoir des opérations de sous/sur-échantillonnage de la sphère.

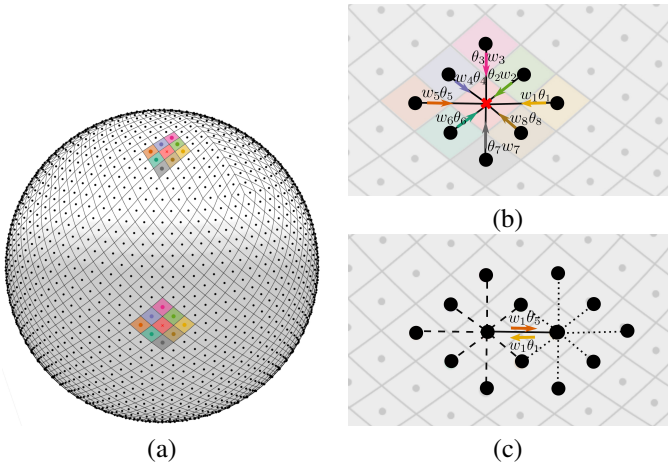

FIGURE 2 – Convolution proposée : (a) Noyau de convolution à différentes positions (chaque couleur représente un poids assigné à un voisin identifié), (b) implémentation "message-passing" efficace, (c) Une même paire de pixels est associée différemment selon la position du pixel central.

2.2 Convolution 1-hop

Soit $\mathbf{x}$ un vecteur décrivant l'image sphérique suivant l'indexation naturelle induite par HEALPix. Soit x_i l'information visuelle du pixel d'indice i. On note $\mathcal{N}_i(k)$, son voisin d'indice $1 \leq k \leq 8$ sur la sphère. Il est important de préciser que ce voisin est identifable quelque soit la position du pixel i sur la sphère (*e.g.,* au nord). On note $\boldsymbol{\theta}$ le noyau de convolution (*i.e.*, le vecteur de poids à apprendre dans un CNN). La convolution proposée s'écrit :

$$(\mathbf{x} \star \boldsymbol{\theta})(i) := \theta_0 \cdot x_i + \sum_{k=1}^{8} \theta_k \cdot x_{\mathcal{N}_i(k)} \cdot w_{\mathcal{N}_i(k),i} \qquad (1)$$

La fonction w permet de gérer les cas où seulement 7 voisins sont disponibles :

$$w_{\mathcal{N}_i(k),i} = \begin{cases} 1, & \text{si le voisin } \mathcal{N}_i(k) \text{ existe} \\ 0, & \text{sinon.} \end{cases}$$

Comme illustrée en Figure 2, cette définition de la convolution permet que son calcul en différents points de la sphère se fasse par une simple translation du noyau, comme pour une convolution en 2D. La convolution proposée est ainsi peu complexe (elle bénéfécie de plus d'une implémentation compatible avec le *message-passing* [5]).

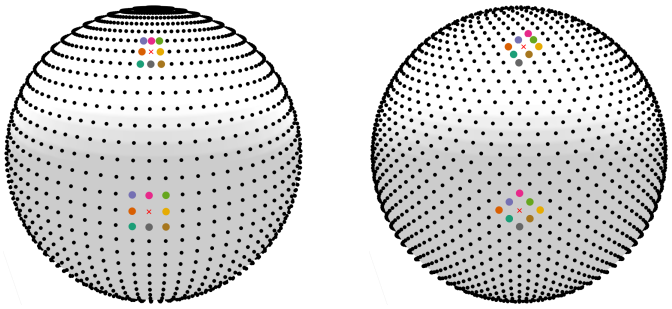

FIGURE 3 – Représentation du noyau de convolution à différents endroits de la sphères pour deux échantillonnages : équirectangulaire à gauche, et HEALPix à droite.

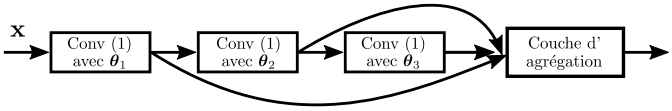

FIGURE 4 – Exemple d'extension à un voisinage à 3 sauts.

Comparaison avec la convolution basée échantillonnage équirectangulaire : La convolution proposée en Equation (1) peut s'appliquer dès lors que la sphère est échantillonnée de manière régulière (c'est à dire que presque tous les pixels ont le même nombre de voisins directs). C'est le cas, par exemple de l'échantillonnage correspondant à la projection équirectangulaire, *i.e.,* de type planisphère. En Figure 3, nous comparons le déplacement d'un noyau de convolution sur les deux échantillonnages équirectangulaire et HEALPix. Nous voyons la très forte variation entre les distances inter-pixels lorsque l'échantillonnage équirectangulaire est utilisé. Nous montrons dans la Section 3 comme une plus grande uniformité dans la distribution des pixels rend l'apprentissage plus efficace lorsque HEALPix est utilisé.

Comparaison avec la convolution basée graphe : à titre de comparaison, nous donnons maintenant l'expression de la convolution basée graphe, qui peut, par exemple, être employée sur l'échantionnage HEALPix [12]. Soit $\mathbf{L} = [l_{ij}]_{ij}$ le Laplacien normalisé du graphe où les noeuds sont les pixels et les arrêtes relient chaque pixel i à ses pixels adjacents $\{\forall k, \mathcal{N}_i(k)\}$. Si le noyau de convolution est décrit par le vecteur $\boldsymbol{\alpha}$, la convolution s'écrit alors :

$$\mathbf{x} \star \boldsymbol{\alpha} := \sum_{l=0}^{L} \alpha_l \mathbf{L}^l \mathbf{x}, \tag{2}$$

où L est le degré du polynôme de convolution qui régule la taille de la fenêtre de convolution (étant donné que $\mathbf{L}^l$ décrit le voisinage à l sauts dans le graphe). Si l'on écrit l'expression d'une telle convolution à un voisinage de $L = 1$, comme dans (1), on obtient :

$$(\mathbf{x} \star \boldsymbol{\alpha})(i) := \alpha_0 x_i + \alpha_1 . \left(\sum_{k=1}^{8} l_{i,\mathcal{N}_i(k)} x_{\mathcal{N}_i(k)} \right). \tag{3}$$

On voit bien qu'au lieu de discriminer chaque voisin par un poids différent comme dans (1), la convolution basée graphe a un poids par saut dans le voisinage, réduisant ainsi l'expressivité du filtre. Nous montrons dans la Section 3 les limites d'une telle approche.

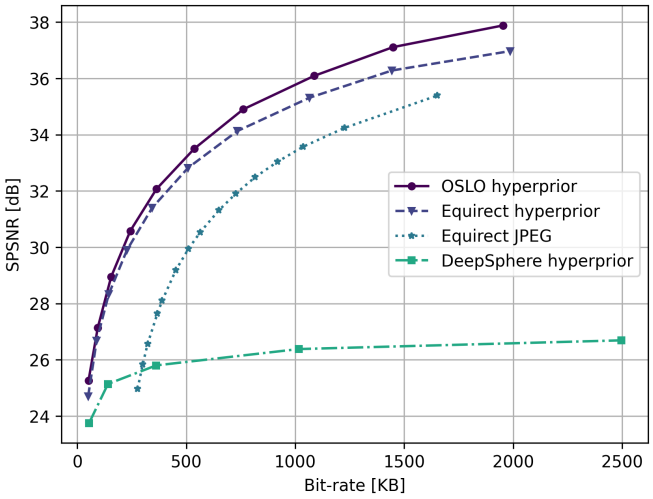

FIGURE 5 – Comparaison débit-distortion.

2.3 Convolution à n sauts

Une manière de généraliser notre convolution en (1) à une taille de voisinage quelconque serait d'appliquer le même principe : identifier les voisins ainsi que leur position relative par rapport au pixel central, et leurs assigner un poids différent. Bien que la navigation au voisin direct dans HEALPix soit très rapide, cela peut s'avérer bien plus complexe pour des voisins indirects. Nous proposons plutôt la stratégie illustrée dans la Figure 4. L'idée est d'utiliser la convolution proposée dans (1) de manière récursive et d'agréger les résultats par addition.

2.4 Boîte à outils OSLO

Bien qu'étant une opération cruciale, une architecture CNN ne se résume pas à la convolution. Ainsi, dans [10], nous avons proposé une boîte à outils complète incluant l'ensemble des briques de base d'une architecture CNN, tels que les *stride*, le *pooling*, l'*unpooling* et le *patching*. L'ensemble des ces outils reposent sur la structure hiérarchique de l'échantillonnage HEALPix.

3 Application à la compression

Nous illustrons maintenant l'efficacité de la boîte à outils OSLO dans le cadre de la compression d'images omnidirectionnelles. Nous partons d'une architecture bien connue en compression d'images classiques : scale hyperprior [1]. Nous en implémentons trois versions :

Equirect hyperprior : l'architecture de [1] est directement appliquée sur l'image équirectangulaire.

DeepSphere hyperprior : l'architecture de [1] est étendue sur la

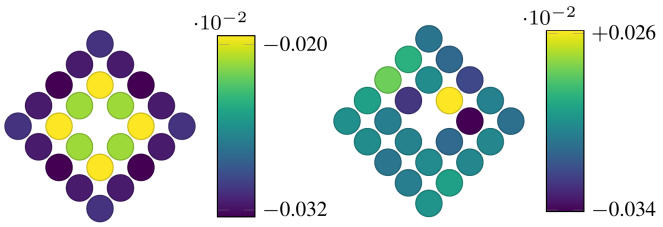

FIGURE 6 – Exemple de noyau de convolution pour la convolution basée graphe à gauche et pour la convolution proposée à droite.

sphère en utilisant la boîte à outils OSLO, mais la convolution basée graphe est utilisée.
OSLO hyperprior : l'architecture de [1] est étendue sur la sphère en utilisant la boîte à outils OSLO complète (incluant la convolution proposée).

Analyse des résultats : les résultats débit-distortions sont montrés en Figure 5. D'abord, on observe que l'approche proposée est plus performante que *Equirect hyperprior*, ce qui démontre l'intérêt de travailler directement sur la sphère. En effet, une meilleure cohérence du noyau de convolution sur la sphère améliore l'efficacité d'apprentissage de celui-ci. Ensuite, nous voyons que la méthode proposée est bien plus performante que la méthode *DeepSphere hyperprior*, ce qui démontre l'intérêt de construire une convolution plus expressive (cf Figure 6). Les résultats visuels de la Figure 7 montrent clairement que la convolution proposée permet la reconstruction des détails par rapport à la convolution basée graphe.

Conclusion : Grâce à l'approche OSLO, il est possible de construire des architecture CNN opérant directement sur la sphère, tout en garantissant la même expressivité qu'une architecture 2D, et une distribution quasi-uniforme des pixels. Tout cela est mis en oeuvre dans une approche peu complexe ce qui permet d'envisager de nombreuses autres applications.

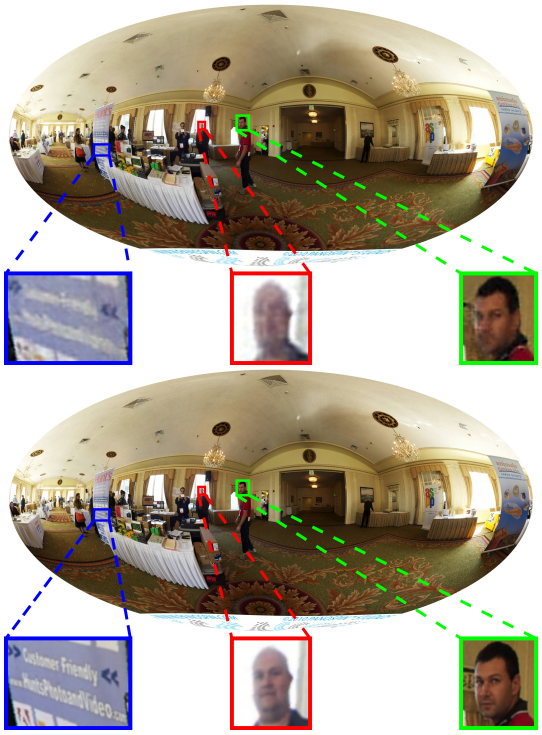

FIGURE 7 – Images décompréssées en visualisation Mollweide : en haut en utilisant la convolution basée graphe (2408 KB), en bas en utilisant notre convolution (938 KB).

Références

[1] Johannes Ballé, David Minnen, Saurabh Singh, Sung Jin Hwang, and Nick Johnston. Variational image compression with a scale hyperprior. In *International Conference on Learning Representations (ICLR)*, 2018.

[2] Taco S. Cohen, Mario Geiger, Jonas Koehler, and Max Welling. Spherical CNNs. In *International Conference on Learning Representations (ICLR)*, 2018.

[3] Carlos Esteves, Christine Allen-Blanchette, Ameesh Makadia, and Kostas Daniilidis. Learning SO(3) equivariant representations with spherical CNNs. In *Proceedings of the European Conference on Computer Vision (ECCV), 2018*.

[4] Carlos Esteves, Ameesh Makadia, and Kostas Daniilidis. Spin-weighted spherical CNNs. In *Advances in Neural Information Processing Systems (NIPS)*, 2020.

[5] Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. Neural message passing for quantum chemistry. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning (ICML)*, volume 70 of *Proceedings of Machine Learning Research*, pages 1263–1272. PMLR, 06–11 Aug 2017.

[6] K. M. Górski, E. Hivon, A. J. Banday, B. D. Wandelt, F. K. Hansen, M. Reinecke, and M. Bartelmann. HEALPix : A framework for high-resolution discretization and fast analysis of data distributed on the sphere. *The Astrophysical Journal*, 622(2) :759–771, apr 2005.

[7] Renata Khasanova and Pascal Frossard. Graph-based classification of omnidirectional images. In *Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCVW)*, pages 869–878, 2017.

[8] Renata Khasanova and Pascal Frossard. Geometry aware convolutional filters for omnidirectional images representation. In *International Conference on Machine Learning (ICML)*, pages 3351–3359. PMLR, 2019.

[9] Yeonkun Lee, Jaeseok Jeong, Jongseob Yun, Wonjune Cho, and Kuk-Jin Yoon. Spherephd : Applying CNNs on a spherical polyhedron representation of 360deg images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9181–9189, 2019.

[10] Navid Mahmoudian Bidgoli, Roberto G de A Azevedo, Thomas Maugey, Aline Roumy, and Pascal Frossard. Oslo : On-the-sphere learning for omnidirectional images and its application to 360-degree image compression. *arXiv e-prints*, pages arXiv–2107, 2021.

[11] Rafael Monroy, Sebastian Lutz, Tejo Chalasani, and Aljosa Smolic. Salnet360 : Saliency maps for omni-directional images with cnn. *Signal Processing : Image Communication*, 69 :26–34, 2018.

[12] N. Perraudin, M. Defferrard, T. Kacprzak, and R. Sgier. DeepSphere : Efficient spherical convolutional neural network with HEALPix sampling for cosmological applications. *Astronomy and Computing*, 27 :130–146, April 2019.

[13] Patrick J Roddy and Jason D McEwen. Sifting convolution on the sphere. *IEEE Signal Processing Letters*, 28 :304–308, 2021.

[14] Yu-Chuan Su and Kristen Grauman. Learning spherical convolution for fast features from 360° imagery. In *Advances in Neural Information Processing Systems (NIPS)*, 2017.

[15] Q. Yang, C. Li, W. Dai, J. Zou, G.-J. Qi, and H. Xiong. Rotation equivariant graph convolutional network for spherical image classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4302–4311, 2020. ISSN : 2575-7075.