

Towards Hyperbolic Regularizers For Point Cloud Part Segmentation

*Original*

Towards Hyperbolic Regularizers For Point Cloud Part Segmentation / Montanaro, Antonio; Valsesia, Diego; Magli, Enrico. - ELETTRONICO. - (2023), pp. 1-5. (Intervento presentato al convegno ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) tenutosi a Rhodes Island, Greece nel 04-10 June 2023) [10.1109/ICASSP49357.2023.10095253].

*Availability:*

This version is available at: 11583/2982650 since: 2023-10-02T10:04:27Z

*Publisher:*

IEEE

*Published*

DOI:10.1109/ICASSP49357.2023.10095253

*Terms of use:*

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

*Publisher copyright*

IEEE postprint/Author's Accepted Manuscript

©2023 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collecting works, for resale or lists, or reuse of any copyrighted component of this work in other works.

(Article begins on next page)

# TOWARDS HYPERBOLIC REGULARIZERS FOR POINT CLOUD PART SEGMENTATION

*Antonio Montanaro, Diego Valsesia, Enrico Magli*

Politecnico di Torino  
Department of Electronics and Telecommunications  
Turin, Italy

## ABSTRACT

Hyperbolic neural networks are emerging as an effective technique to better capture hierarchical representations of many data types, from text to images and, recently, point clouds. In this paper, we extend our earlier work, that showed how to use regularizers in the hyperbolic space to improve performance of point cloud classification models, to the problem of part segmentation. This requires careful modeling of the hierarchical relationships between parts and whole point cloud to properly control the hyperbolic geometry of the feature space produced by the neural network. We show how the proposed method improves the performance of commonly used neural network architectures, reaching state-of-the-art performance on the part segmentation task.

*Index Terms*— Point clouds, part segmentation, hyperbolic neural networks

## 1. INTRODUCTION

The compositional nature of 3D objects, for which the point cloud representation is one of the most commonly used, is evident in the fact that a whole object can be decomposed in progressively more elementary constituent parts. This induces a hierarchy where on one end simple universal shapes are found, whose combination leads to the other end where the complex geometries of individual objects are found. Mathematically, a tree can be used to represent such hierarchy where one can place a full object at root, whereas leaves are generated by progressively decomposing the object into the constituent parts.

Deep learning models for point cloud data have become increasingly powerful in the last years, exploiting sophisticated building blocks such as graph neural networks [1] to overcome the irregularity of the point cloud domain. However, despite some attempts, an effective exploitation of the aforementioned hierarchical nature of data has remained elusive. Indeed, works like PointGRL [2], DCGLR [3], Info3D [4] recognized the importance of capturing the interactions between local and global features and while they provide good performance in unsupervised problems, they fail to

model such dependencies in the supervised setting. In our recent work [5], we showed that this is due to the flawed geometry of the feature spaces adopted by these methods. Spaces that are either flat (Euclidean) or with positive curvature (spherical) are not able to embed tree structures representing hierarchies of features with low distortion. It is instead known that only spaces with negative curvature (hyperbolic) allow the exponential increase in volume as function of the radius that is necessary to represent trees. Our earlier work [5] focused on the point cloud classification problem, where it was shown that moving to the hyperbolic space and using regularizers to explicitly guide the geometry of the feature space to incorporate the part-whole hierarchy resulted in improved performance of any state-of-the-art neural network model for the problem.

In this paper, we address the point cloud part segmentation problem, expanding the previous work with a novel regularization framework called HyCoRe-seg (Hyperbolic Compositional Regularization for part segmentation). We will show that there are significant differences in the construction of effective hyperbolic regularizers between the whole point cloud classification problem and part segmentation. In fact, we will discuss how, somewhat counterintuitively, part segmentation requires one to enforce a part-whole hierarchy that is reversed with respect to the case of classification of whole objects. Moreover, we also introduce improvements to how parts are defined for purpose of regularization, showing that assembling them as clusters of points in a feature space increases effectiveness thanks to the exploitation of self-similarities.

Overall, our main contributions can be summarized as:

- we propose a novel regularizer for supervised training of point cloud part segmentation models using neural networks;
- this expands the previously known technique for classification, investigating some counterintuitive differences in the hierarchy design;
- we show that defining parts as neighborhoods in a feature space rather than in the original 3D space improves regularization effectiveness;

- we experimentally show improvements on widely-used neural networks for part segmentation.

## 2. BACKGROUND

### 2.1. Point cloud neural networks

Point clouds are unordered collections of positions in 3D space. This irregular domain makes them very challenging to process with neural networks which cannot exploit widely known primitives such as convolution. Recent years have seen progressive refinements of neural network architectures capable of working on the raw 3D coordinates, starting from PointNet [6], which processed each point independently before aggregating them with a globally symmetric operation, and PointNet++ [7], which introduced localized operations. Graph neural networks like DGCNN [8] define a graph to capture relations among points and successfully exploited graph convolutional operations to extract localized and hierarchical features.

### 2.2. Hyperbolic neural networks

The hyperbolic space is a Riemannian manifold with constant negative curvature  $c < 0$  in the metric:

$$\mathbf{g}_R = (\lambda_x^c)^2 \mathbf{g}_E = \frac{2}{1 + c\|\mathbf{x}\|^2} \mathbf{g}_E,$$

where  $\lambda_x^c$  is the conformal factor and  $\mathbf{g}_E$  is the metric tensor of Euclidean space  $\mathbb{R}^n$ , i.e.,  $\mathbf{I}_n$ . The Poincarè Ball in  $n$  dimensions  $\mathbb{D}^n$  is a hyperbolic space with  $c = -1$ . The distance is defined as:

$$d_{\mathbb{D}}(\mathbf{x}, \mathbf{y}) = \cosh^{-1} \left( 1 + 2 \frac{\|\mathbf{x} - \mathbf{y}\|^2}{(1 - \|\mathbf{x}\|^2)(1 - \|\mathbf{y}\|^2)} \right).$$

Since the Poincarè Ball is a Riemannian manifold, for each point  $\mathbf{x} \in \mathbb{D}^n$  we can define a logarithmic map  $\log_{\mathbf{x}} : \mathbb{D}^n \rightarrow T_{\mathbf{x}}\mathbb{D}^n$  that maps points from the Poincarè Ball to the corresponding tangent space  $T_{\mathbf{x}}\mathbb{D}^n \in \mathbb{R}^n$ , and an exponential map  $\exp_{\mathbf{x}} : T_{\mathbf{x}}\mathbb{D}^n \rightarrow \mathbb{D}^n$  that does the opposite.

The work by Sarkar [9] showed that trees can be embedded in the hyperbolic space with arbitrarily low distortions. For this reason, several works [10, 11] have sought to move representation learning to the hyperbolic space, especially for problems where clear hierarchies need to be captured. Hyperbolic counterparts [12] of classic Euclidean neural networks have been studied to combine their powerful processing capabilities with the desirable properties of the hyperbolic space. They have shown success in capturing hierarchical properties for task relating to language, images, and graphs. In the definition introduced by Ganea et al. [12] classic vector spaces are replaced by gyrovector spaces, where addition, scalar multiplication and matrix multiplication are redefined

via Mobius transformations as follows:

$$\mathbf{y} = \mathbf{M}^{\otimes c}(\mathbf{x}) = \frac{1}{\sqrt{c}} \tanh \left( \frac{\|\mathbf{M}\mathbf{x}\|}{\|\mathbf{x}\|} \tanh^{-1}(\sqrt{c}\|\mathbf{x}\|) \right) \frac{\mathbf{M}\mathbf{x}}{\|\mathbf{x}\|}$$

$$\mathbf{z} = \mathbf{y} \oplus_c \mathbf{b} = \exp_{\mathbf{y}}^c \left( \frac{\lambda_{\mathbf{y}}^c}{\lambda_{\mathbf{y}}^c} \log_0^c(\mathbf{b}) \right)$$

where  $\mathbf{M}$  and  $\mathbf{b}$  are the weight matrix and bias vector of a neural network layer. Pointwise non-linearities are applied via logarithmic and exponential maps:

$$\phi^{\otimes c}(\mathbf{z}) = \exp_{\mathbf{z}}^c(\phi(\log_0^c(\mathbf{z}))).$$

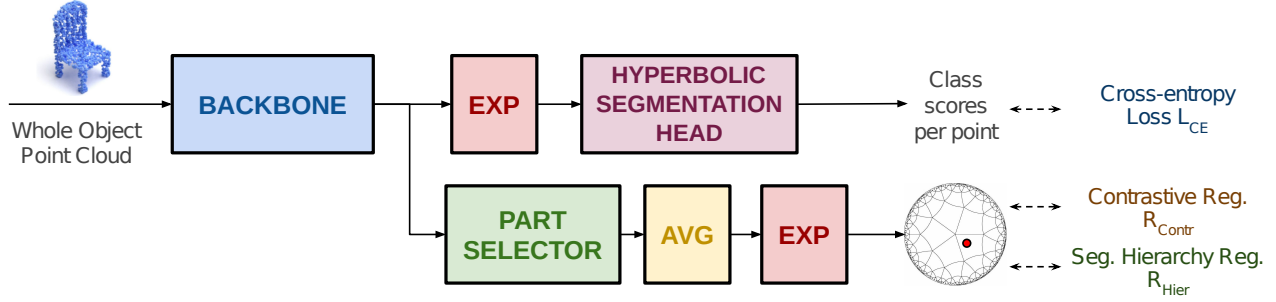
## 3. PROPOSED METHOD

An overview of the proposed method and how it fits a generic neural network architecture for the segmentation task is shown in Fig. 1. At a high level, the segmentation head of the neural network is replaced with a hyperbolic counterpart with Mobius layers. The segmentation loss is augmented by two regularizers which work on the embedding of the entire point cloud (feature vector average-pooled from all points) and of a part (feature vector average-pooled from a neighborhood of points, defined as  $k$ -NN in feature space) after mapping to the hyperbolic space. The contrastive regularizer promotes separation among embeddings of different classes, while the hierarchy regularizer promotes a part-whole hierarchy, placing whole objects closer to the origin of the Poincarè ball and smaller parts towards the edge. As mentioned, the geometry of this hyperbolic setting is very different from that employed in [5] for classification; the geometry is further explained in Sec. 3.1, whereas intuition about the obtained whole-part hierarchy is given in Sec. .

### 3.1. Hyperbolic feature space regularization

In this section, we present the proposed method to regularize neural networks for part segmentation of point clouds via the hyperbolic space. Let the input to the model be a point cloud  $P_N$  as a set of  $N$  3D points  $\mathbf{p} \in \mathbb{R}^3$ . We consider a generic encoder backbone providing a feature vector per point, i.e.,  $E : \mathbb{R}^{N \times 3} \rightarrow \mathbb{R}^{N \times F}$ . A segmentation head processes these features to derive class scores for each point. In HyCoRe-seg, this module is implemented with hyperbolic neural network layers. In particular, we use an exponential map to move from the Euclidean to hyperbolic space and then use one or more Mobius layers shared across points to reduce the dimensionality from  $F$  to the desired number of classes.

We seek to regularize the feature space produced by the encoder and ultimately used to estimate the segmentation labels so that it leverages prior knowledge about the existence of a hierarchy between parts and whole objects. In particular, we assume that this hierarchy can be described by a tree whose root is a whole object and leaves are small constituent parts. Each level of the hierarchy decomposes the object into



**Fig. 1.** HyCoRe-seg architecture. A state-of-art network encodes a point cloud into a feature space with per-point feature vectors. A part is extracted as the  $k$  nearest neighbors of a random point in the feature space, its average feature vector is computed and moved to the hyperbolic space via Exponential map. Regularizers impose the desired part-whole hierarchy and correct clustering according to labels.

smaller and simpler parts. This tree can be embedded in the Poincaré ball by placing the embedding of a whole object close to the origin of the ball and the embedding of small parts close to the ball edge. This is motivated by the properties of the hyperbolic space. In fact, a geodesic (shortest path) between two points passes closer to the origin, emulating the fact that traversing a tree from leaf to leaf requires to pass closer to the root. With this objective in mind, we need to analytically define what a part is for the purpose of training. A straightforward definition would be a number of spatially neighboring points, or the subset defined by the available part labels. Notice, however, that our definition of hierarchy may be more general than the specific semantic part labels that are available, so it even makes sense to create parts, for the purpose of regularization, that do not necessarily follow either the labels or clear semantic concepts. In this paper, we introduce a different definition of part, based on finding the  $N'$  nearest neighbors of a random point in the feature space produced by the encoder  $E$ . This definition allows to capture “parts” in a more general sense and exploiting, possibly non-local, self-similarities thanks to the fact that neighbors in a feature space capture higher-level properties.

HyCoRe-seg combines the classic cross-entropy loss on the outputs of the hyperbolic segmentation head with two regularizers working on the embeddings of the parts and the whole point cloud. In particular, the overall loss function for training is as follows:

$$L = L_{CE} + \alpha R_{\text{contr}} + \beta R_{\text{hier}}. \quad (1)$$

The  $R_{\text{hier}}$  regularizer promotes the aforementioned part-whole hierarchy by means of a triplet cost:

$$R_{\text{hier}}(\mathbf{z}_{\text{whole}}, \mathbf{z}_{\text{part}}) = \max(0, \|\mathbf{z}_{\text{whole}}\|_{\mathbb{D}} - \|\mathbf{z}_{\text{part}}\|_{\mathbb{D}} + \gamma(N')).$$

The embedding of the whole point cloud  $\mathbf{z}_{\text{whole}}$  is obtained by averaging the features of all points in the Euclidean space, as produced by the encoder  $E$ , and then mapped to the hyperbolic space via exponential map, while the embedding of the

part  $\mathbf{z}_{\text{whole}}$  is obtained in the same way but only restricted to the points selected as a part. We remark that we experimentally verified that average pooling in the Euclidean domain seems superior to doing that in the hyperbolic space. This might be related to the difficulty in defining the hyperbolic average operation (Einstein midpoint), as it is not available in closed form for the Poincaré ball, and the typical approach of mapping to the Klein model seems to introduce undesirable approximations.

Additionally, the  $R_{\text{contr}}$  regularizer promotes correct clustering of parts and whole point clouds in the hyperbolic space. It is defined as follows:

$$R_{\text{contr}}(\mathbf{z}_{\text{whole}}^+, \mathbf{z}_{\text{part}}^+, \mathbf{z}_{\text{part}}^-) = \max(0, d_{\mathbb{D}}(\mathbf{z}_{\text{whole}}^+, \mathbf{z}_{\text{part}}^+) - d_{\mathbb{D}}(\mathbf{z}_{\text{whole}}^+, \mathbf{z}_{\text{part}}^-) + \delta(N'))$$

where  $\mathbf{z}_{\text{whole}}^+$  and  $\mathbf{z}_{\text{part}}^+$  are the hyperbolic representation of the whole and a part from the same point cloud, while  $\mathbf{z}_{\text{part}}^-$  is the embedding of a part of a different point cloud from a different class. In both regularizers, hyperparameters  $\gamma, \delta$  are functions of the number of points  $N'$ , computed as  $\gamma(N') = \delta(N') = 1024/N'$ .

### 3.2. Differences with respect to whole point cloud classification

In our earlier work [5], we showed that the hyperbolic framework is very effective for the problem of whole point cloud classification. However, we need to remark one fundamental, and possibly counterintuitive, difference when addressing the point cloud part segmentation task. The classification problem in [5] defines the desired compositional hierarchy by placing whole objects as leaves of a tree and smaller parts towards the root. This hierarchy captured the idea that whole objects are specializations of simpler universal parts, and traversing the hierarchy from object A to object B requires passing through common part ancestors. This is reflected in the hyperbolic embedding by placing whole objects

**Table 1.** Part segmentation results on ShapeNetPart dataset.

	Cls.	Inst.	mIOU	mIOU	aero	bag	cap	car	chair	ear	guitar	knife	lamp	lapt	moto	mug	pistol	rock	stake	table
PointNet	80.4	83.7	83.4	78.7	82.5	74.9	89.6	73.0	91.5	85.9	80.8	95.3	65.2	93.0	81.2	57.9	72.8	80.6		
PointNet++	81.9	85.1	82.4	79.0	87.7	77.3	90.8	71.8	91.0	85.9	83.7	95.3	71.6	94.1	81.3	58.7	76.4	82.6		
DGCNN	80.9	85.1	83.4	77.7	85.4	77.9	90.9	74.4	91.6	88.9	83.7	96.2	59.8	91.8	81.1	54.0	74.2	82.6		
DGCNN+Self-Recon. [13]	-	85.3	84.1	84.0	85.8	77.0	90.9	80.0	91.5	87.0	83.2	95.8	71.6	94.0	82.6	60.0	77.9	81.8		
<b>DGCNN+HyCoRe-seg</b>	<b>82.8</b>	<b>85.7</b>	84.8	86.5	86.7	79.1	91.4	78.6	91.8	87.9	84.0	95.9	63.0	94.4	83.0	59.9	75.1	82.9		

close to the edge of the Poincaré ball. Notice how this is exactly reversed with respect to the hierarchy designed for the segmentation task in this paper.

The reason why this description is needed to properly exploit the regularizing effect of the hierarchy is related to the geometry of the hyperbolic space as function of the task we want to solve. In the classification task, we want to separate as much as possible the clusters of embeddings of whole objects belonging to different classes. For part segmentation, we wish to do the same, but at a part level, rather than object level. This requirement is better satisfied if those embeddings are closer to edge of the Poincaré ball where space is expanding exponentially fast and more room is available to better separate clusters.

#### 4. EXPERIMENTAL RESULTS

In this section, we report the experimental results obtained by HyCoRe-seg for the part segmentation task. We use the widely-known ShapeNetPart dataset [14] composed of 16881 3D objects spread across 16 categories. We apply our method to the DGCNN architecture to observe how HyCoRe-seg can boost the performance of a recent widely-used model. The Euclidean segmentation head of the original model is replaced with a hyperbolic equivalent with a Mobius and softmax layer shared across points. The overall number of parameters is unchanged. We set  $\alpha = \beta = 0.01$ ,  $\gamma = 1000$  and  $\delta = 4$ . A part has  $N'$  points selected as nearest neighbors in the feature space of a random point.  $N'$  is chosen as a random number between 200 and 600. We train the models using Riemannian SGD optimization. Pytorch and the geopt library are used for hyperbolic operations. Models are trained on Nvidia Titan RTX GPUs.

Table 1 shows the results for part segmentation. It can be seen that regularization via HyCoRe-seg significantly improves the performance of the DGCNN baseline model. We can also see that HyCoRe-seg outperforms techniques such as self-reconstruction [13] aimed at capturing global-local hierarchies via pretraining strategies.

Table 2 shows that the combination of the two proposed regularizers, i.e., hierarchy and contrastive, is superior to us-

**Table 2.** Effectiveness of regularizers.

	Inst. mIOU
<b>DGCNN+HyCoRe-seg</b>	<b>85.7</b>
DGCNN+R <sub>contr</sub>	85.6
DGCNN+R <sub>hier</sub>	85.5

**Table 3.** Performance of DGCNN+HyCoRe-seg where the parts are defined as local neighborhood of a point in the feature space or in the input space.

	Inst. mIOU
<b>Parts in Feature Space</b>	<b>85.7</b>
Parts in Input Space	85.6

ing each individually. This can be attributed to a better penalization of degenerate feature space configurations promoted by the individual costs.

Finally, Table 3 shows the effect of defining parts as spatial neighborhoods rather than feature-space neighbors. It can be seen that exploiting similarities in the feature space leads to a more effective definition of parts.

#### 5. CONCLUSIONS

We showed how to use the hyperbolic space and hyperbolic neural network layers to improve the performance of state-of-the-art models for point cloud part segmentation. The proposed regularization framework allows to effectively capture the hierarchical nature of 3D parts thanks to the properties of the hyperbolic space. More work is needed to generalize the framework beyond objects to address tasks like scene segmentation.

#### 6. REFERENCES

- [1] Weijing Shi and Raj Rajkumar, “Point-gnn: Graph neural network for 3d object detection in a point cloud,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 1711–1719.

- [2] Yongming Rao, Jiwen Lu, and Jie Zhou, “Global-local bidirectional reasoning for unsupervised representation learning of 3d point clouds,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 5376–5385.
- [3] Kexue Fu, Peng Gao, Renrui Zhang, Hongsheng Li, Yu Qiao, and Manning Wang, “Distillation with contrast is all you need for self-supervised point cloud representation learning,” *arXiv preprint arXiv:2202.04241*, 2022.
- [4] Aditya Sanghi, “Info3d: Representation learning on 3d objects using mutual information maximization and contrastive learning,” in *European Conference on Computer Vision*. Springer, 2020, pp. 626–642.
- [5] Antonio Montanaro, Diego Valsesia, and Enrico Magli, “Rethinking the compositionality of point clouds through regularization in the hyperbolic space,” in *Advances in Neural Information Processing Systems*, 2022.
- [6] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas, “Pointnet: Deep learning on point sets for 3d classification and segmentation,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 652–660.
- [7] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas, “Pointnet++: Deep hierarchical feature learning on point sets in a metric space,” *Advances in neural information processing systems*, vol. 30, 2017.
- [8] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E Sarma, Michael M Bronstein, and Justin M Solomon, “Dynamic graph cnn for learning on point clouds,” *ACM Transactions on Graphics (TOG)*, vol. 38, no. 5, pp. 146, 2019.
- [9] Rik Sarkar, “Low distortion delaunay embedding of trees in hyperbolic plane,” in *International Symposium on Graph Drawing*. Springer, 2011, pp. 355–366.
- [10] Maximillian Nickel and Douwe Kiela, “Poincaré embeddings for learning hierarchical representations,” *Advances in neural information processing systems*, vol. 30, 2017.
- [11] Gary Bécigneul and Octavian-Eugen Ganea, “Riemannian Adaptive Optimization Methods,” in *International Conference on Learning Representations*, 2019.
- [12] Octavian Ganea, Gary Bécigneul, and Thomas Hofmann, “Hyperbolic neural networks,” *Advances in neural information processing systems*, vol. 31, 2018.
- [13] Jonathan Sauder and Bjarne Sievers, “Self-supervised deep learning on point clouds by reconstructing space,” *Advances in Neural Information Processing Systems*, vol. 32, 2019.
- [14] Li Yi, Vladimir G Kim, Duygu Ceylan, I-Chao Shen, Mengyan Yan, Hao Su, Cewu Lu, Qixing Huang, Alla Sheffer, and Leonidas Guibas, “A scalable active framework for region annotation in 3d shape collections,” *ACM Transactions on Graphics (ToG)*, vol. 35, no. 6, pp. 1–12, 2016.