

# Hyperbolic vs Euclidean Embeddings in Few-Shot Learning: Two Sides of the Same Coin

Gabriel Moreira  
 Language Technologies Institute  
 Carnegie Mellon University  
 gmoreira@cs.cmu.edu

João Paulo Costeira  
 Institute for Systems and Robotics  
 Instituto Superior Técnico  
 jpc@isr.tecnico.ulisboa.pt

Manuel Marques  
 Institute for Systems and Robotics  
 Instituto Superior Técnico  
 manuel@isr.tecnico.ulisboa.pt

Alexander Hauptmann  
 Language Technologies Institute  
 Carnegie Mellon University  
 alex@cs.cmu.edu

## Abstract

Recent research in representation learning has shown that hierarchical data lends itself to low-dimensional and highly informative representations in hyperbolic space. However, even if hyperbolic embeddings have gathered attention in image recognition, their optimization is prone to numerical hurdles. Further, it remains unclear which applications stand to benefit the most from the implicit bias imposed by hyperbolicity, when compared to traditional Euclidean features. In this paper, we focus on prototypical hyperbolic neural networks. In particular, the tendency of hyperbolic embeddings to converge to the boundary of the Poincaré ball in high dimensions and the effect this has on few-shot classification. We show that the best few-shot results are attained for hyperbolic embeddings at a common hyperbolic radius. In contrast to prior benchmark results, we demonstrate that better performance can be achieved by a fixed-radius encoder equipped with the Euclidean metric, regardless of the embedding dimension.

## 1. Introduction

Hyperbolic embeddings first appeared in machine learning as an inductive bias for representing data assumed to have a *latent hierarchical structure* [3, 5, 20], outperforming Euclidean-based approaches in community detection, link prediction and word embedding benchmarks [7]. Not long after, they were enthusiastically adopted by the computer vision community, finding applications in classification [12], zero-shot/few-shot learning [13, 16], contrastive learning [9, 15, 28], segmentation [10] and VAEs [17, 19, 24]. The theoretical motivation underlying their use stems from

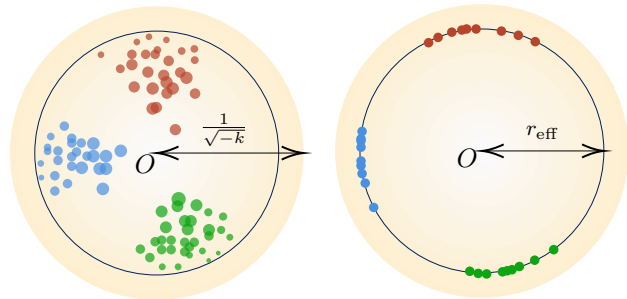


Figure 1. Hyperbolic image embeddings in the Poincaré ball: expectation (left) versus reality (right). In high-dimensional hyperbolic space, the volume of a ball is concentrated near its surface where the hyperbolic metric varies monotonically with the angle. Thus, the hierarchy-revealing property of hyperbolic space is lost.

the fact that hyperbolic geometry can theoretically capture complex hierarchical features that are otherwise elusive in low-dimensional Euclidean space [3, 20, 21].

In spite of the promising empirical results reported in the computer vision literature thus far, ascertaining which tasks can benefit the most from hyperbolic geometry is not yet clear. Up until recently, even classical cross-entropy based image classification in the Poincaré ball model of hyperbolic space was hardly on par with its Euclidean counterpart, much less competing with successful non-computer vision applications. Recent research [12] has linked this performance gap to embeddings converging towards the boundary of the ball. There, the Riemannian gradient vanishes, which effectively brings the optimization to a halt. It is known that the hyperboloid model outperforms the Poincaré ball partly because it circumvents many such numerical issues when learning low-dimensional word embed-

dings [21]. In image recognition applications, where representations are high-dimensional, a possible fix is to clip the embeddings' magnitude [12], which prevents the gradient underflow. Despite producing better classifiers, this technique does not prevent the embeddings from saturating all at the boundary of the smaller 'clipped' ball. Such a structural fact was obscured by performance indicators.

Our insight is that hyperbolic prototypical networks with a high-dimensional output space, typically in the order of  $10^3$  in few-shot learning, tend to produce embeddings with the largest possible magnitude, as illustrated in Fig. 1. In other words, instead of populating the entire space of the ball as expected, we observe that top performers place the visual embeddings near the border. In fact, the prototypical loss [25] is unbounded below.

We show that the hyperbolic measure of a high-dimensional ball is concentrated at its boundary, similarly to what happens in Euclidean space. This is corroborated by our experiments in popular few-shot datasets, which demonstrate that, regardless of magnitude clipping, in high-dimensional hyperbolic space, image embeddings lie at the maximum radius allowed. Thus, even if we employ a hyperbolic metric, the representations are restricted to a sphere where the distance between points varies monotonically with the angle between them. Such a scenario can be mirrored in Euclidean space since, in this case, there is no evidence of the property that sets hyperbolic space apart: its ability to encode hierarchical information in low dimensions.

In summary, we make a threefold contribution: 1) We show that, in high-dimensional hyperbolic space, the volume of a ball is concentrated at its boundary. This phenomenon, well established in Euclidean space, is often linked to the *curse of dimensionality*; 2) We demonstrate empirically that the boundary saturation is actually the configuration wherein the current best hyperbolic few-shot results are attained; 3) Since the boundary of the space where embeddings lie is a hyperbolic sphere, we show that a fixed-radius Euclidean encoder is able to match and even improve the hyperbolic classification accuracy without any of the intricacies of Riemannian optimization.

Prior benchmarks did not perform such a comparison, leading to the belief that hyperbolic embeddings were better suited towards few-shot recognition than traditional Euclidean features, by a considerable margin. Our findings present a stark contrast with previous Euclidean versus hyperbolic benchmarks [12, 13], where a considerable difference between the two geometries was observed.

Intuitively, our results suggest that individual images are simply instances of visual semantic relations and do not convey higher level concepts represented by a semantic hierarchy. In fact, while thinking of a concept, say a bird, we have great difficulty in finding one single image that con-

stitutes a 'parent' of the whole class. In this view, the best classifier should represent all images as leaves of a tree, in other words, at the same radius in the Poincaré ball.

## 2. Related work

The use of hyperbolic space in machine learning can be traced back to earlier works [2, 14] that set forth the hypothesis that complex networks may be thought of as discrete observations of a latent smooth metric space. For data points which are semantically meaningful and thus, amenable to being hierarchically categorized, hyperbolic geometry arises naturally as a consequence of node similarity distances. This duality between hierarchies, or more generally trees, and hyperbolic geometry was also shown by Sarkar [23] who proposed a combinatorial algorithm for embedding any tree in the Poincaré ball, while preserving the finite graph metric.

**Symbolic data** As a means to circumvent the limitations of Euclidean space in capturing complex symbolic patterns, Chamberlain *et al.* [3] described a hyperbolic version of the celebrated word2vec architecture [18] in the Poincaré ball model. The authors showed that community detection and node attribute prediction in low-dimensional hyperbolic space outperforms methods based on higher-dimensional Euclidean embeddings. Nickel and Kiela [20], adopting a similar model, reported similar findings for WordNet reconstruction and link prediction tasks. Unlike the former approach however, the authors derived a stochastic Riemannian optimization method. Subsequent work [21], motivated by the numerical hurdles presented by optimization in the Poincaré ball, proposed supplanting it for the Lorentz model, also known as the hyperboloid model. It was through the use of this model that Sala *et al.* [22] derived a matrix factorization-based hyperbolic multidimensional scaling algorithm (MDS) and Chami *et al.* designed hyperbolic graph convolutions [5].

**Image recognition** In image recognition, hyperbolic representation spaces have been employed with different end goals. One of them being the case wherein a hierarchy is established before-hand and the desiderata is for the representation space to be structured accordingly. Such was the basis for the zero-shot architecture proposed by Liu *et al.* [16], where WordNet was used as prior hierarchical knowledge of ImageNet, and its embeddings interpreted as class prototypes. Conversely, actual hyperbolic prototypical learning approaches have been set forth, such as ideal hyperbolic prototypes, proposed by Atigh *et al.* [1], making use of Busemann functions. More closely related to the original prototypical learning paper [25] is the work by Khrulkov *et al.* [13], where the authors retain the entire architecture,

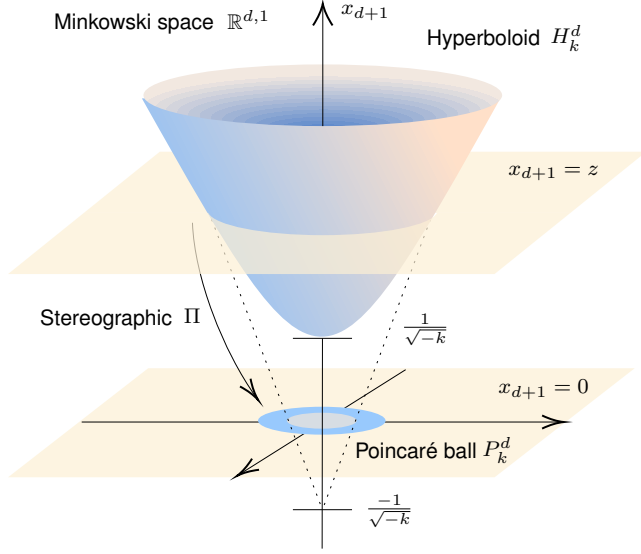


Figure 2. Minkowski ambient space  $\mathbb{R}^{d,1}$ , hyperboloid  $H_k^d$ , stereographic projection  $\Pi$  and Poincaré ball model  $P_k^d$ .

replacing simply the Euclidean metric by the Poincaré one, which yielded superior results. Advances have also been made in classical image classification. Based on empirical evidence suggesting that hyperbolic networks were not on par with their Euclidean counterparts, Guo *et al.* [12] attributed this to gradient underflow arising from the inverse Riemannian metric tensor and proposed clipping the magnitude of the Euclidean features before these are projected to the hyperbolic manifold. Finally, hyperbolic space has also been applied to image segmentation [10], VAEs [17, 19, 24] and contrastive learning [9, 15, 28].

The remainder of this paper is organized as follows. In Section 3 we lay out a brief summary of hyperbolic geometry, namely the hyperboloid and Poincaré ball models. In Section 4 we put forward the reasoning behind why we should expect hyperbolic image encoders to behave similarly to fixed-radius Euclidean ones. Finally, we present empirical results to validate our approaches in Section 5 and draw concluding remarks subsequently.

### 3. Hyperbolic space

Much like Euclidean space, hyperbolic space is isotropic and completely defined by its constant curvature, which unlike the former is negative everywhere. Several isomorphic models of hyperbolic space exist and in the machine learning literature, the most commonly adopted are the hyperboloid, also known as the Lorentz model [5, 7, 21], and the Poincaré ball [1, 3, 12, 13, 20], obtained from a stereographic projection of the former.

Throughout this paper we will be working with the Poincaré ball model of hyperbolic space, which can be de-

rived from the hyperboloid model as follows. Consider the Minkowski space  $\mathbb{R}^{d,1} := \{\mathbf{x} = (x_1, \dots, x_{d+1}) \in \mathbb{R}^d \times \mathbb{R}\}$ , together with the bilinear form (Lorentz pseudometric)

$$\langle \mathbf{x}, \mathbf{y} \rangle_L := \sum_{i=1}^d x_i y_i - x_{d+1} y_{d+1}. \quad (1)$$

The form  $\langle \mathbf{x}, \mathbf{x} \rangle_L$  is known as a Lorentz quadratic form. While not positive-definite everywhere in  $\mathbb{R}^{d,1}$  and thus, not a metric, it is positive-definite when restricted to the upper sheet of the  $d$ -hyperboloid

$$H_k^d := \left\{ \mathbf{x} \in \mathbb{R}^{d,1} \mid \langle \mathbf{x}, \mathbf{x} \rangle_L = \frac{1}{k}, x_{d+1} > 0 \right\}, \quad (2)$$

with curvature  $k < 0$ . We can write  $H_k^d$  in local coordinates  $u_1, \dots, u_d$  via the inclusion map  $\phi : H_k^d \rightarrow \mathbb{R}^{d,1}$

$$\phi(\mathbf{u}) := \left( u_1, \dots, u_d, \sqrt{\|\mathbf{u}\|_2^2 - \frac{1}{k}} \right). \quad (3)$$

**Poincaré model** The Poincaré ball model with sectional curvature  $k < 0$ , denoted as  $P_k^d$ , is obtained via the stereographic projection  $\Pi : H_k^d \rightarrow P_k^d$  of the  $d$ -hyperboloid through the origin (see Figure 2)

$$\Pi(\mathbf{x}) := \left( \frac{x_1}{1 + \sqrt{-k}x_{d+1}}, \dots, \frac{x_d}{1 + \sqrt{-k}x_{d+1}} \right). \quad (4)$$

This yields the ball

$$B_k^d = \Pi(H_k^d) = \left\{ x \in \mathbb{R}^d : \|\mathbf{x}\|_2^2 < -\frac{1}{k} \right\}, \quad (5)$$

where the boundary corresponds to infinity. The metric tensor in the canonical basis of  $\mathbb{R}^d$  is  $\lambda(\mathbf{u})^2 \mathbf{I}_d$ , with  $\lambda(\mathbf{u}) = 2/(1 + k\|\mathbf{u}\|_2^2)$ . The inverse projection  $\Pi^{-1} : P_k^d \rightarrow H_k^d$  takes the form

$$\Pi^{-1}(\mathbf{u}) = \left( \lambda(\mathbf{u})\mathbf{u}, \frac{1}{\sqrt{-k}}(\lambda(\mathbf{u}) - 1) \right). \quad (6)$$

Since  $\lambda(\mathbf{u}) \rightarrow \infty$  as  $\|\mathbf{u}\|_2 \rightarrow 1/\sqrt{-k}$ , for numerical reasons the effective radius of the ball is usually capped [8, 13] *i.e.* for  $\epsilon > 0$ ,

$$r_{\text{eff}} := \frac{1 - \epsilon}{\sqrt{-k}}. \quad (7)$$

The Poincaré exponential map at the origin  $\text{Exp}_0(\mathbf{v}) : TP_k^d \rightarrow P_k^d$  allows us to project a tangent vector back to the ball. This map is the element responsible for converting an Euclidean neural network to a hyperbolic one and reads

$$\text{Exp}_0(\mathbf{v}) = \tanh\left(\sqrt{-k}\|\mathbf{v}\|_2\right) \frac{\mathbf{v}}{\sqrt{-k}\|\mathbf{v}\|_2}. \quad (8)$$

Finally, the Poincaré geodesic (shortest-path) distance between any  $\mathbf{x}$  and  $\mathbf{y} \in P_k^d$  is given by

$$d_{P_k^d}(\mathbf{x}, \mathbf{y}) := \frac{2}{\sqrt{-k}} \operatorname{atanh} \left( \sqrt{-k} \|\mathbf{x} \oplus \mathbf{y}\|_2 \right), \quad (9)$$

where  $\oplus$  denotes the Möbius addition,

$$\mathbf{x} \oplus \mathbf{y} = \frac{(1 - 2k\langle \mathbf{x}, \mathbf{y} \rangle - k\|\mathbf{y}\|_2^2)\mathbf{x} + (1 + k\|\mathbf{x}\|_2^2)\mathbf{y}}{1 - 2k\langle \mathbf{x}, \mathbf{y} \rangle + k^2\|\mathbf{x}\|_2^2\|\mathbf{y}\|_2^2}. \quad (10)$$

**Poincaré image encoder** Given an Euclidean backbone  $f$  with parameters  $\theta$ , we consider hyperbolic image encoders of the type

$$h(\mathbf{x}; \theta) = \operatorname{Exp}_0^P(f(\mathbf{x}; \theta)). \quad (11)$$

The clipping strategy put forward in [12] consists of setting a maximum magnitude  $c$  for  $f(\mathbf{x}; \theta)$ , restricting the Poincaré ball to a ball with effective radius  $r_{\text{eff}} = \tanh(\sqrt{-k}c)/\sqrt{-k}$ . During training, for a suitable loss function  $L$ , and letting  $\mathbf{z} := h(\mathbf{x}; \theta) \in P_k^d$ , the Euclidean gradient  $\nabla_{\mathbf{z}} L$  may be backpropagated as is (see implementation by [13]), or it may be converted to a Riemannian gradient via scaling by the inverse metric tensor  $\operatorname{grad}_{\mathbf{z}} L = \lambda(\mathbf{z})^{-2} \nabla_{\mathbf{z}} L(\mathbf{z})$ , as in [12].

#### 4. Few-shot in the boundary

It is known that in hyperbolic neural networks, embeddings are prone to converge to the boundary of  $P_k^d$  (in practice, the effective radius  $r_{\text{eff}}$ ). We focus on the fact that, from a measure concentration argument, at the dimensions typically employed in few-shot learning, hyperbolic embeddings should concentrate at the boundary, regardless of whether their magnitude is clipped.

**Hyperbolic prototypical learning** We assume an image dataset  $\mathcal{I}$  with  $C$  semantic classes. If  $\mathcal{M}$  is a manifold and  $f_{\theta} : \mathcal{I} \rightarrow \mathcal{M}$  an image encoder parameterized by  $\theta$ , a typical prototypical learning approach [25] models the probability of  $\mathbf{z}_i$  being of class  $c$  as

$$p(c|\mathbf{z}_i) = \frac{\exp(-d_{\mathcal{M}}(\mathbf{w}_c, f(\mathbf{z}_i; \theta)))}{\sum_k \exp(-d_{\mathcal{M}}(\mathbf{w}_k, f(\mathbf{z}_i; \theta)))}, \quad (12)$$

where  $\mathbf{w}_c \in \mathcal{M}$  is the centroid of the  $c$ -th class. In classical prototypical networks, the squared Euclidean norm  $\ell_2^2$  is chosen as  $d_{\mathcal{M}}$  [25]. In Poincaré networks, the geodesic distance (9) is used instead. For simplicity, let  $\mathbf{x}_i = f(\mathbf{z}_i; \theta)$  and assume we are optimizing over  $\mathbf{x}$  instead of  $\theta$ . The negative log-likelihood associated with the  $i$ -th image is then

$$L_i = d_{\mathcal{M}}(\mathbf{w}_c, \mathbf{x}_i) + \log \sum_k e^{-d_{\mathcal{M}}(\mathbf{w}_k, \mathbf{x}_i)} \quad (13)$$

and the loss is  $L = \sum_i L_i$ . We can easily verify that  $L$  is unbounded below. Pick  $C$  directions in  $\mathbb{R}^d$ , one for each class centroid  $\mathbf{w}_c$ . Then set the direction of each embedding to match that of its class  $\mathbf{x}_i = r\mathbf{w}_c$  for a certain  $r > 0$ . The first term of (13) is automatically zero and the second goes to  $-\infty$  as the embeddings approach the boundary  $r \rightarrow 1/\sqrt{-k}$ . This justification could just as well be applied to Euclidean space, with the difference that we would require the output of the encoder to be unbounded. This is not necessary in the Poincaré model. In fact, the Poincaré exponential map at the origin (8) used to map points from the tangent space  $\mathbb{R}^d$  back to the manifold, is a scaling factor that allows us to approach the boundary at an exponential rate.

**High-dimensional hyperbolic space** In  $d$ -dimensional Euclidean space, the ratio of the volume over the area of a ball decreases with  $1/d$  i.e., in high-dimensional Euclidean space the volume is *concentrated* close to the surface. In hyperbolic space, the same thing happens.

**Proposition 1** (Hyperbolic measure concentration). *For large  $d$ , the volume of a hyperbolic ball is concentrated close to its boundary.*

*Proof.* From III.4 [6], the volume of a hyperbolic ball of (hyperbolic) radius  $r$  is

$$V_k(r) = \frac{2\pi^{d/2}}{\Gamma(d/2)} \int_0^r \left( \frac{\sinh(\sqrt{-k}t)}{\sqrt{-k}} \right)^{d-1} dt \quad (14)$$

and its area is given by

$$A_k(r) = \frac{2\pi^{d/2}}{\Gamma(d/2)} \left( \frac{\sinh(\sqrt{-k}r)}{\sqrt{-k}} \right)^{d-1}. \quad (15)$$

Hence,

$$\begin{aligned} \frac{V_k(r)}{A_k(r)} &= \left( \frac{\sinh(\sqrt{-k}r)}{\sqrt{-k}} \right)^{1-d} \int_0^r \left( \frac{\sinh(\sqrt{-k}t)}{\sqrt{-k}} \right)^{d-1} dt \\ &= \int_0^r \left( \frac{\sinh(\sqrt{-k}t)}{\sinh(\sqrt{-k}r)} \right)^{d-1} dt. \end{aligned} \quad (16)$$

For any  $d > 1$ , the function  $\frac{\sinh(\sqrt{-k}t)}{\sinh(\sqrt{-k}r)}$  is convex for  $t > 0$ , and thus, for  $t \in [0, r]$ , it is bounded above by  $t/r$ . From

$$0 \leq \left( \frac{\sinh(\sqrt{-k}t)}{\sinh(\sqrt{-k}r)} \right)^{d-1} \leq \frac{t^{d-1}}{r^{d-1}}, \quad (17)$$

the integral can be bounded above as

$$\frac{V_k(r)}{A_k(r)} \leq \int_0^r \frac{t^{d-1}}{r^{d-1}} dt = \frac{r}{d}. \quad (18)$$

Thus,  $\frac{V_k(r)}{A_k(r)} \rightarrow 0$  as  $d \rightarrow \infty$ .  $\square$

Note that this result holds for hyperbolic space in general, namely the Poincaré ball and the hyperboloid models commonly employed in machine learning. These arguments lead us to our hypothesis: given the high dimensionality of the Poincaré ball used in the hyperbolic few-shot literature, embeddings should lie at, or close to,  $r_{\text{eff}}$ . If this holds, it brings into question the hyperbolicity of the learnt representation space. In fact, a hyperbolic  $(d - 1)$ -sphere containing embeddings at  $r_{\text{eff}}$  from the origin is isometric to an Euclidean  $(d - 1)$ -sphere of radius

$$\frac{2/\sqrt{-k} + 2r_{\text{eff}}}{1/\sqrt{-k} - r_{\text{eff}}}. \quad (19)$$

This can be shown by attending to the fact that any  $(d - 1)$ -sphere in hyperbolic space with hyperbolic radius  $r$  is isometric to an Euclidean  $(d - 1)$ -sphere with radius  $\frac{1}{\sqrt{-k}} \sinh(r\sqrt{-k})$ . If embeddings lie at  $r_{\text{eff}}$  from the origin, their hyperbolic radius is

$$\frac{1}{\sqrt{-k}} \log \left( \frac{1/\sqrt{-k} + r_{\text{eff}}}{1/\sqrt{-k} - r_{\text{eff}}} \right) \quad (20)$$

and (19) follows. This isometry, in turn, implies that the representation space is not negatively curved.

**The Euclidean vs hyperbolic disparity** Assume our hypothesis holds and the embeddings all concentrate at  $r_{\text{eff}}$ . While the metric (9) employed in prior work [11, 13] is hyperbolic, it only encodes the angular distance. In fact, consider two embeddings  $\mathbf{x}, \mathbf{y}$  in  $P_k^d$  such that  $\|\mathbf{x}\|_2 = \|\mathbf{y}\|_2 = r_{\text{eff}}$  and  $\angle(\mathbf{x}, \mathbf{y}) = \alpha$ . The hyperbolic distance between them is

$$d_{P_k^d}(\mathbf{x}, \mathbf{y}) = \frac{1}{\sqrt{-k}} \text{acosh}(a + b \cos(\alpha)), \quad (21)$$

where  $b = \frac{4kr_{\text{eff}}^2}{(1+kr_{\text{eff}}^2)^2}$  and  $a = \left( \frac{1-kr_{\text{eff}}^2}{1+kr_{\text{eff}}^2} \right)^2$  (see supplemental material), which up to a multiplicative factor, depends only on  $\theta$  and  $kr_{\text{eff}}^2$ . A comparison between this metric, the  $\ell_2$  and the  $\ell_2^2$  is shown in Fig. 3. As we observe on the rightmost plot, the derivative of  $d_{P_k^d}$  at  $\alpha \rightarrow 0$  surges as we approach the boundary. As this happens, the penalty applied to the angular distance  $\alpha$  increases. Surprisingly, the Euclidean metric  $\ell_2$  (center plot) is not that different from the hyperbolic. In fact, it becomes a good approximation far from the boundary. In contrast, consider the squared Euclidean distance  $\ell_2^2$  depicted in the leftmost plot, often employed in prototypical networks since it defines a Bregman divergence [25]. Its derivative approaches 0 as  $\alpha \rightarrow 0$ .

Prior work conducted on hyperbolic prototypical learning benchmarked Euclidean space via  $\ell_2^2$ . Based on the previous paragraph, we propose instead fixed-radius Euclidean embeddings with the  $\ell_2 \propto \sqrt{1 - \cos(\alpha)}$  metric. Given a

radius hyperparameter  $r = 1/\sqrt{k}$  ( $k > 0$  is a spherical curvature), and the Euclidean backbone  $f(\cdot; \theta)$ , the embeddings fed to the prototypical loss (13) are computed as

$$r \frac{f(\mathbf{x}; \theta)}{\|f(\mathbf{x}; \theta)\|_2}. \quad (22)$$

As our experiments show, this Euclidean architecture is ahead of the hyperbolic few-shot state-of-the-art in most of the experiments conducted.

## 5. Experiments

We conducted our empirical evaluation in the CUB-200-2011 [27] and MiniImageNet [26] few-shot recognition tasks, in a prototypical learning regime.

**Architecture** We used a 4-layer ConvNet as backbone, with variable output dimension equal to  $d$ . These embeddings were then projected to the Poincaré ball through the exponential map at the origin (8) and through magnitude rescaling in the Euclidean case (22). In the latter, the radius is a hyperparameter, similarly to the curvature in the former. In the 5-shot 5-way setting, Poincaré centroids are computed through the Einstein midpoint closed-form expression set forth in [13]. In the Euclidean and the fixed-radius case, the Euclidean centroid was used. In terms of preprocessing, instead of cropping a centered square patch and resizing it to  $84 \times 84$  during inference as done in [13], we padded the cropped bounding boxes in the case of CUB, and the original images in MiniImageNet, so as to obtain a square image and then resized them. All other data augmentations were the same and can be consulted in the supplemental material.

**Training** All encoders were trained for 200 episodes, where each episode consisted of 15 queries, each of which associated with  $s$ -shot and  $w$ -way images. Adam was used with an initial learning rate of  $10^{-3}$  on CUB and  $5 \times 10^{-3}$  on MiniImageNet. Similarly to prior work, a step LR scheduler was used (see supplemental material). Both on the CUB dataset and on MiniImageNet, we used the train-validation-test split from [13]. In the former, 100 classes were used for training, 50 for validation and 50 for testing, and the shot-way was the same for training and inference *i.e.*, 1-shot 5-way and 5-shot 5-way. In the latter, the 100 class split was 64-16-20. The 1-shot 5-way model was trained in a 1-shot 30-way regime, and the 5-shot 5-way model in a 1-shot 20-way setting.

**Results** The CUB\_200\_2011 results are shown in Tables 1 and 2. Those of MiniImageNet are presented in Tables 3 and 4. Given that hyperbolic space is often praised for its ability to encode information in few dimensions, for each



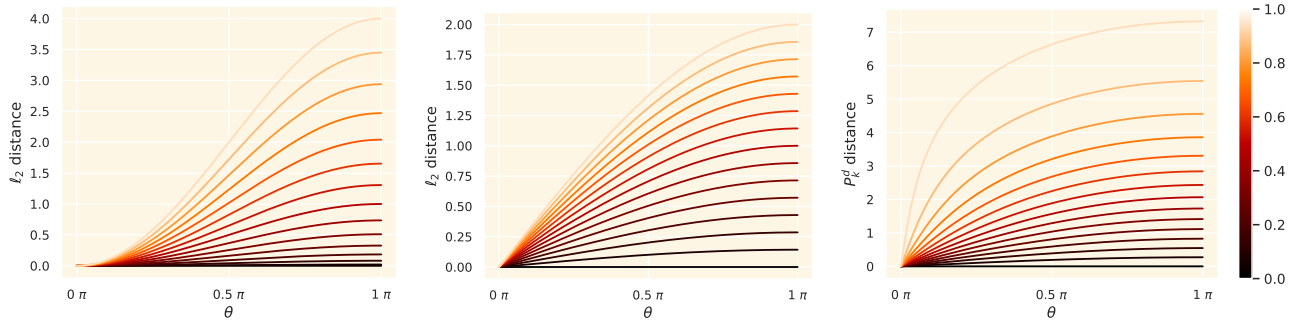


Figure 3. Comparison of metrics at different distances from the origin from 0 to  $r_{\text{eff}} = 0.99$  (color encodes fraction of  $r_{\text{eff}}$ ). Left: squared Euclidean distance ( $\ell_2^2$ ). Middle: Euclidean distance ( $\ell_2$ ). Right: Poincaré distance ( $d_{P^d_k}$ ) for  $k = -1$ .

dataset we present the test accuracy computed for four embedding dimensions  $d = 128, 256, 512$  and  $1024$ . For hyperbolic and fixed-radius architectures the minimum, average and maximum  $\ell_2$  embedding norms are reported as  $r_{\min}$ ,  $r_{\text{avg}}$  and  $r_{\max}$ , respectively. The traditional Euclidean architecture used in the literature as the Euclidean baseline (ProtoNet) is denoted as  $(\mathbb{R}^d, \ell_2^2)$ , the Poincaré ball with curvature  $k$  reads as  $P^d_k$  and the proposed fixed-radius Euclidean embeddings as  $(S^d, \ell_2)$ . The curvatures used in the latter were 0.006, 0.002 in 1s5w, 5s5w (CUB) and 0.002, 0.0002 in 1s5w, 5s5w (Mini), respectively.

As posited, when  $d$  increases the Poincaré embeddings approach the effective radius. In fact, plugging  $\epsilon = 0.01$  in (7) we obtain  $r_{\text{eff}} = 4.47$  for  $k = -0.05$ ,  $r_{\text{eff}} = 9.99$  for  $k = -0.01$  and  $r_{\text{eff}} = 14.13$  for  $k = -0.005$ . In the CUB dataset, the 5s5w training configuration avoids boundary saturation up to  $d = 512$ , however, the results are worse than the fixed-radius Euclidean architecture for the same dimensions. In MiniImageNet 5s5w, we observe embeddings not at the boundary for the largest dimension  $d = 1024$  considered. That being the case, the test accuracy is higher than that obtained in lower dimensions with embeddings farther from  $r_{\text{eff}}$ . Note that in the two datasets considered, we observed the same Euclidean versus hyperbolic ProtoNet few-shot accuracy gap reported in prior work [12, 13], especially in the 1s5w setting. However, the empirical advantage of hyperbolic representations no longer holds once normalized Euclidean embeddings together with the  $\ell_2$ -norm are introduced in the benchmark. In fact, even when Poincaré embeddings yield better results, the difference w.r.t fixed-radius embeddings is minor. Further, this is verified across all the dimensions considered, unlike e.g., graph or word embeddings where, in low dimensions, hyperbolic space offers a sizeable advantage.

Finally, results on the effect of magnitude clipping in high-dimensions for the CUB dataset are shown in Table 5. These experiments were performed for  $d = 1024$ ,  $k = -0.05$  and using Riemannian gradient as mentioned

$d$	Space	Test acc	$r_{\min}$	$r_{\text{avg}}$	$r_{\max}$
$2^7$	$(\mathbb{R}^d, \ell_2^2)$	$59.94 \pm 0.23$	-	-	-
	$P^d_{-0.05}$	$64.08 \pm 0.23$	4.47	4.47	4.47
	$P^d_{-0.01}$	$56.46 \pm 0.22$	9.98	9.99	9.99
	$(S^d, \ell_2)$	<b><math>66.13 \pm 0.23</math></b>	12.91	12.91	12.91
$2^8$	$(\mathbb{R}^d, \ell_2^2)$	$56.27 \pm 0.23$	-	-	-
	$P^d_{-0.05}$	$65.70 \pm 0.23$	4.47	4.47	4.47
	$P^d_{-0.01}$	$59.74 \pm 0.22$	9.99	9.99	9.99
	$(S^d, \ell_2)$	<b><math>66.47 \pm 0.23</math></b>	12.91	12.91	12.91
$2^9$	$(\mathbb{R}^d, \ell_2^2)$	$56.71 \pm 0.23$	-	-	-
	$P^d_{-0.05}$	$66.65 \pm 0.23$	4.47	4.47	4.47
	$P^d_{-0.01}$	$61.52 \pm 0.23$	9.99	9.99	9.99
	$(S^d, \ell_2)$	<b><math>67.90 \pm 0.23</math></b>	12.91	12.91	12.91
$2^{10}$	$(\mathbb{R}^d, \ell_2^2)$	$55.64 \pm 0.23$	-	-	-
	$P^d_{-0.05}$	$67.21 \pm 0.23$	4.47	4.47	4.47
	$P^d_{-0.01}$	$63.85 \pm 0.23$	9.99	9.99	9.99
	$(S^d, \ell_2)$	<b><math>67.92 \pm 0.23</math></b>	12.91	12.91	12.91

Table 1. CUB\_200\_2011 1-shot 5-way test results, 95% confidence intervals and embedding radii.

in the original paper. For each clipping value  $c$ , the corresponding effective ball radius can be obtained by plugging  $c$  in the exponential map (8) and taking the  $\ell_2$ -norm. Thus, for  $c = 2, 3$  and  $4$  we have 1.877, 2.618 and 3.191, respectively. We see the saturation effect occurring, especially for larger clipping values, where the test accuracy is coincidentally higher.

## 6. Discussion

As evidence by our results, in few-shot recognition, Euclidean embeddings do not perform worse than hyperbolic ones, as previously assumed. In fact, they can fare even better. This disparity with respect to the state-of-the-art can be

$d$	Space	Test acc	$r_{\min}$	$r_{\text{avg}}$	$r_{\max}$
$2^7$	$(\mathbb{R}^d, \ell_2^2)$	$78.95 \pm 0.16$	-	-	-
	$P_{-0.05}^d$	$79.17 \pm 0.17$	4.35	4.47	4.47
	$P_{-0.01}^d$	$82.30 \pm 0.15$	8.07	9.59	9.87
	$(S^d, \ell_2)$	<b><math>83.13 \pm 0.14</math></b>	22.36	22.36	22.36
$2^8$	$(\mathbb{R}^d, \ell_2^2)$	$80.14 \pm 0.16$	-	-	-
	$P_{-0.05}^d$	$80.58 \pm 0.16$	4.33	4.47	4.47
	$P_{-0.01}^d$	$84.51 \pm 0.14$	9.89	9.99	9.99
	$(S^d, \ell_2)$	<b><math>85.01 \pm 0.14</math></b>	22.36	22.36	22.36
$2^9$	$(\mathbb{R}^d, \ell_2^2)$	$79.95 \pm 0.15$	-	-	-
	$P_{-0.05}^d$	$81.04 \pm 0.16$	4.46	4.47	4.47
	$P_{-0.01}^d$	$84.60 \pm 0.14$	9.90	9.99	9.99
	$(S^d, \ell_2)$	<b><math>85.18 \pm 0.14</math></b>	22.36	22.36	22.36
$2^{10}$	$(\mathbb{R}^d, \ell_2^2)$	$78.83 \pm 0.15$	-	-	-
	$P_{-0.05}^d$	$81.06 \pm 0.16$	4.47	4.47	4.47
	$P_{-0.01}^d$	$84.70 \pm 0.14$	9.99	9.99	9.99
	$(S^d, \ell_2)$	<b><math>85.37 \pm 0.14</math></b>	22.36	22.36	22.36

Table 2. **CUB.200.2011 5-shot 5-way** test results, 95% confidence intervals and embedding radii.

$d$	Space	Test acc	$r_{\min}$	$r_{\text{avg}}$	$r_{\max}$
$2^7$	$(\mathbb{R}^d, \ell_2^2)$	$49.11 \pm 0.20$	-	-	-
	$P_{-0.005}^d$	$49.10 \pm 0.20$	3.78	5.85	7.41
	$P_{-0.01}^d$	$49.60 \pm 0.20$	2.78	3.53	4.15
	$(S^d, \ell_2)$	<b><math>50.24 \pm 0.20</math></b>	22.36	22.36	22.36
$2^8$	$(\mathbb{R}^d, \ell_2^2)$	$49.14 \pm 0.20$	-	-	-
	$P_{-0.005}^d$	$49.25 \pm 0.20$	8.77	9.93	10.44
	$P_{-0.01}^d$	$47.07 \pm 0.19$	7.54	8.05	9.43
	$(S^d, \ell_2)$	<b><math>50.36 \pm 0.20</math></b>	22.36	22.36	22.36
$2^9$	$(\mathbb{R}^d, \ell_2^2)$	$48.84 \pm 0.20$	-	-	-
	$P_{-0.005}^d$	$45.59 \pm 0.18$	14.12	14.13	14.13
	$P_{-0.01}^d$	$48.71 \pm 0.19$	9.98	9.99	9.99
	$(S^d, \ell_2)$	<b><math>50.97 \pm 0.19</math></b>	22.36	22.36	22.36
$2^{10}$	$(\mathbb{R}^d, \ell_2^2)$	$49.10 \pm 0.20$	-	-	-
	$P_{-0.005}^d$	$49.19 \pm 0.19$	14.13	14.13	14.13
	$P_{-0.01}^d$	<b><math>51.37 \pm 0.20</math></b>	9.99	9.99	9.99
	$(S^d, \ell_2)$	$51.36 \pm 0.20$	22.36	22.36	22.36

Table 3. **MiniImageNet 1-shot 5-way** test results, 95% confidence intervals and embedding radii.

attributed to the fact that the benchmarks conducted in previous papers considered Euclidean networks endowed with the squared  $\ell_2$ -norm as a distance function, as originally proposed by Snell *et al.* [25]. We, on the other hand, used the  $\ell_2$ -norm and normalized the embeddings’ magnitude,

$d$	Space	Test acc	$r_{\min}$	$r_{\text{avg}}$	$r_{\max}$
$2^7$	$(\mathbb{R}^d, \ell_2^2)$	$67.32 \pm 0.16$	-	-	-
	$P_{-0.005}^d$	$67.87 \pm 0.16$	12.40	13.42	13.69
	$(S^d, \ell_2)$	<b><math>68.57 \pm 0.16</math></b>	70.71	70.71	70.71
$2^8$	$(\mathbb{R}^d, \ell_2^2)$	$67.32 \pm 0.16$	-	-	-
	$P_{-0.005}^d$	<b><math>69.40 \pm 0.16</math></b>	13.03	13.78	13.93
	$(S^d, \ell_2)$	$69.10 \pm 0.16$	70.71	70.71	70.71
$2^9$	$(\mathbb{R}^d, \ell_2^2)$	$66.83 \pm 0.16$	-	-	-
	$P_{-0.005}^d$	<b><math>70.20 \pm 0.16</math></b>	14.06	14.13	14.13
	$(S^d, \ell_2)$	$69.95 \pm 0.16$	70.71	70.71	70.71
$2^{10}$	$(\mathbb{R}^d, \ell_2^2)$	$67.92 \pm 0.16$	-	-	-
	$P_{-0.005}^d$	<b><math>70.76 \pm 0.16</math></b>	14.06	14.13	14.13
	$(S^d, \ell_2)$	$70.48 \pm 0.16$	70.71	70.71	70.71

Table 4. **MiniImageNet 5-shot 5-way** test results, 95% confidence intervals and embedding radii.

Clipping value	Test acc	$r_{\min}$	$r_{\text{avg}}$	$r_{\max}$
2.0	$62.21 \pm 0.24$	1.873	1.877	1.877
3.0	$65.04 \pm 0.24$	2.618	2.618	2.618
4.0	$66.26 \pm 0.23$	3.191	3.191	3.191

Table 5. Effect of feature clipping before the exponential map. **CUB.200.2011 1-shot 5-way** test results, 95% confidence intervals and embedding radii in  $P_k^d$  for  $d = 1024$  and  $k = -0.05$ .

resulting in a chordal metric on a sphere.

In order to interpret these results in light of the success that hyperbolic embeddings have had in other areas, one must take into account that the dimensions required to have reasonable few-shot classification accuracies, (typically  $d > 10^3$  [12, 13]), are much higher than those used to embed words [3, 20], taxonomies or graphs [4, 5] in general. In addition, while in natural language one may argue that hyperbolicity arises naturally from the co-occurrence of symbols and hence the graph-like nature of the data, this property does not translate directly to the images. For instance, ImageNet is endowed with a taxonomy, inherited from the hypernymy and hyponymy relations in WordNet. The latter has been successfully embedded in low-dimensional hyperbolic space given its symbolic nature [16, 20, 21]. However, there may not be a unique visual representation of the abstract concepts in high-levels of this hierarchy. Only labels in leaf-nodes can be ascribed a unique image. In the case of balanced hierarchies or trees, from a symmetry standpoint, the embeddings of these leaf nodes are expected to lie at the same distance from the origin, akin to what is obtained for the image embeddings.

## 7. Conclusion

In this paper we set out to demonstrate that, contrary to widespread belief, hyperbolic embeddings are not necessarily better than Euclidean ones in few-shot learning. We showed that, in high-dimensional Poincaré balls, embeddings tend to saturate at the effective boundary. Consequently, the hierarchy-encoding property of hyperbolic space is lost and, in this situation, whatever accuracy hyperbolic few-shot classifiers have, can be ascribed to the angular separation of their representations. Based on this, we presented empirical results demonstrating that hyperbolic performance on two popular few-shot learning tasks can actually be matched and even improved upon via traditional Euclidean encoders with fixed-radius embeddings. We believe this work can open up new research directions on the design of more effective benchmarks for hyperbolic embeddings, as well as on the identification of computer vision applications wherein hyperbolic geometry presents an unambiguous advantage over Euclidean space.

**Acknowledgments** The authors would like to thank the reviewers for their comments and suggestions. This work was funded by Fundação para a Ciência e Tecnologia (FCT), within the scope of the CMU-Portugal Program [iFetch project - LISBOA-01-0247-FEDER-045920], LA/P/0083/2020 and UIDB/50009/2020. M Marques was supported by the SmartRetail project [PRR - C645440011-00000062], through IAPMEI - Agência para a Competitividade e Inovação.

## References

- [1] Mina Ghadimi Atigh, Martin Keller-Ressel, and Pascal Mettes. Hyperbolic Busemann Learning with Ideal Prototypes. In *Advances in Neural Information Processing Systems*, pages 103–115, 6 2021. [1](#), [2](#), [3](#)
- [2] Marian Boguna, Dmitri Krioukov, and kc claffy. Navigability of complex networks. *Nature Physics*, 5(1):74–80, 9 2009. [2](#)
- [3] Benjamin Paul Chamberlain, James Clough, and Marc Peter Deisenroth. Neural embeddings of graphs in hyperbolic space. Technical report, 5 2017. [1](#), [2](#), [3](#), [7](#)
- [4] Ines Chami, Albert Gu, Vaggos Chatziafratis, and Christopher Ré. From trees to continuous embeddings and back: hyperbolic hierarchical clustering. In *Advances in Neural Information Processing Systems*, pages 15065–15076, 2020. [7](#)
- [5] Ines Chami, Rex Ying, Christopher Ré, and Jure Leskovec. Hyperbolic Graph Convolutional Neural Networks. In *Advances in Neural Information Processing Systems*, 2019. [1](#), [2](#), [3](#), [7](#)
- [6] Isaac Chavel. *Riemannian Geometry: A Modern Introduction*. Cambridge University Press, 2006. [4](#)
- [7] Weize Chen, Xu Han, Yankai Lin, Hexu Zhao, Zhiyuan Liu, Peng Li, Maosong Sun, and Jie Zhou. Fully Hyperbolic Neural Networks. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, 5 2021. [1](#), [3](#)
- [8] Octavian-Eugen Ganea, Gary Bécigneul, and Thomas Hofmann. Hyperbolic Neural Networks. In *Advances in Neural Information Processing Systems*, 5 2018. [3](#)
- [9] Songwei Ge, Shlok Mishra, Simon Kornblith, Chun-Liang Li, and David Jacobs. Hyperbolic Contrastive Learning for Visual Representations beyond Objects. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6840–6849, 12 2022. [1](#), [3](#)
- [10] Mina GhadimiAtigh, Julian Schoep, Erman Acar, Nanne van Noord, and Pascal Mettes. Hyperbolic Image Segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4453–4462, 3 2022. [1](#), [3](#)
- [11] Albert Gu, Frederic Sala, Beliz Gunel, and Christopher Ré. Learning mixed-curvature representations in products of model spaces. In *International Conference on Learning Representations*, 2019. [5](#)
- [12] Yunhui Guo, Xudong Wang, Yubei Chen, and Stella X Yu. Clipped Hyperbolic Classifiers Are Super-Hyperbolic Classifiers. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. [1](#), [2](#), [3](#), [4](#), [6](#), [7](#)
- [13] Valentin Khrulkov, Leyla Mirvakhabova, Evgeniya Ustinova, Ivan Oseledets, and Victor Lempitsky. Hyperbolic Image Embeddings. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4 2019. [1](#), [2](#), [3](#), [4](#), [5](#), [6](#), [7](#)
- [14] Dmitri Krioukov, Fragkiskos Papadopoulos, Maksim Kitsak, Amin Vahdat, and Marian Boguna. Hyperbolic geometry of complex networks. *Physics Review E*, 82(3), 6 2010. [2](#)
- [15] Fangfei Lin, Bing Bai, Kun Bai, Yazhou Ren, Peng Zhao, and Zenglin Xu. Contrastive Multi-view Hyperbolic Hierarchical Clustering. In *arXiv:2205.02618*, 5 2022. [1](#), [3](#)
- [16] Shaoteng Liu, Jingjing Chen, Liangming Pan, Chong-Wah Ngo, Tat-Seng Chua, and Yu-Gang Jiang. Hyperbolic Visual Embedding Learning for Zero-Shot Recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020. [1](#), [2](#), [7](#)
- [17] Emile Mathieu, Charline Le Lan, Chris J. Maddison, Ryota Tomioka, and Yee Whye Teh. Continuous hierarchical representations with Poincaré variational auto-encoders. In *Advances in Neural Information Processing Systems*, 1 2019. [1](#), [3](#)
- [18] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed Representations of Words and Phrases and their Compositionality. In *Advances in Neural Information Processing Systems*, 2013. [2](#)
- [19] Yoshihiro Nagano, Shoichiro Yamaguchi, Yasuhiro Fujita, and Masanori Koyama. A Wrapped Normal Distribution on Hyperbolic Space for Gradient-Based Learning. In *International Conference on Machine Learning*, pages 4693–4702, 2 2019. [1](#), [3](#)
- [20] Maximilian Nickel and Douwe Kiela. Poincaré embeddings for learning hierarchical representations. In *Advances in Neural Information Processing Systems*, 5 2017. [1](#), [2](#), [3](#), [7](#)
- [21] Maximilian Nickel and Douwe Kiela. Learning continuous hierarchies in the Lorentz model of hyperbolic geometry.



- try. In *International Conference on Machine Learning*, pages 3779–3788, 2018. [1](#), [2](#), [3](#), [7](#)
- [22] Frederic Sala, Christopher De Sa, Albert Gu, and Christopher Ré. Representation tradeoffs for hyperbolic embeddings. In *International Conference on Machine Learning*, pages 4460–4469, 2018. [2](#)
- [23] Rik Sarkar. Low distortion Delaunay embedding of trees in hyperbolic plane. In *International Symposium on Graph Drawing*, pages 355–366, 2011. [2](#)
- [24] Ondrej Skopek, Octavian-Eugen Ganea, and Gary Bécigneul. Mixed-curvature variational autoencoders. In *International Conference on Learning Representations*, 2020. [1](#), [3](#)
- [25] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical Networks for Few-shot Learning. In *Advances in Neural Information Processing Systems*, 2017. [2](#), [4](#), [5](#), [7](#)
- [26] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching Networks for One Shot Learning. *Advances in Neural Information Processing Systems*, 29, 6 2016. [5](#)
- [27] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The Caltech-UCSD Birds-200-2011 Dataset, 2011. [5](#)
- [28] Yun Yue, Fangzhou Lin, Kazunori D Yamada, and Ziming Zhang. Hyperbolic Contrastive Learning. 2 2023. [1](#), [3](#)