HMSN: Hyperbolic Self-Supervised Learning by Clustering with Ideal Prototypes

Aiden Durrant   **Georgios Leontidis**
University of Aberdeen
Department of Computing Science & Interdisciplinary Centre for Data and AI
Aberdeen, United Kingdom
{a.durrant.20, georgios.leontidis}@abdn.ac.uk

Abstract

Hyperbolic manifolds for visual representation learning allow for effective learning of semantic class hierarchies by naturally embedding tree-like structures with low distortion within a low-dimensional representation space. The highly separable semantic class hierarchies produced by hyperbolic learning have shown to be powerful in low-shot tasks, however, their application in self-supervised learning is yet to be explored fully. In this work, we explore the use of hyperbolic representation space for self-supervised representation learning for prototype-based clustering approaches. First, we extend the Masked Siamese Networks to operate on the Poincaré ball model of hyperbolic space, secondly, we place prototypes on the ideal boundary of the Poincaré ball. Unlike previous methods we project to the hyperbolic space at the output of the encoder network and utilise a hyperbolic projection head to ensure that the representations used for downstream tasks remain hyperbolic. Empirically we demonstrate the ability of these methods to perform comparatively to Euclidean methods in lower dimensions for linear evaluation tasks, whilst showing improvements in extreme few-shot learning tasks.

1   Introduction

Self-supervised representation learning for natural images has continued to make vast progress in past years [21, 33, 42, 18, 8, 5], quickly approaching, and in cases with significant data preprocessing surpassing supervised learning performance [44]. The advantage of self-supervision lies in the ability to leverage the large quantities of data that exist in the world without human annotations to learn high-quality representations. However, most established self-supervised visual learning methods typically project representations on Euclidean or Hyperspherical manifolds, and in some cases disregarding the underlying hyperbolic structure of the data.

When attempting to capture hyperbolic data structures, zero and positive curvature spaces exhibit some inherent implications as opposed to negative curvature hyperbolic space, most notably the inability to embed hierarchical semantic relationships between points in space, a well-established principle of learning good representations [10]. Although there has been debate to what extent natural images exhibit underlying hyperbolic structure of semantics, recent works have demonstrated via empirical metrics the presence of latent hierarchical tree-like structures in standard computer vision datasets [35, 24], and subsequently shown the capabilities of hyperbolic representations to excel in these settings [50, 35, 24]. Many of these advancements in hyperbolic learning have been seen in metric and prototype learning settings, specifically for few-shot learning, where the highly separable semantic hierarchies lead to better-performing few-shot classifiers [29]. Self-supervision via prototype learning has also demonstrated state-of-the-art performance in few-shot and low-shot

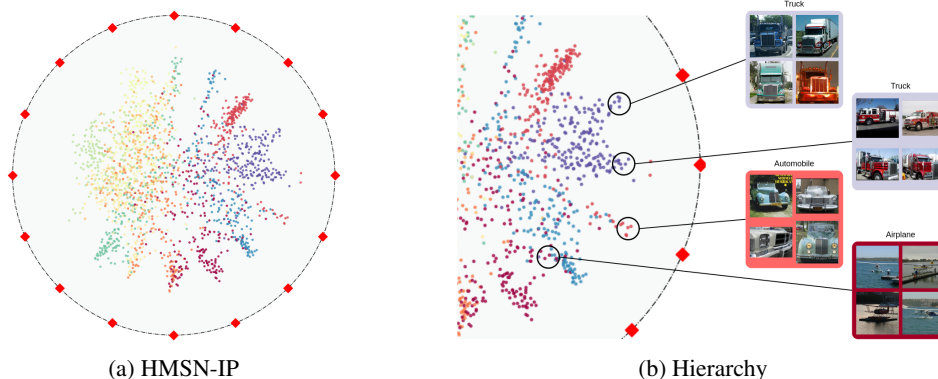|	
(a) HMSN-IP	(b) Hierarchy

Figure 1: **Depiction of the 2D embeddings of the STL-10 validation dataset.** The learnt embeddings of our proposed hyperbolic MSN with ideal prototypes. The red points represent the prototypes, the dotted line is the boundary of the Poincaré ball. For (a) natural semantic class clusters form at individual prototypes. Neighbouring prototypes capture similar semantic class sub-features (b), this is observed clearly with fire trucks being separately clustered to trucks (*purple points*), and large grilled cars (*light-red points*) being positioned in a similar manner. Seaplanes are positioned alongside boats (*blue points*) rather than airplanes (*dark red*) yet lie closer to the origin given there ambiguity.

tasks [3], therefore we leverage the hierarchical learning capabilities of hyperbolic prototype learning in the self-supervised setting for improved few-shot learning.

In this work, we propose the use of hyperbolic representation spaces in Self-Supervised Learning (SSL) to more appropriately embed the natural semantic class hierarchies presented in the data. We first demonstrate the capability of hyperbolic learning on a leading low-shot learning method of Masked Siamese Networks (MSNs) [3]. Here we project the output Euclidean representation onto the Poincaré ball and use the Poincaré distance in gyrovector space in place of the cosine similarity in the probability computation of the codes. We empirically show that such a conversion to hyperbolic space can lead to an improvement in representation quality for few-shot downstream tasks. Importantly and unlike previous methods [27, 52], we propose the use of fully hyperbolic projection networks, projecting the output of the encoder to hyperbolic space to ensure the hyperbolicity we aim to learn in the representations is utilised in downstream tasks.

In addition, we propose a new self-supervised method based on MSNs that leverages the advancements in hyperbolic prototype learning [28] where instead of continually learning prototypes, we place the prototypes on the ideal boundary of the Poincaré ball of hyperbolic space. We train our network to produce good hyperbolic representations through a new loss function based on the Busemann distance metric [28]. We empirically demonstrate improvements over the Euclidean baseline and our hyperbolic conversion on few-shot and extreme low-shot learning tasks. Furthermore, we show that our hyperbolic methods are competitive with other Euclidean methods through standard self-supervised linear evaluation and transfer learning benchmarks.

To summarise, the main contributions of the paper are the following:

- We propose a hyperbolic reformulation of the MSN clustering-based loss function, Hyperbolic Masked Siamese Networks (HMSN).
- We utilise ideal prototypes that lie on the ideal boundary of the Poincaré ball to encourage full utilisation of the space. We introduce Hyperbolic Masked Siamese Networks with Ideal Prototypes (HMSN-IP), based on the MSN method, employing the Busemann prototype loss from metric learning as a measure of distance between embeddings and the ideal prototypes.
- We propose to project Euclidean representations to the Poincaré ball of Hyperbolic space at the output of the encoder and present the use of hyperbolic projection heads as a solution to preserve hyperbolic structure in the output of the Euclidean encoder for downstream tasks.
- We empirically demonstrate that both our propositions outperform the Euclidean counterparts on few-shot and low-shot learning tasks in fewer embedding dimensions, whilst remaining competitive in linear evaluation tasks.

2 Related Work

Self-Supervised Representation Learning. In self-supervised learning, we consider a set of unlabelled images D which we aim to learn representations for use in downstream tasks. We pre-train on D and then adapt the representations via a supervised task using a set of images S and their corresponding labels where here $\|S\| << \|U\|$. The most successful methods to learn good representations employ view-invariant joint-embedding architectures [38, 47, 34, 6, 33, 37, 8, 5] which aim to predict the embedding of a view from another view of the same image. There exist a number of methods to train joint embedding predictive architectures, non-contrastive, which maximise the information content of the embeddings [30, 21, 8], and distillation, in which the outputs of one branch of the Siamese join embedding architecture act as a target for the other branch [15, 16, 56, 17, 51, 2]. The latter is the focus of this work, primarily the methodologies DINO [18] and its later derivation MSN [3], which utilise discrete cluster prototypes to quantise the output representations.

Clustering approaches have excelled with the use of vision transformers achieving near-to or state-of-the-art performance in most self-supervised benchmarks [17, 18, 3]. More recently there has been greater thought placed into masking strategies of these approaches with the aim to learn better representations through prediction or invariance to the missing regions [55, 48, 3, 32, 20]. These approaches are particularly of interest in this work as a result of their exceptional performance in low-shot and extreme low-shot training settings [3]. As such, given our aims to improve low-shot learning, we base our loss and hyperbolic representation space reformulations on the leading architectural designs, specifically MSN.

Hyperbolic Learning. The advocation for learning representations or embeddings in non-Euclidean space in deep learning has, in recent years, increased rapidly. Hyperbolic reformulations of deep learning layers across both intermediate [26] and classification layers [29, 35], as well as whole architectural propositions [41] have been proposed with improved performance and computational efficiency. Hyperbolic deep learning has seen great success in tasks where the representation of tree-like structures is beneficial like natural language [1, 43, 54] and graph neural networks [36, 19, 7]. The application of hyperbolic deep learning in vision is still however foundational, yet vast work has been undertaken in visual metric learning [11, 35, 24], with [50] performing hierarchical unsupervised similarity based metric learning, [24] extending the DINO [18] architecture with hyperbolic contrastive learning for metric learning. The latter which projects the output embedding space to hyperbolic space further motivates our decision to base our work on MSN given its known capabilities, albeit not in a self-supervised setting.

Moreover, hyperbolic metric learning and prototype learning approaches have demonstrated their capabilities in few-shot and zero-shot [25, 49, 28] learning tasks, outperforming Euclidean embedding methods by some margins. Given the connections between metric learning, and prototype learning to self-supervision, there exhibit clear enablers between the domains. The work [24] explores these, initially investigating hyperbolic self-supervised learning before re-evaluating it as a metric learning approach given improved performance in this domain. Contrastive self-supervision has also been addressed in [27, 52] which proposes a number of hyperbolic reformulations of prominent SSL and contrastive objectives. Our work aims to further explore the use of hyperbolic embedding space for self-supervised learning, advocating for its use to help provide greater insights and representation quality for all tasks while leveraging its strong performance in few-shot and low-shot learning.

3 Prerequisites

3.1 Hyperbolic Learning: The Poincaré Ball Model

Hyperbolic space $\mathbb{D}^d$ is the unique simply connected d-dimensional Riemannian manifold of constant negative curvature, where curvatures measure the deviation from flat Euclidean geometry. The constant negative curvature of the hyperbolic space, although analogous to the Euclidean sphere, presents some significant differences in geometric properties. As such hyperbolic space cannot be isometrically embedded into Euclidean space, yet there exist a number of conformal models of hyperbolic geometry [14] employing hyperbolic metrics providing a subset of Euclidean space. In this work, we employ the Poincaré ball model for hyperbolic geometry given its wide adoption in computer vision and unique properties ideal for embedding between euclidean and hyperbolic representations.

The Poincaré ball model $(\mathbb{D}_c^d, g^{\mathbb{D}_c})$ is defined by the manifold $\mathbb{D}_c^d = \{x \in \mathbb{R}^d : c\|x\|^2 < 1\}$ with the Riemannian metric

$$g^{\mathbb{D}_c} = (\lambda_x^c)^2 g^E = \left(\frac{2}{1 - c\|x\|^2}\right)^2 \mathbb{I}^d \tag{1}$$

where $g^E = \mathbb{I}^d$ is the Euclidean metric tensor and $\lambda_x^c = \frac{2}{1 - c\|x\|^2}$ is the conformal factor with c, a hyperparameter, controlling the curvature and radius of the ball. The conformal factor scales the local distances which approach infinity near the boundary of the ball, providing the unique property of space expansion. Such space expansion of hyperbolic spaces makes them continuous analogues of trees, given volumes of an object with diameter r scale exponentially with r. Thus, when referring to a tree with branching factor k, there are $\mathcal{O}(k^l)$ nodes at level l, where l serves as a discrete analogue of the radius. This is the fundamental property which the advocating work [26, 35, 50] and ours takes advantage of, allowing for the efficient embedding of natural hierarchies [40].

Our approach employs encoders that operate in Euclidean space, and as such, we need to define a bijection from Euclidean embeddings of the encoder to the Poincaré ball of hyperbolic space. To achieve this we apply an exponential map $\exp_v^c(x) : \mathbb{R}^d \to \mathbb{D}_c^d$ on Euclidean vector x with some fixed base point $v \in \mathbb{D}_c^d$ which we set v to be the origin, simplifying the exponential map and measures of distance which will be defined later. The exponential map is as follows,

$$\exp_v^c(x) = v \oplus_c \left(\tanh \left(\sqrt{c} \frac{\lambda_v^c \|x\|}{2} \right) \frac{x}{\sqrt{c}\|x\|} \right) \tag{2}$$

with its inverse logarithm map given by

$$\log_v^c(x) = \frac{2}{\sqrt{c}\lambda_v^c} \operatorname{arctanh} \left(\sqrt{c} \| - v \oplus_c x\| \right) \frac{-v \oplus_c x}{\| - v \oplus_c x\|}. \tag{3}$$

Given the change in geometry, hyperbolic spaces do not allow for standard vector space operations, as such we employ gyrovector formalism for standard operations such as addition, subtraction, multiplication [45, 26]. Therefore, from Eq. 2, $\oplus_c$ is defined as the gyrovector or Möbius addition of a pair of points $x, y \in \mathbb{D}_c^d$

$$v \oplus_c w = \frac{(1 + 2c\langle v, w \rangle + c\|w\|^2)v + (1 - c\|v\|^2)w}{1 + 2c\langle v, w \rangle + c^2\|v\|^2\|w\|^2}. \tag{4}$$

Leading from gyrovector formalism is the notion of distance, vital for self-supervised losses where typically the Euclidean cosine similarity and distance, are employed [39, 21, 8]. On the Poincaré ball of hyperbolic space we define the distance between $x, y \in \mathbb{D}_c^d$ as follows:

$$\operatorname{dist}_{\mathbb{D}}(x, y) = \frac{2}{\sqrt{c}} \operatorname{arctanh} \left(\sqrt{c} \| - x \oplus_c y\| \right), \tag{5}$$

which with $c = 1$ the geodesic is recovered, a vital concept given cosine similarity is analogous to sphere geodesic distance, whereas $c \to 0$ the Euclidean distance is produced.

3.2 Self-Supervised Learning: Masked Siamese Networks

In this work, we use the Masked Siamese Network (MSN) [3] as a base for our hyperbolic implementation due to its leading performance as a few-shot learner in self-supervision, its computational efficiency, and established clustering based loss formulation. This therefore provides us with the best opportunity for baseline comparison when striving for improved low-shot performance when employing hyperbolic representation space.

In MSN data augmentations are applied to image $\mathbf{x}_i$ to produce the target view $\mathbf{x}_i^+$ and a set of $M \geq 1$ anchor views $\mathbf{x}_{i,1}, \mathbf{x}_{i,2}, \ldots, \mathbf{x}_{i,M}$ where i is the index of the sample mini-batch of images $B \geq 1$. The anchor views $\mathbf{x}_{i,m}$ are subsequently patched into $N \times N$ non-overlapping regions and masked randomly or via a focal scheme [3] denoted by $\hat{\mathbf{x}}_{i,m}$. The encoders f_θ and $f_{\bar{\theta}}$ are identical although differently parameterised trunks of the ViT [23] outputting representations corresponding to the [CLS] token. The anchor views $\hat{\mathbf{x}}_{i,m}$ processed by the anchor encoder which is parameterised by θ produce the representations $z_{i,m} \in \mathbb{R}^d$ while the target views $\mathbf{x}_i^+$ processed by the target encoder

parameterised by $\bar{\theta}$ produce the representations $z_i^+ \in \mathbb{R}^d$. The target encoder is not directly updated by the optimisation process with gradients only computed with respect pot the anchor predictions, rather $\bar{\theta}$ are updated via an exponential moving average of the anchor encoder. Each encoder is trained with a 3-layer non-linear projection head $g_\theta(\cdot)$ and $g_{\bar{\theta}}(\cdot)$ with batch-normalisation at the input and hidden layers, which is later discarded during evaluation.

The metric which drives invariance between views is the soft distribution over a set of $K > 1$ learnable prototypes of dimension d denoted by $\mathbf{q} \in \mathbb{R}^{K \times d}$. The distribution is computed as the cosine similarity between the prototypes $\mathbf{q}$ and the L_2-normalized anchor and target views pairs where for the anchor view representation $z_{i,m}$ the prediction distribution $p_{i,m} \in \Delta_K$ is given by

$$p_{i,m} := \mathrm{softmax}\left(\frac{z_{i,m} \cdot \mathbf{q}}{\tau} \right). \tag{6}$$

The same formulation applies to the target view representations z_i^+ substituting the anchor views to produce target predictions $p_i^+ \in \Delta_K$. The temperature $\tau \in (0,1)$ is always chosen to be larger for the anchor predictions ($\tau^+ < \tau$) to encourage sharper target predictions producing confident low entropy anchor predictions which have been shown to provably discourage collapsing solutions [3].

The network is trained by the cross-entropy loss $H(p_i^+, p_{i,m})$ to penalise differing predictions of views that originate from the same image. This cross-entropy loss is regularised by mean entropy maximisation to encourage the use of the full set of prototypes, which maximises the entropy of the mean anchor predictions $H(\bar{p})$. The overall objective to be minimised when optimising over θ and q is given by Eq.7 where λ controls the weight of the mean entropy maximisation regularisation.

$$\frac{1}{MB} \sum_{i=1}^{B} \sum_{m=1}^{M} H(p_i^+, p_{i,m}) - \lambda H(\bar{p}) \tag{7}$$

For a more detailed description of MSN, we refer the reader to [3], and for implementation details we refer to the supplementary material.

4 Hyperbolic Masked Siamese Networks

Learning of hyperbolic embeddings under the MSN framework can most simply be achieved by mapping the euclidean output embeddings of the network to the Poincaré ball model, via Eq.2 followed by the substitution of the euclidean vector operations of the objective function Eq.7 for hyperbolic gyrovector equivalents. This approach to hyperbolic reformulation has shown to be an effective method of learning hyperbolic visual representations [24] and in contrastive self-supervision [27, 52]. We therefore first follow this methodology to examine the capabilities of hyperbolic self-supervision, we refer to the reformulation as Hyperbolic Masked Siamese Network (HMSN). We begin by projecting the anchor and target representations to the Poincaré ball model by the exponential map Eq.2, and initialising the prototypes $\mathbf{q}$ normally on the same hyperbolic space. The standard euclidean cosine similarity in Eq. 6 to compute the prediction metric $p_{i,m}$, is substituted for the geodesic distance of Eq.5. The reformulation of Eq. 6 results in the following prediction:

$$p_{i,m}^{\mathbb{D}} := \mathrm{softmax}\left(\frac{\mathrm{dist}_{\mathbb{D}}(z_{i,m}, \mathbf{q})}{\tau} \right). \tag{8}$$

The overall objective function in Eq.7 remains identical substituting $p_{i,m}$ for $p_{i,m}^{\mathbb{D}}$ although is trained by Riemannian Adam [9] as we are directly optimising jointly the prototypes in hyperbolic space and the euclidean parameters θ. We initialise the prototypes to be normal with a small standard deviation (0.01) centred around the origin for improved stability early in training. We also clip the Euclidean representations before projection to the Poincaré ball model as in [31] to assist in vanishing gradients when backpropagating from the hyperbolic space to the euclidean space as embeddings tend towards the boundary of hyperbolic space during training.
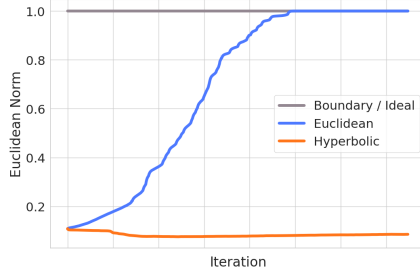
Figure 2: **Distance of prototypes from origin during training.** Mean Euclidean norm of Prototypes during training of Euclidean MSN baseline (*blue*), Hyperbolic MSN (*orange*), and Hyperbolic MSN with ideal prototypes (*grey*).

(a) 1 Epochs

(b) 5 Epochs

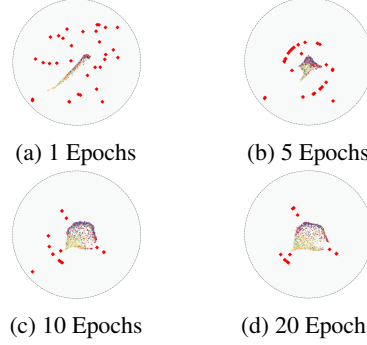(c) 10 Epochs

(d) 20 Epochs

Figure 4: **2D visualisation of learnable prototype positioning.** During early stages of HMSN training.

5   Hyperbolic Masked Siamese Networks with Ideal Prototypes

Learning prototypes by HMSN exhibits behaviour that results in the under utilisation of the hyperbolic embedding space. During training, the learnable prototypes tend towards the origin and converge to a region that is significantly distant from the boundary of the space, this phenomena is visualised in Figure 2 and Figure 4. This in-turn restricts the embedding space resulting in uncertain and less abstract positioning. A solution is to encourage the prototypes to lie closer to the boundary through additional regularisation term maximising euclidean norm, or to provide some prior to the embeddings to place them more akin to the hierarchies we aim to capture [29]. However, The former is naive and the latter requires annotations from human observers, not feasible in the self-supervised setting. To address this we place the prototypes at ideal points of the Poincaré ball.

The ideal points, $\mathbb{I}_d$, are positioned prior to training based on separation on the unit hypersphere $\mathbb{S}_d$ for $d \geq 3$ while positioned uniformly on $\mathbb{S}_d$ when $d = 2$ given ideal points of the hyperbolic space $\mathbb{D}_d$ are homeomorphic to $\mathbb{S}_d$ [29]. As the set of ideal points lies on the boundary of the hyperbolic space, the geodesic distance Eq.5 from an ideal point to any point in hyperbolic space is infinite. Therefore, to measure the assignment of a hyperbolic embedding to an ideal prototype the Busemann loss function is used. In the Poincaré ball model, the Busemann function is given by Eq.9.

$$b_{\mathbf{q}}(z_{i,m}) = \log \frac{\|\mathbf{q} - z_{i,m}\|^2}{(1 - \|z_{i,m}\|^2)} \tag{9}$$

The Busemann function [13] is considered to be a distance measured to infinity defined in any space. As with the hyperbolic reformulation in Section 4 we can substitute the cosine similarity for the Busemann function and position the prototypes at ideal points to produce the following prediction.

$$p_{i,m}^{\mathbb{I}} := \text{softmax}\left(\frac{-b_{\mathbf{q}}(z_{i,m})}{\tau} \right). \tag{10}$$

An important distinction from the work in [28] is that our function does not require a penalty term to penalise the overconfidence of the embeddings. Instead, the temperature τ scaling of the Softmax in Eq. 10 increases the magnitude of the hyperbolic embedding as τ decreases [31]. As a result, the embeddings are prevented from approaching the boundary of the ball as the Softmax is sharpened and certainty increases. In practice, tuning τ for performance whilst ensuring the embeddings do not lie on the boundary – the cause of vanishing gradients – is non-trivial. Instead, we clip the euclidean representations before the exponential mapping as done in 4. To avoid collapsed representations we introduce an entropy term to encourage unique prototype assignment [3], the resulting objective is,

$$\frac{1}{MB} \sum_{i=1}^{B} \sum_{m=1}^{M} H(p_i^{\mathbb{I}+}, p_{i,m}^{\mathbb{I}}) - \lambda H(\bar{p}^{\mathbb{I}}) + \beta H(p_{i,m}^{\mathbb{I}}). \tag{11}$$
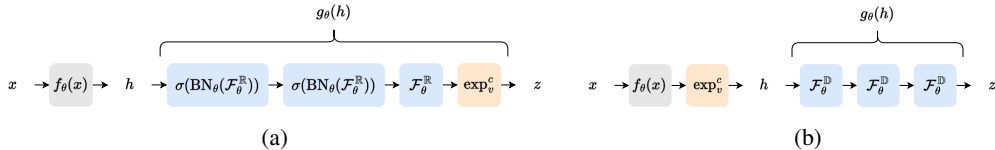
6

Figure 5: **Visualisation of the projection head architectures:** (a) The Euclidean projection head projects the euclidean embeddings to the Poincaré ball before the computation of the loss. (b) The Hyperbolic projector receiving Poincaré ball hyperbolic representations from the encoder.

6 Hyperbolic Projection Head

Most notable approaches to hyperbolic self-supervised learning propose solutions that are akin to the reformulation procedure aforementioned [24, 27, 52]. However, all such methods fail to address that the representations used for downstream tasks remain euclidean. We instead project the representations to hyperbolic space at the output of the encoder trunk before the projection head, $g_\theta(\cdot)$, rather than at the output of the projection head as done by prior methods (Figure 5). The motivation is that the projection head is present during training and then removed for downstream tasks which is when we intend to most effectively utilise the structure captured in hyperbolic space. Recent works [12] which examine the role of the projection head have identified its role to remove the training tasks overfitting bias, which when removed results in representations that are significantly more generalisable to downstream tasks.

Given that we propose to project to the hyperbolic space before the projection head, the Multi-Layer Perception (MLP) comprised of three fully-connected layers and their non-linear activation must remain hyperbolic to preserve the embedding structure. To achieve this, we employ a Hyperbolic projection head comprised of three Hyperbolic fully-connected layers, $\mathcal{F}_\theta^{\mathbb{D}}$, define in [41](Figure 5b) followed by a hyperbolic ReLU non-linearity [46] with exception to the final layer. We examine in more detail the effect of the projection head in Section 7.3.

7 Experimentation

To examine the quality of hyperbolic representations, a number of standardised benchmark tasks are performed. We first evaluate the representations learnt by the ViT encoder on the ImageNet-1K dataset under linear evaluation, followed by few-shot evaluation on ImageNet-1K using only 1% of the labelled training images per class[1] as per [3]. Given our representations lie in hyperbolic space we cannot directly compare using Euclidean classifiers, as such, we employ a hyperbolic multi-linear regression classifier with implementation identical to that reported in [41]. For all methods, we pre-train with a batch size of 1024 [2], producing views identically to [3] with 1 anchor, 1 random mask, and 10 focal mask views.

Table 1: **Linear Classification on ImageNet-1K**. Top-1 accuracy for linear models trained on frozen features from different self-supervised methods.

Method	Arch.	Params.	Epochs	Dims.	Top-1 (%)
SimCLR v2 [22]	RN50	24M	800	2048	71.7
BYOL [30]	RN50	24M	1000	2048	74.4
Barlow-T [53]	RN50	24M	1000	8192	73.2
VICReg [8]	RN50	24M	1000	8192	73.2
DINO [18]	ViT-S/16	22M	800	2048	77.0
iBOT [55]	ViT-S/16	22M	800	8192	77.9
MSN [3]	ViT-S/16	22M	800	256	76.9
HMSN (*ours*)	ViT-S/16	22M	800	64	76.0
HMSN-IP (*ours*)	ViT-S/16	22M	800	64	76.8

[1]ImageNet subsets: https://github.com/facebookresearch/msn
[2]ViT-S/16 Trained on 6x Nvidia A100 80GB GPUs for 800 epochs $\approx$ 150 hours.

7.1 Linear Evaluation

We train a hyperbolic linear classifier on the labelled ImageNet-1K training set on the representations produced by of our frozen pre-trained, self-supervised hyperbolic vision transformer. Table 1 reports the top-1 linear evaluation accuracies (%) of both our proposed method compared against other leading approaches on the ImageNet-1K validation set, the results are the average of 3 randomly initialised runs. The hyperbolic reformulation (HMSN) performs marginally worse that its euclidean baseline albeit with fewer embedding dimensions 64 instead of 256. On the other hand, the Hyperbolic Masked Siamese Networks with Ideal Prototypes (HMSN-IP) performs comparatively to the MSN baseline showing a 0.1% difference with the same reduction in embedding dimensions. Encouraging, a performance drop is not observed in HMSN-IP given the fixed ideal prototypes. We note that training HMSN results in uniformly distributed prototypes akin to the ideal prototypes, albeit positions far closer to the origin restricting the representation space (Figure 2).

Table 2: **Low-shot Linear Evaluation on ImageNet-1K**. Top-1 Accuracy for linear models trained on frozen features from different methods, fine-tuning only uses 1% of the labels.

Method	Arch.	Params.	Dims.	Top-1 (%)
Barlow-Twins [53]	RN50	24M	8192	55.0
SimCLR v2 [22]	RN50	24M	2048	57.9
PAWS [4]	RN50	24M	2048	66.5
DINO [18]	ViT-S/16	22M	2048	64.5
iBOT [55]	ViT-S/16	22M	8192	65.9
MSN [3]	ViT-S/16	22M	256	67.2
HMSN (*ours*)	ViT-S/16	22M	64	67.6
HMSN-IP (*ours*)	ViT-S/16	22M	64	68.7

7.2 Low-Shot Linear Evaluation

The ability to learn representations from unlabelled data that are of high enough quality to be used in downstream tasks with very few labelled examples is the key motivator behind self-supervised learning. Moreover, our design decisions to employ hyperbolic representation space to learn hierarchies and as such represent semantic concepts in a more structured manner are driven by the goal of improving few-shot learning. As with the linear evaluation, we pre-train our encoder on the ImageNet-1K dataset, freezing the weights and training a linear classifier on top using a subset of the ImageNet-1K labelled training set. The performance of our self-supervised models by performing linear evaluation on very few labelled examples for each class is reported in Table 2.

In the standard low-shot benchmark 1% of the ImageNet-1K labels are employed for linear evaluation (approximately 13 images per class), the results are presented in Table 2 alongside alternative competitive self-supervised methods. Our hyperbolic reformulation (HMSN) outperforms its Euclidean counterpart with a 0.4% performance improvement with the ViT-S/16. The extension, Hyperbolic Masked Siamese Networks with Ideal Prototypes (HMSN-IP) sees a further 1.0% improvement over the hyperbolic reformulation, HMSN.

7.3 Projection Head

The approaches described in 4 and 5 both make the important distinction from previous works [27, 52] regarding the projection head in the training procedure. Typically the projection head is disregarded from the reformulation of euclidean SSL methods into hyperbolic ones, where embeddings are typically hyperbolic and representations remain Euclidean for comparison in downstream tasks (visually depicted in 5). Therefore, hyperbolic properties are lost when utilising the representations in downstream tasks.

Table 3: **Hyperbolic and Euclidean Projection Heads.** Linear evaluation accuracy on the Imagenet-1K validation set training for both Hyperbolic ($\mathbb{D}$) and Euclidean ($\mathbb{R}$) classifiers.

Projector $g(\cdot)$	$\mathbb{R}$ **Top-1 (%)**	$\mathbb{D}$ **Top-1 (%)**
Euclidean	58.1	52.0
Hyperbolic	48.1	66.2

Table 3 reports the linear evaluation top-1 accuracy on the ImageNet-1K validation set for a pre-trained ViT-S/16 by HMSN-IP loss for 100 epochs [3] with either a Euclidean or Hyperbolic projection head. The downstream linear evaluation top-1 accuracy are given for both a Euclidean linear evaluation procedure as described in [3] and the Hyperbolic linear evaluation procedure (further details given in Supplementary Material). The results clearly demonstrate that when evaluating using a downstream hyperbolic classifier the HMSN-IP with the hyperbolic projection head produces representations that are of a higher hyperbolicity compared to Euclidean counterpart.

7.4 Embedding Dimensions

An important and unique property of the negative curvature hyperbolic space is the exponentially expanding volume with respect to distance from the origin. This results in a representation space that exhibits the volume necessary for separability at far fewer dimensions. To access this, we pre-train ViT-S/16 with baseline MSN and HMSN-IP on ImageNet-1K for 100 epochs under different output dimensions.

We report in Figure 6 the linear evaluation top-1 accuracy on the ImageNet-1K test of both Euclidean and Hyperbolic classifiers for each given dimension and projection head. We can see the expected increase in performance when the Euclidean projector dimensions are increased (*blue* and *orange* bars), this expected result aligns with previous investigations of the projection head [12]. For the Hyperbolic projector and hyperbolic classifier setting (*red*), we observe steady increase in accuracy until a significant drop-off occurred at 128 dimensions.
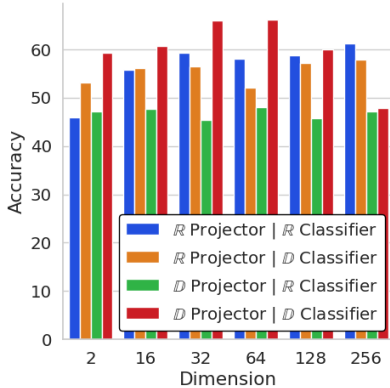

Figure 6: **Embedding Dimensions.** Linear evaluation accuracy on the Imagenet validation set training for Hyperbolic ($\mathbb{D}$) and Euclidean ($\mathbb{R}$) classifiers and projectors.

8 Conclusion

This work investigates the hyperbolic self-supervised learning, introducing a hyperbolic extension to the Masked Siamese Network model where we empirically show improved downstream performance in hyperbolic classifiers for linear evaluation, transfer learning, and low-shot learning. We further improve on this by introducing a method that instead uses prototypes placed on the ideal boundary of the Poincaré ball model. We empirically demonstrate that this method improves low-shot downstream task performance over the standard hyperbolic reformulation. Both our proposed methods outperform or perform competitively to their Euclidean counterparts but do so at fewer embedding dimensions (Figure 6) whilst exhibiting clear semantic class hierarchies (Figure 1b).

Limitations & Broader Impact Our work aims to produce better representations of images in setting where data annotations are scarce, it can therefore be seen how such methods can lead to more accurate or informative models for a number of downstream tasks with positive societal impact. However, as is the case with all vision systems, there is potential for exploitation and security concerns and one should take into consideration AI misuse when extending our method.

Hyperbolic self-supervision can improve the compactness of these representations, therefore providing promising research directions in applications such as data transmission and compression via SSL. Importantly, the improved interpretability due to uncertainty proxy of learned representations by embedding latent tree-like hierarchies leads to exciting new SSL understanding. However, computing in the hyperbolic space introduces challenges regarding matrix and vector operations and as such, there exists implementation and computational difficulties compared to Euclidean approaches. In practice, we do not find these significantly impactful at the presented scale, regardless, future work or extensions should take care in their case.

---

[3]Best performance is not achieved by 100 epochs, however, the results allow for reasonable ablations.

References

[1] R. Aly, S. Acharya, A. Ossa, A. Köhn, C. Biemann, and A. Panchenko. Every child should have parents: a taxonomy refinement algorithm based on hyperbolic term embeddings. *arXiv preprint arXiv:1906.02002*, 2019.

[2] Y. M. Asano, C. Rupprecht, and A. Vedaldi. Self-labelling via simultaneous clustering and representation learning. *arXiv preprint arXiv:1911.05371*, 2019.

[3] M. Assran, M. Caron, I. Misra, P. Bojanowski, F. Bordes, P. Vincent, A. Joulin, M. Rabbat, and N. Ballas. Masked siamese networks for label-efficient learning. *arXiv preprint arXiv:2204.07141*, 2022.

[4] M. Assran, M. Caron, I. Misra, P. Bojanowski, A. Joulin, N. Ballas, and M. Rabbat. Semi-supervised learning of visual features by non-parametrically predicting view assignments with support samples. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8443–8452, 2021.

[5] M. Assran, Q. Duval, I. Misra, P. Bojanowski, P. Vincent, M. Rabbat, Y. LeCun, and N. Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. *arXiv preprint arXiv:2301.08243*, 2023.

[6] P. Bachman, R. D. Hjelm, and W. Buchwalter. Learning representations by maximizing mutual information across views. In *Advances in Neural Information Processing Systems*, pages 15535–15545, 2019.

[7] G. Bachmann, G. Bécigneul, and O. Ganea. Constant curvature graph convolutional networks. In *International Conference on Machine Learning*, pages 486–496. PMLR, 2020.

[8] A. Bardes, J. Ponce, and Y. LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. *arXiv preprint arXiv:2105.04906*, 2021.

[9] G. Bécigneul and O.-E. Ganea. Riemannian adaptive optimization methods. *arXiv preprint arXiv:1810.00760*, 2018.

[10] Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

[11] Y. Bi, B. Fan, and F. Wu. Beyond mahalanobis metric: cayley-klein metric learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2339–2347, 2015.

[12] F. Bordes, R. Balestriero, Q. Garrido, A. Bardes, and P. Vincent. Guillotine regularization: Improving deep networks generalization by removing their head. *arXiv preprint arXiv:2206.13378*, 2022.

[13] H. Busemann. *The geometry of geodesics*. Courier Corporation, 2012.

[14] J. W. Cannon, W. J. Floyd, R. Kenyon, W. R. Parry, et al. Hyperbolic geometry. *Flavors of geometry*, 31(59-115):2, 1997.

[15] M. Caron, P. Bojanowski, A. Joulin, and M. Douze. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European conference on computer vision (ECCV)*, pages 132–149, 2018.

[16] M. Caron, P. Bojanowski, J. Mairal, and A. Joulin. Unsupervised pre-training of image features on non-curated data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2959–2968, 2019.

[17] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *34th Conference on Neural Information Processing Systems, NeurIPS'20*, volume 33, pages 9912–9924. Curran Associates, Inc., 2020.

[18] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin. Emerging properties in self-supervised vision transformers. *arXiv preprint arXiv:2104.14294*, 2021.

[19] I. Chami, Z. Ying, C. Ré, and J. Leskovec. Hyperbolic graph convolutional neural networks. *Advances in neural information processing systems*, 32, 2019.

[20] H. Chang, H. Zhang, J. Barber, A. Maschinot, J. Lezama, L. Jiang, M.-H. Yang, K. Murphy, W. T. Freeman, M. Rubinstein, et al. Muse: Text-to-image generation via masked generative transformers. *arXiv preprint arXiv:2301.00704*, 2023.

[21] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[22] X. Chen, H. Fan, R. Girshick, and K. He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020.

[23] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[24] A. Ermolov, L. Mirvakhabova, V. Khrulkov, N. Sebe, and I. Oseledets. Hyperbolic vision transformers: Combining improvements in metric learning. *arXiv preprint arXiv:2203.10833*, 2022.

[25] P. Fang, M. Harandi, and L. Petersson. Kernel methods in hyperbolic spaces. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10665–10674, 2021.

[26] O. Ganea, G. Bécigneul, and T. Hofmann. Hyperbolic neural networks. *Advances in neural information processing systems*, 31, 2018.

[27] S. Ge, S. Mishra, S. Kornblith, C.-L. Li, and D. Jacobs. Hyperbolic contrastive learning for visual representations beyond objects. *arXiv preprint arXiv:2212.00653*, 2022.

[28] M. Ghadimi Atigh, M. Keller-Ressel, and P. Mettes. Hyperbolic busemann learning with ideal prototypes. *Advances in Neural Information Processing Systems*, 34, 2021.

[29] M. GhadimiAtigh, J. Schoep, E. Acar, N. van Noord, and P. Mettes. Hyperbolic image segmentation. *arXiv preprint arXiv:2203.05898*, 2022.

[30] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. Richemond, E. Buchatskaya, C. Doersch, B. Pires, Z. Guo, M. Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. In *Neural Information Processing Systems*, 2020.

[31] Y. Guo, X. Wang, Y. Chen, and S. X. Yu. Clipped hyperbolic classifiers are super-hyperbolic classifiers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11–20, 2022.

[32] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009, 2022.

[33] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2020.

[34] R. D. Hjelm, A. Fedorov, S. Lavoie-Marchildon, K. Grewal, P. Bachman, A. Trischler, and Y. Bengio. Learning deep representations by mutual information estimation and maximization. *arXiv preprint arXiv:1808.06670*, 2018.

[35] V. Khrulkov, L. Mirvakhabova, E. Ustinova, I. Oseledets, and V. Lempitsky. Hyperbolic image embeddings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6418–6428, 2020.

[36] Q. Liu, M. Nickel, and D. Kiela. Hyperbolic graph neural networks. *Advances in Neural Information Processing Systems*, 32, 2019.

[37] I. Misra and L. v. d. Maaten. Self-supervised learning of pretext-invariant representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6707–6717, 2020.

[38] A. v. d. Oord, Y. Li, and O. Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[39] P. H. Richemond, J.-B. Grill, F. Altché, C. Tallec, F. Strub, A. Brock, S. Smith, S. De, R. Pascanu, B. Piot, et al. Byol works even without batch statistics. *arXiv preprint arXiv:2010.10241*, 2020.

[40] R. Sarkar. Low distortion delaunay embedding of trees in hyperbolic plane. In *International Symposium on Graph Drawing*, pages 355–366. Springer, 2011.

[41] R. Shimizu, Y. Mukuta, and T. Harada. Hyperbolic neural networks++. *arXiv preprint arXiv:2006.08210*, 2020.

[42] Y. Tian, D. Krishnan, and P. Isola. Contrastive multiview coding. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16*, pages 776–794. Springer, 2020.

[43] A. Tifrea, G. Bécigneul, and O.-E. Ganea. Poincar\'e glove: Hyperbolic word embeddings. *arXiv preprint arXiv:1810.06546*, 2018.

[44] N. Tomasev, I. Bica, B. McWilliams, L. Buesing, R. Pascanu, C. Blundell, and J. Mitrovic. Pushing the limits of self-supervised resnets: Can we outperform supervised learning without labels on imagenet? *arXiv preprint arXiv:2201.05119*, 2022.

[45] A. A. Ungar. A gyrovector space approach to hyperbolic geometry. *Synthesis Lectures on Mathematics and Statistics*, 1(1):1–194, 2008.

[46] M. van Spengler, E. Berkhout, and P. Mettes. Poincar\'e resnet. *arXiv preprint arXiv:2303.14027*, 2023.

[47] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3733–3742, 2018.

[48] Z. Xie, Z. Zhang, Y. Cao, Y. Lin, J. Bao, Z. Yao, Q. Dai, and H. Hu. Simmim: A simple framework for masked image modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9653–9663, 2022.

[49] Y. Xu, L. Mu, Z. Ji, X. Liu, and J. Han. Meta hyperbolic networks for zero-shot learning. *Neurocomputing*, 491:57–66, 2022.

[50] J. Yan, L. Luo, C. Deng, and H. Huang. Unsupervised hyperbolic metric learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12465–12474, 2021.

[51] X. Yan, I. Misra, A. Gupta, D. Ghadiyaram, and D. Mahajan. Clusterfit: Improving generalization of visual representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6509–6518, 2020.

[52] Y. Yue, F. Lin, K. D. Yamada, and Z. Zhang. Hyperbolic contrastive learning. *arXiv preprint arXiv:2302.01409*, 2023.

[53] J. Zbontar, L. Jing, I. Misra, Y. LeCun, and S. Deny. Barlow twins: Self-supervised learning via redundancy reduction. *arXiv preprint arXiv:2103.03230*, 2021.

[54] Y. Zhang, X. Wang, C. Shi, X. Jiang, and Y. F. Ye. Hyperbolic graph attention network. *IEEE Transactions on Big Data*, 2021.

[55] J. Zhou, C. Wei, H. Wang, W. Shen, C. Xie, A. Yuille, and T. Kong. ibot: Image bert pre-training with online tokenizer. *arXiv preprint arXiv:2111.07832*, 2021.

[56] C. Zhuang, A. L. Zhai, and D. Yamins. Local aggregation for unsupervised learning of visual embeddings. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6002–6012, 2019.