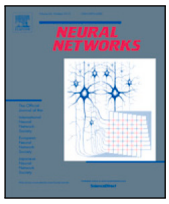
Neural Networks 181 (2025) 106803

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet



Full Length Article

Adaptive indefinite kernels in hyperbolic spaces

Pengfei Fang *

School of Computer Science and Engineering, Southeast University, Nanjing, 210096, China
Key Laboratory of New Generation Artificial Intelligence Technology and Its Interdisciplinary Applications (Southeast University), Ministry of Education, China

ARTICLE INFO

Keywords:
Hyperbolic space
Data hierarchy
Indefinite Lorentz kernels
Indefinite Poincaré kernels

ABSTRACT

Learning embeddings in hyperbolic space has gained increasing interest in the community, due to its property of negative curvature, as a way of encoding data hierarchy. Recent works investigate the improvement of the representation power of hyperbolic embeddings through kernelization. However, existing developments focus on defining positive definite (pd) kernels, which may affect the intriguing property of hyperbolic spaces. This is due to the structures of hyperbolic spaces being modeled in indefinite spaces (*e.g.*, Kreĭn space). This paper addresses this issue by developing adaptive indefinite kernels, which can better utilize the structures in the Kreĭn space. To this end, we first propose an adaptive embedding function in the Lorentz model and define indefinite Lorentz kernels (iLks) via the embedding function. Due to the isometric relationship between the Lorentz model and the Poincaré ball, these iLks are further extended to the Poincaré ball, resulting in the development of what are termed indefinite Poincaré kernels (iPKs). We evaluate the proposed indefinite kernels on a diversity of learning scenarios, including image classification, few-shot learning, zero-shot learning, person re-identification, knowledge distillation, *etc*. We show that the proposed indefinite kernels can bring significant performance gains over the baselines and enjoy better representation power from RKKSs than pd kernels.

1. Introduction

In this paper, we study the potential of employing indefinite kernels in the hyperbolic space for the deep neural networks (DNNs). Specifically, we develop a set of indefinite kernels to map the hyperbolic representations in the Lorentz model and Poincaré ball into the Reproducing Kernel Kreĭn Spaces (RKKSs), where the embeddings can preserve the properties of negative curvature in hyperbolic geometry and jointly benefit from the representation power of kernel methods in Kreĭn spaces.

Recent advances in the learning community have explored the benefit of the hyperbolic space as the feature embeddings (Peng, Varanka, Mostafa, Shi, & Zhao, 2021). This is due to the fact that the hyperbolic space reduces the distortion of encoding the data hierarchy, as a result of its negative curvature (Bdeir, Schwethelm, & Landwehr, 2023; Ganea, Bécigneul, & Hofmann, 2018). The intriguing property enables the learning community to develop learning algorithms (Mathieu, Le Lan, Maddison, Tomioka, & Teh, 2019; Nagano, Yamaguchi, Fujita, & Koyama, 2019) or neural networks (Ganea et al., 2018; Khrulkov, Mirvakhabova, Ustinova, Oseledets, & Lempitsky, 2020), creating discriminative feature embeddings for various data, *e.g.*, language data, image data, graph data *etc*. Ganea et al. first defined a series of hyperbolic neural layers, as the building blocks for the hyperbolic neural networks, and demonstrated the improvement on the NLP task (Ganea et al., 2018). Graph-structured data is built inherently by hierarchy, and the work in Liu, Nickel, and Kiela (2019b) developed graph neural networks in the hyperbolic space, enabling graph representation learning. The hierarchical structure is also successfully modeled for the image data, where the hierarchy arises for the whole-fragment relation of the object or the degradation of images (Khrulkov et al., 2020).

Even though effective hyperbolic algorithms have been extensively studied, they still suffer from the complex computation in hyperbolic space. For example, calculating the distance metric in the hyperbolic space requires computing the hyperbolic cosine or hyperbolic tangent functions (Ganea et al., 2018; Gulcehre et al., 2019). In addition, averaging points in the hyperbolic spaces leverages the transition in the Klein space (Khrulkov et al., 2020). These basic operations bring additional computational costs for the learning algorithms. The kernel methods in the hyperbolic space have been proposed to address this issue since the kernel function can be understood as the similarity measure of the points and realized by implicit mappings, which can significantly save the computational cost, along with considerable performance gain for DNNs (Fang, Harandi, Lan & Petersson, 2023a; Fang, Harandi, & Petersson, 2021a; Yang, Fang, & Xue, 2023). In Fang, Harandi, Le and Phung (2023b), Fang, Harandi et al. (2021a), two types of bijective functions were proposed to map the hyperbolic embedding

* Correspondence to: School of Computer Science and Engineering, Southeast University, Nanjing 210096, China
E-mail address: fangpengfei@seu.edu.cn.

https://doi.org/10.1016/j.neunet.2024.106803
Received 11 May 2024; Received in revised form 17 August 2024; Accepted 10 October 2024
Available online 16 October 2024

to the tangent space or spherical space, where the positive definite (pd) kernels can be well defined. Yang et al. proposed an isometric embedding function, which can well preserve the geometric property in the kernel. To make those kernels pd, the authors adopt the Poincaré ball as the modeling of hyperbolic space. In this context, the Lorentz model, another popular model of hyperbolic space, has not been studied thoroughly. The Lorentz model is defined in the Kreĭn space, where indefinite kernels can be defined (Bognár, 1974; Ong, Mary, Canu, & Smola, 2004). This raises a question: *Can the indefinite kernels defined in the Lorentz model still benefit the DNNs and explore more information for the hyperbolic space?*

This paper studies this question by proposing adaptive indefinite kernels in hyperbolic space. To be specific, we first employ the Lorentz model to represent the hyperbolic space and propose an adaptive embedding function. The adaptive embedding function is realized by projecting one point to the tangent space of another. Optimization over the kernels makes the embedding function adaptive to the local relations of samples. With the proposed adaptive embedding function serving as a bridge, one can define indefinite kernels on the Lorentz model. Since the Lorentz model and the Poincaré ball are isometric, one can further develop the adaptive embedding function in the Poincaré ball and extend the indefinite kernels there. Extensive studies show that the proposed indefinite kernels outperform their pd counterpart and even surpass the advanced multiple kernel learning scheme in the Poincaré ball.

The contribution of this paper can be summarized as follows:

- We propose three indefinite kernels in the Lorentz model, namely, indefinite Lorentz Tangent kernel, indefinite Lorentz RBF kernel, and indefinite Lorentz Laplace kernel.
- We further extend the proposed indefinite kernels in Lorentz model to the Poincaré ball, named indefinite Poincaré Tangent kernel, indefinite Poincaré RBF kernel, and indefinite Poincaré Laplace kernel.
- Thorough empirical studies on image classification, few-shot learning, zero-shot learning, person re-identification and knowledge distillation tasks demonstrate the effectiveness of these indefinite kernels.

The code will be made available freely to the research community.

2. Related work

This paper presents the kernel methods in hyperbolic space. In this paper, we briefly survey the related work including the hyperbolic embedding learning and hyperbolic kernel.

This paper presents the kernel methods in hyperbolic space and this section briefly surveys related work, including hyperbolic embedding learning and hyperbolic kernels.

2.1. Hyperbolic embedding learning

Building the hierarchical structure of concepts benefits the understanding of the world for humans, and many works also explore the benefits of encoding the data hierarchy in learning algorithms. However, Bourgain's theorem (Linial, London, & Rabinovich, 1995) has theoretically proven that Euclidean space is unable to encode comparably low distortion for trees. That said, encoding inherited such structured data causes distortion in the flat space. Hyperbolic space becomes an alternative to address this issue, encoding the hierarchical relationships for language, graph, and visual data (Fang, Harandi, Le et al., 2023b; Peng et al., 2021). Ganea et al. first generalize the basic operations in hyperbolic space, bridging the gap of defining building components for hyperbolic neural networks (Ganea et al., 2018). The following works in Bdeir et al. (2023), Gulcehre et al. (2019), Liu, Nickel et al. (2019b) further develop more complex neural networks, *e.g.*, attention network, graph neural network, and convolutional network, which expand the usage of hyperbolic geometry. To facilitate the variational algorithm, *e.g.*, variational auto-encoder works in Mathieu et al. (2019), Nagano et al. (2019) also propose the Gaussian distribution in the hyperbolic spaces endowing the hierarchical encoding in the latent space. Along with the closed-set learning tasks (Peng et al., 2021), recent years also demonstrate the hyperbolic geometry attains better generalization capacity to the open-set tasks, including few-shot learning (Khrulkov et al., 2020), zero-shot learning (Liu et al., 2020), anomaly recognition (Hong et al., 2023), *etc*. These tasks either benefit the geodesic property of the hyperbolic space or define more efficient metrics in the curved space, clearly showing the importance of leveraging data hierarchy in real-world applications.

2.2. Hyperbolic kernel

The hyperbolic space has shown its practical potential in many machine learning applications, given its intriguing property of encoding the data hierarchy. To enable the hyperbolic embedding to enjoy the rich representation power of the reproducing kernel Hilbert space (RKHS), recent works develop the positive definite (pd) kernels in the hyperbolic space (Cho, DeMeo, Peng, & Berger, 2019; Fang, Harandi, Lan et al., 2023a; Fang, Harandi et al., 2021a; Yang et al., 2023).

Cho et al. first develop a valid Minkowski inner product kernel, which is defined in the hyperboloid model (Cho et al., 2019). It brings nonlinearity to boost SVM in the hyperbolic space. Fang et al. propose a bijective embedding function, which enables the development of valid radial kernels, *e.g.*, hyperbolic RBF kernel and hyperbolic Laplace kernel, in the Poincaré ball (Fang, Harandi et al., 2021a). The bijective embedding function enjoys the property of curve length equivalence, guarantying the data hierarchy and RKHS are leveraged by those kernels. On top of the embedding function proposed in Fang, Harandi et al. (2021a), a normalized embedding function is further studied, facilitating the proposal of multiple-kernel learning (MKL) in the hyperbolic space (Fang, Harandi, Lan et al., 2023a). In contrast to Fang, Harandi, Lan et al. (2023a), Fang, Harandi et al. (2021a), an isometric embedding function, which establishes an isometry from the Poincaré ball to a special RKHS, is proposed in Yang et al. (2023). This isometric embedding function can jointly reduce the distortion of data hierarchy and fit the data adaptively in hyperbolic spaces.

2.3. Indefinite kernel

Kernel methods have been studied extensively in the learning community, and can enable the machine to encode the non-linear representations of the data (Hofmann, Scholkopf, & Smola, 2008). Along with the positive definite (pd) kernels, recent studies on the non-positive kernels, a.k.a., indefinite kernels, also gain research interest as a generalization of the pd kernels (Schleif & Tino, 2015). In Ong et al. (2004), Ong et al. developed the theoretical framework of the indefinite kernels and the learning procedure of such kernels. The following efforts have been made to develop various formulations of indefinite kernels, including the hyperbolic tangent kernel (Smola, Óvári, & Williamson, 2000), the truncated l_1 kernel (Huang, Suykens, Wang, Hornegger, & Maier, 2018), geodesic kernel (Feragen, Lauze, & ren Hauberg, 2015), *etc*., and the superior of those kernel are also proven in the machine learning field. To adapt the indefinite kernels to the large-scale learning issue, Liu et al. further developed random features (Liu, Huang, Shi, Yang & Suykens, 2019a) for indefinite kernels, and positive decomposition via the signed measure (Liu, Huang, Chen, & Suykens, 2021). In contrast to the existing works, our work focuses on developing indefinite kernels in the hyperbolic space, making the learning algorithm enjoy the rich structures in hyperbolic space.

3. Preliminaries and background

3.1. Notations

Formally, we use $\mathbb{H}^n$, $\mathbb{R}^n$ and $\mathbb{R}^{m\times n}$ to denote n-dimensional hyperbolic spaces, n-dimensional Euclidean spaces and the space of $m \times n$ real matrices. The symbols $\mathcal{V}$, $\mathcal{H}$ and $\mathcal{K}$ denote the vector space, Hilbert space and Kreĭn space, respectively. Throughout the paper, the matrices and vectors are denoted by bold capital letters (*e.g.*, $\boldsymbol{X}$) and bold lowercase letters (*e.g.*, $\boldsymbol{x}$), respectively. The transpose of a matrix (*e.g.*, $\boldsymbol{X}$) or a vector (*e.g.*, $\boldsymbol{x}$) is denoted by the superscript $\top$, *e.g.* $\boldsymbol{X}^\top$ or $\boldsymbol{x}^\top$. $\sinh(\cdot) : \mathbb{R} \to \mathbb{R}, \sinh(x) := \frac{1-e^{-2x}}{2e^{-x}}$, $\cosh(\cdot) : \mathbb{R} \to \mathbb{R}, \cosh(x) := \frac{1+e^{-2x}}{2e^{-x}}$ and $\tanh(\cdot) : \mathbb{R} \to \mathbb{R}, \tanh(x) := \frac{e^{2x}-1}{e^{2x}+1}$ refer to the hyperbolic sine function, the hyperbolic cosine function and the hyperbolic tangent function, respectively.

3.2. Hyperbolic geometry

The n-dimensional hyperbolic geometry, denoted by $\mathbb{H}^n$, is characterized by a Riemannian manifold with a constant sectional negative curvature (Absil, Mahony, & Sepulchre, 2007). Five isometric models, including the Poincaré ball, the Poincaré half-space model, the Lorentz model, the Klein model, and the Hemisphere model, are employed to model the hyperbolic space (Cannon, Floyd, Kenyon, & Parry, 1997). Among them, the Lorentz model and Poincaré ball have been used extensively in the learning community (Cho et al., 2019; Ganea et al., 2018; Khrulkov et al., 2020; Liu, Nickel et al., 2019b). In this paper, we develop the indefinite kernels on top of the Lorentz model and Poincaré ball.

3.3. Lorentz model

We first introduce the Lorentz model. An n dimensional Lorentz model with curvature $-c, c > 0$ is denote by $\mathbb{L}_c^n$, identifying a manifold embedded in the $n+1$ dimensional Minkowski space, defined as $\mathbb{L}_c^n = \{\boldsymbol{l} \in \mathbb{R}^{n+1} : c\langle \boldsymbol{l}, \boldsymbol{l}\rangle_{\mathbb{L}} = -1, l_0 > 0\}$, where $\langle , \rangle_{\mathbb{L}}$ represents the Lorentzian inner product. For $\boldsymbol{l}_i, \boldsymbol{I}_j \in \mathbb{L}_c^n$, the Lorentzian inner product is given by

$$\langle \boldsymbol{l}_i, \boldsymbol{l}_j\rangle_{\mathbb{L}} = \boldsymbol{l}_i^\top g^{\mathbb{L}} \boldsymbol{l}_j = -l_{i,0}l_{j,0} + \sum_{t=1}^{n} l_{i,t}l_{j,t}, \tag{1}$$

where $g^{\mathbb{L}}$ is the riemannian metric for the Lorentz model. One can further define the geodesic distance on the Lorentz model for two points $\boldsymbol{l}_i, \boldsymbol{l}_j \in \mathbb{L}_c^n$, as

$$d_c^{\mathbb{L}}(\boldsymbol{l}_i, \boldsymbol{l}_j) = \frac{1}{\sqrt{c}} \cosh^{-1}(-c\langle \boldsymbol{l}_i, \boldsymbol{l}_j\rangle_{\mathbb{L}}) \tag{2}$$

The exponential map provides a way to project a point $\boldsymbol{u} \in T_{\boldsymbol{l}}\mathbb{L}_c^n$ to the Lorentz model $\mathbb{L}_c^n$, as follows:

$$\exp_{\boldsymbol{l}}^{\mathbb{L}}(\boldsymbol{u}) = \cosh(\sqrt{c}\langle \boldsymbol{u}, \boldsymbol{u}\rangle_{\mathbb{L}})\boldsymbol{l} + \boldsymbol{u}\frac{\sinh(\sqrt{c}\langle \boldsymbol{u}, \boldsymbol{u}\rangle_{\mathbb{L}})}{\sqrt{c\langle \boldsymbol{u}, \boldsymbol{u}\rangle_{\mathbb{L}}}}, \tag{3}$$

The inverse process is termed logarithm map, which projects a point $\boldsymbol{v} \in \mathbb{L}_c^n$, to the tangent plane of $\boldsymbol{l}$, given by:

$$\log_{\boldsymbol{l}}^{\mathbb{L}}(\boldsymbol{v}) = \frac{\cosh^{-1}(-c\langle \boldsymbol{l}, \boldsymbol{v}\rangle_{\mathbb{L}})}{\sinh\left(\cosh^{-1}(-c\langle \boldsymbol{l}, \boldsymbol{v}\rangle_{\mathbb{L}})\right)}(\boldsymbol{v} + c\langle \boldsymbol{l}, \boldsymbol{v}\rangle_{\mathbb{L}}\boldsymbol{l}). \tag{4}$$

In the Lorentz model, it holds that $\log_{\boldsymbol{l}}^{\mathbb{L}}\left(\exp_{\boldsymbol{l}}^{\mathbb{L}}(\boldsymbol{u})\right) = \boldsymbol{u} \in T_{\boldsymbol{l}}\mathbb{L}_c^n$.

3.4. Poincaré ball

Another popular model is known as the Poincaré ball, a model of n-dimensional hyperbolic geometry in which all points are embedded within an n-dimensional sphere. In this regard, we formally define the Poincaré ball, with curvature $-c$, as a manifold $\mathbb{D}_c^n = \{\boldsymbol{p} \in \mathbb{R}^n : c\|\boldsymbol{p}\| < 1\}$. In $\mathbb{D}_c^n$, the Riemannian metric is calculated by $g_{\boldsymbol{p}}^{\mathbb{D}}(\boldsymbol{p}) = \lambda_c^2(\boldsymbol{p}) \cdot g^E$, in which $\lambda_c(\boldsymbol{p})$ is the conformal factor, defined as $\frac{2}{1-c\|\boldsymbol{p}\|^2}$, and $g^E = \mathbf{I}_n$ is the Euclidean metric tensor.

To enable the basic vector operations, *e.g.*, vector addition, the Möbius gyrovector space comes in handy. We first introduce the Möbius addition, the basic operation to define other necessary algebraic formalism, For any two points, $\boldsymbol{p}_i, \boldsymbol{p}_j \in \mathbb{D}_c^n$, the Möbius addition is given by:

$$\boldsymbol{p}_i \oplus_c \boldsymbol{p}_j = \frac{(1 + 2c\langle \boldsymbol{p}_i, \boldsymbol{p}_j\rangle + c\|\boldsymbol{p}_j\|^2)\boldsymbol{p}_i + (1 - c\|\boldsymbol{p}_i\|^2)\boldsymbol{p}_j}{1 + 2c\langle \boldsymbol{p}_i, \boldsymbol{p}_j\rangle + c^2\|\boldsymbol{p}_i\|^2\|\boldsymbol{p}_j\|^2}. \tag{5}$$

The geodesic distance on $\mathbb{D}_c^n$ is:

$$d_c^{\mathbb{D}}(\boldsymbol{p}_i, \boldsymbol{p}_j) = \frac{2}{\sqrt{c}} \tanh^{-1}(\sqrt{c}\|-\boldsymbol{p}_i \oplus_c \boldsymbol{p}_j\|). \tag{6}$$

In the Poincaré ball, the exponential map can be derived as

$$\exp_{\boldsymbol{z}}^{\mathbb{D}}(\boldsymbol{q}) = \boldsymbol{z} \oplus_c \left(\tanh(\sqrt{c}\frac{\lambda_c(\boldsymbol{z}) \cdot \|\boldsymbol{q}\|}{2})\frac{\boldsymbol{q}}{\sqrt{c}\|\boldsymbol{q}\|}\right), \tag{7}$$

where $\boldsymbol{z} \in \mathbb{D}_c^n$ and $\boldsymbol{q} \in T_{\boldsymbol{z}}\mathbb{D}_c^n$. The logarithm map is

$$\log_{\boldsymbol{z}}^{\mathbb{D}}(\boldsymbol{p}) = \frac{2}{\sqrt{c}\lambda_c(\boldsymbol{z})} \tanh^{-1}(\sqrt{c}\|-\boldsymbol{z} \oplus_c \boldsymbol{p}\|)\frac{-\boldsymbol{z} \oplus_c \boldsymbol{p}}{\|-\boldsymbol{z} \oplus_c \boldsymbol{p}\|}, \tag{8}$$

where $\boldsymbol{z}, \boldsymbol{p} \in \mathbb{D}_c^n$. Note that it also holds that $\log_{\boldsymbol{z}}^{\mathbb{D}}\left(\exp_{\boldsymbol{z}}^{\mathbb{D}}(\boldsymbol{q})\right) = \boldsymbol{q} \in T_{\boldsymbol{z}}\mathbb{D}_c^n$ in the Poincaré ball.

3.5. Translation between Poincaré ball and Lorentz model

Since the Poincaré ball and the Lorentz model are isometric models, meaning that those tow models can be translated freely. For a point $\boldsymbol{l} \in \mathbb{L}_c^n, \boldsymbol{l} = [l_0, l_1, \ldots, l_n]$, $T_{\mathbb{L}\to\mathbb{D}}(\cdot)$ realizes the mapping from Lorentz model to the Poincaré ball, as

$$T_{\mathbb{L}\to\mathbb{D}}(\cdot) : \mathbb{L}_c^n \to \mathbb{D}_c^n, T_{\mathbb{L}\to\mathbb{D}}(\boldsymbol{l}) := \frac{(l_1, \ldots, l_n)}{l_0 + 1}. \tag{9}$$

The inverse translation is denoted by $T_{\mathbb{B}\to\mathbb{L}}(\cdot)$, defined as

$$T_{\mathbb{D}\to\mathbb{L}}(\cdot) : \mathbb{D}_c^n \to \mathbb{L}_c^n, T_{\mathbb{D}\to\mathbb{L}}(\boldsymbol{z}) := \frac{(1 + \|\boldsymbol{z}\|^2, 2z_1, \ldots, 2z_n)}{1 - \|\boldsymbol{z}\|^2}. \tag{10}$$

4. Method

In this section, we propose a series of indefinite kernels in the hyperbolic space. The pd kernels in the hyperbolic space have been developed recently and benefit the representation learning for the DNNs. Since the initial modeling of the hyperbolic space lies in the Kreĭn space, where one can define the indefinite kernels. That said, developing indefinite kernels in hyperbolic space can enjoy its intriguing properties properly. This paper proposes a set of adaptive kernels in the hyperbolic spaces and studies the potential of the indefinite kernels on the DNNs.

We start by formally define the inner product and Kreĭn space.

Definition 1 (*Inner Product*)**.** Let $\mathcal{V}$ be a vector space on the scalar field.[1] An inner product $\langle , \rangle_{\mathcal{V}}$ on $\mathcal{V}$ is a bilinear form where for all $\boldsymbol{f}, \boldsymbol{g}, \boldsymbol{h} \in \mathcal{V}$, $\alpha \in \mathbb{R}$:

- $\langle \boldsymbol{f}, \boldsymbol{g}\rangle_{\mathcal{V}} = \langle \boldsymbol{g}, \boldsymbol{f}\rangle_{\mathcal{V}}$
- $\langle \alpha\boldsymbol{f} + \boldsymbol{g}, \boldsymbol{h}\rangle_{\mathcal{V}} = \alpha\langle \boldsymbol{f}, \boldsymbol{h}\rangle_{\mathcal{V}} + \langle \boldsymbol{g}, \boldsymbol{h}\rangle_{\mathcal{V}}$
- $\langle \boldsymbol{f}, \boldsymbol{g}\rangle_{\mathcal{V}} = 0$ for all $\boldsymbol{g} \in \mathcal{V}$ implies $\Rightarrow \boldsymbol{f} = \boldsymbol{0}$

[1] Like Hilbert spaces, Kreĭn space can be defined on $\mathbb{R}$ or $\mathbb{C}$. We use $\mathbb{R}$ in this paper.

An inner product is said to be positive if for all $\boldsymbol{f} \in \mathcal{K}$ we have $\langle \boldsymbol{f}, \boldsymbol{f} \rangle \geq 0$. It is negative if for all $f \in \mathcal{K}$ $\langle \boldsymbol{f}, \boldsymbol{f} \rangle \leq 0$. Otherwise, it is called *indefinite*.

Definition 2 (*Kreĭn Space*)**.** An inner product space $(\mathcal{K}, \langle , \rangle_{\mathcal{K}})$ is a Kreĭn space if there exist two Hilert spaces $\mathcal{H}_+$, $\mathcal{H}_-$ spanning $\mathcal{K}$ such that

- All $\boldsymbol{f} \in \mathcal{K}$ can be decomposed into $\boldsymbol{f} = \boldsymbol{f}_+ + \boldsymbol{f}_-$, where $f_+ \in \mathcal{H}_+$ and $f_- \in \mathcal{H}_-$.
- $\forall \boldsymbol{f}, \boldsymbol{g} \in \mathcal{K}$, $\langle \boldsymbol{f}, \boldsymbol{g} \rangle_{\mathcal{K}} = \langle \boldsymbol{f}_+, \boldsymbol{g}_+ \rangle_{\mathcal{H}_+} - \langle \boldsymbol{f}_-, \boldsymbol{g}_- \rangle_{\mathcal{H}_-}$.

This suggests that there is an "associated" Hilbert space, where the difference in scalar products is replaced by a sum:

Definition 3 (*Associated Hilbert Space*)**.** Let $\mathcal{K}$ be a kreĭn space with decomposition into Hilbert spaces $\mathcal{H}_+$ and $\mathcal{H}_-$. Then we denote by $\bar{\mathcal{K}}$ the associated Hilbert space defined by $\bar{\mathcal{K}} = \mathcal{H}_+ \oplus \mathcal{H}_-$ hence $\langle \boldsymbol{f}, \boldsymbol{g} \rangle_{\bar{\mathcal{K}}} = \langle \boldsymbol{f}_+, \boldsymbol{g}_+ \rangle_{\mathcal{H}_+} + \langle \boldsymbol{f}_-, \boldsymbol{g}_- \rangle_{\mathcal{H}_-}$ Likewise, we can introduce the symbol $\ominus$ to indicate that $\mathcal{K} = \mathcal{H}_+ \ominus \mathcal{H}_-$ hence $\langle \boldsymbol{f}, \boldsymbol{g} \rangle_{\mathcal{K}} = \langle \boldsymbol{f}_+, \boldsymbol{g}_+ \rangle_{\mathcal{H}_+} - \langle \boldsymbol{f}_-, \boldsymbol{g}_- \rangle_{\mathcal{H}_-}$.

Note that $\bar{\mathcal{K}}$ is the smallest Hilbert space majorizing the Kreĭn space $\mathcal{K}$ and one defines the strong topology on $\mathcal{K}$ as the Hilbertian topology of $\bar{\mathcal{K}}$. The topology does not depend on the decomposition chosen. Clearly, it holds that $|\langle \boldsymbol{f}, \boldsymbol{f} \rangle_{\mathcal{K}}| \leq \|\boldsymbol{f}\|^2_{\bar{\mathcal{K}}}$.

Definition 4 (*RKKS*)**.** A Kreĭn space $(\mathcal{K}, \langle , \rangle_{\mathcal{K}})$ is a Reproducing Kernel Kreĭn Spaces (Alpay, 2002) if $\mathcal{K} \in \mathbb{R}^{\mathcal{X}}$ and the evaluation functional is continuous on $\mathcal{K}$ endowed with its strong topology (that is, via $\bar{\mathcal{K}}$).

Proposition 1 (*Reproducing Kernel*)**.** *Let $\mathcal{K}$ be an RKKS wiht $\mathcal{K} = \mathcal{H}_+ \ominus \mathcal{H}_-$. Then*

1. *$\mathcal{H}_+$ and $\mathcal{H}_-$ are RKHS (with kernels k_+ and k_-),*
2. *There is a unique symmetric $k(\boldsymbol{x}, \boldsymbol{x}')$ with $k(\boldsymbol{x}, \cdot) \in \mathcal{K}$ such that for all $\boldsymbol{f} \in \mathcal{K}$, $\langle \boldsymbol{f}, k(\boldsymbol{x}, \cdot) \rangle_{\mathcal{K}} = f(\boldsymbol{x})$,*
3. *$k = k_+ - k_-$.*

Having the definition of the indefinite kernel at hand, one can further develop the indefinite kernels in the hyperbolic space. Specifically, we propose two embedding functions for Lorentz model and Poincaré ball, constructing the adaptive indefinite kernels.

4.1. Indefinite Lorentz kernels

Given two points in the Lorentz model, *i.e.*, $\boldsymbol{l}_i, \boldsymbol{l}_j \in \mathbb{L}^n_c$, we first give the following embedding function:

$$F^{\mathbb{L}}_{\boldsymbol{l}_i}(\boldsymbol{l}_j) = \frac{\cosh^{-1}(-c\langle \boldsymbol{l}_i, \boldsymbol{l}_j \rangle_{\mathbb{L}})}{\sinh\big(\cosh^{-1}(-c\langle \boldsymbol{l}_i, \boldsymbol{l}_j \rangle_{\mathbb{L}})\big)} (\boldsymbol{v} + c\langle \boldsymbol{l}_i, \boldsymbol{l}_j \rangle_{\mathbb{L}} \boldsymbol{l}_i). \tag{11}$$

Eq. (11) realizes the mapping of projecting the point $\boldsymbol{l}_j$ to the tangent space at the point $\boldsymbol{l}_i$: $F^{\mathbb{L}}_{\boldsymbol{l}_i}(\boldsymbol{l}_j) = \log^{\mathbb{L}}_{\boldsymbol{l}_i}(\boldsymbol{l}_j)$. Having this embedding function at hand, it is ready to formulate the indefinite kernels in the Lorentz model.

4.1.1. Indefinite Lorentz Tangent kernel

We first formulate the simplest kernel, named indefinite Lorentz Tangent kernel (iLTk). Mathematically, the iLTk is written as:

$$k(\boldsymbol{l}_i, \boldsymbol{l}_j) = F^{\mathbb{L}}_{\boldsymbol{l}_i}(\boldsymbol{l}_j)^\top F^{\mathbb{L}}_{\boldsymbol{l}_j}(\boldsymbol{l}_i). \tag{12}$$

The proposed iLTk is interpreted as the inner product of two vectors in the tangent space. Optimizing over the iLTk can control the distance between $\boldsymbol{l}_i$ and $\boldsymbol{l}_j$ in the Lorentz model.

We further show the definiteness of the iLTk. In this study, we randomly generate 32 points in the Lorentz model (*e.g.*, $\mathbb{L}^4_{0.1}$) and calculate the Gram matrix for the iLTk. Then one can further calculate the number of positive eigenvalue, negative eigenvalue, and zero eigenvalue, reading as 11, 5, and 16, and the maximum contribution comes from the negative eigenvalue, and cannot be regarded as the noise. In other words, the proposed kernel is indeed an indefinite kernel.

4.1.2. Indefinite Lorentz RBF kernel

The radial basis function (RBF) kernel has been recognized as one of the most powerful kernels (Fang, Harandi et al., 2021a; Hofmann et al., 2008; Jayasumana, Hartley, Salzmann, Li, & Harandi, 2013). The two vectors in Fig. (12) further motivate us to develop a pseudo-RBF kernel, termed indefinite Lorentz RBF kernel (iLRk), which can be formulated as:

$$k(\boldsymbol{l}_i, \boldsymbol{l}_j) = \exp(-\sigma \| F^{\mathbb{L}}_{\boldsymbol{l}_i}(\boldsymbol{l}_j) - F^{\mathbb{L}}_{\boldsymbol{l}_j}(\boldsymbol{l}_i) \|^2), \tag{13}$$

for $\sigma > 0$. It worth noting that one cannot prove that the term $\| F^{\mathbb{L}}_{\boldsymbol{l}_i}(\boldsymbol{l}_j) - F^{\mathbb{L}}_{\boldsymbol{l}_j}(\boldsymbol{l}_i) \|^2$ is negative definite, making the proposed iLRk an indefinite kernel. However, the term $\| F^{\mathbb{L}}_{\boldsymbol{l}_i}(\boldsymbol{l}_j) - F^{\mathbb{L}}_{\boldsymbol{l}_j}(\boldsymbol{l}_i) \|^2$ can indicate the divergence of the vectors, and it can also be used to measure the distance between $\boldsymbol{l}_i$ and $\boldsymbol{l}_j$. That said, even though iLRk is an indefinite kernel, learning from iLRk follows a similar principle to the RBF kernel.

In the vanilla RBF kernel, *i.e.*, $k(\boldsymbol{x}, \boldsymbol{y}) = \exp\big(-\sigma \|\boldsymbol{x} - \boldsymbol{y}\|^2\big)$. A straightforward way to prove its definiteness can be realized by first proving the inner kernel, *i.e.*, $\|\boldsymbol{x} - \boldsymbol{y}\|^2$, being negative definite (nd). Then one can obtain a positive definite (pd) RBF kernel. However, the proposed iLRk is given by

$$k(\boldsymbol{l}_i, \boldsymbol{l}_j) = \exp\big(-\sigma \| F^{\mathbb{L}}_{\boldsymbol{l}_i}(\boldsymbol{l}_j) - F^{\mathbb{L}}_{\boldsymbol{l}_j}(\boldsymbol{l}_i) \|^2\big) \tag{14}$$

for $\boldsymbol{l}_i$, $\boldsymbol{l}_i \in \mathbb{L}^n_c$. In iLRk, the inner kernel, *i.e.*, $\| F^{\mathbb{L}}_{\boldsymbol{l}_i}(\boldsymbol{l}_j) - F^{\mathbb{L}}_{\boldsymbol{l}_j}(\boldsymbol{l}_i) \|^2$ is an indefinite kernel, thereby making the iLRK being an indefinite kernel.

4.1.3. Indefinite Lorentz Laplace kernel

The RBF kernel can attain superior performance since it holds the universal approximation property. Another popular kernel with such a property is known as a Laplace kernel. Since the Laplace kernel can be formulated by a minor modification to the RBF kernel, we can derive the indefinite Lorentz Laplace kernel (iLLk), as follows:

$$k(\boldsymbol{l}_i, \boldsymbol{l}_j) = \exp(-\sigma \| F^{\mathbb{L}}_{\boldsymbol{l}_i}(\boldsymbol{l}_j) - F^{\mathbb{L}}_{\boldsymbol{l}_j}(\boldsymbol{l}_i) \|) \tag{15}$$

for $\sigma > 0$. Similar to the iLRk, the term $\| F^{\mathbb{L}}_{\boldsymbol{l}_i}(\boldsymbol{l}_j) - F^{\mathbb{L}}_{\boldsymbol{l}_j}(\boldsymbol{l}_i) \|$ in iLLk is not negative definite, such that iLLk is an indefinite kernel. The proposed iLLk is also an indefinite kernel, which can be proven similar to the iLRk.

4.2. Indefinite Poincaré kernels

In the above, we develop three indefinite kernels in the Lorentz model. Since the Poincaré ball and Lorentz model can be translated isometrically by Eq. (10) and Eq. (9), we can further define the variants in the Poincaré ball. Given two points $\boldsymbol{p}_i, \boldsymbol{p}_j \in \mathbb{D}^n_c$, we can propose the embedding function as follows:

$$F^{\mathbb{D}}_{\boldsymbol{p}_i}(\boldsymbol{p}_j) = \frac{2}{\sqrt{c}\lambda_c(\boldsymbol{p}_i)} \tanh^{-1}(\sqrt{c} \| -\boldsymbol{p}_i \oplus_c \boldsymbol{p}_j \|) \frac{-\boldsymbol{p}_i \oplus_c \boldsymbol{p}_j}{\| -\boldsymbol{p}_i \oplus_c \boldsymbol{p}_j \|}, \tag{16}$$

where it realizes the mapping of project the point $\boldsymbol{p}_j$ to the tangent space at the point $\boldsymbol{p}_i$: $F^{\mathbb{D}}_{\boldsymbol{p}_i}(\boldsymbol{p}_j) = \log^{\mathbb{D}}_{\boldsymbol{p}_i}(\boldsymbol{p}_j)$. Then we can formulate the indefinite Poincaré kernels.

4.2.1. Indefinite Poincaré Tangent kernel

In the Poincaré ball, one can follow the Eq. (12) to formulate the indefinite Poincaré Tangent kernel (iPTk), given by:

$$k(\boldsymbol{p}_i, \boldsymbol{p}_j) = F^{\mathbb{D}}_{\boldsymbol{p}_i}(\boldsymbol{p}_j)^\top F^{\mathbb{D}}_{\boldsymbol{p}_j}(\boldsymbol{p}_i). \tag{17}$$

As illustrated in Eq. (17), iPTk realizes the similarity measure for two vectors in the tangent space.

4.2.2. Indefinite Poincaré RBF kernel

One can further formulate the indefinite Poincaré RBF kernel (iPRk), as follows:

$$k(\boldsymbol{p}_i, \boldsymbol{p}_j) = \exp(-\sigma \| F^{\mathbb{D}}_{\boldsymbol{p}_i}(\boldsymbol{p}_j) - F^{\mathbb{D}}_{\boldsymbol{p}_j}(\boldsymbol{p}_i)\|^2). \tag{18}$$

for $\gamma > 0$. The iPRk can also be understood as measuring the similarity between two vectors in tangent space as a result of calculating the radial value between them.

4.2.3. Indefinite Poincaré Laplace kernel

The indefinite Poincaré Laplace kernel (iPLk) can also be derived easily, and its formulation is given by:

$$k(\boldsymbol{p}_i, \boldsymbol{p}_j) = \exp(-\sigma \| F^{\mathbb{D}}_{\boldsymbol{p}_i}(\boldsymbol{p}_j) - F^{\mathbb{D}}_{\boldsymbol{p}_j}(\boldsymbol{p}_i)\|) \tag{19}$$

for $\gamma > 0$.

4.3. Why the embedding functions?

In Sections 4.1 and 4.2, we propose two embedding functions, *i.e.*, $F^{\mathbb{L}}_{\boldsymbol{l}_i}(\boldsymbol{l}_j)$ and $F^{\mathbb{D}}_{\boldsymbol{p}_i}(\boldsymbol{p}_j)$, as a bridge to develop the indefinite kernels. Since the Poincaré ball and the Lorentz model are isometric, we use the embedding function in Poincaré ball to explain the intention. Given two points $\boldsymbol{p}_i, \boldsymbol{p}_j \in \mathbb{D}^n_c$, the geodesic distance can be obtained as

$$d^{\mathbb{D}}_c(\boldsymbol{p}_i, \boldsymbol{p}_j) = \frac{2}{\sqrt{c}} \tanh^{-1}(\sqrt{c} \| -\boldsymbol{p}_i \oplus_c \boldsymbol{p}_j \|). \tag{20}$$

For the given embedding function, $F^{\mathbb{D}}_{\boldsymbol{q}_i}(\boldsymbol{q}_j)$, its 2-norm can be calculated as

$$\| F^{\mathbb{D}}_{\boldsymbol{p}_i}(\boldsymbol{p}_j)\|^2 = \frac{4}{c\lambda^2_c(\boldsymbol{p}_i)} \left(\tanh^{-1}(\sqrt{c} \| -\boldsymbol{p}_i \oplus_c \boldsymbol{p}_j \|)\right)^2. \tag{21}$$

By observing Eqs. (20) and (21), it holds that:

$$d^{\mathbb{D}}_c(\boldsymbol{p}_i, \boldsymbol{p}_j) = \lambda_c(\boldsymbol{p}_i) \| F^{\mathbb{D}}_{\boldsymbol{p}_i}(\boldsymbol{p}_j)\|. \tag{22}$$

It also hold that $d^{\mathbb{D}}_c(\boldsymbol{p}_i, \boldsymbol{p}_j) = \lambda_c(\boldsymbol{p}_j) \| F^{\mathbb{D}}_{\boldsymbol{p}_j}(\boldsymbol{p}_i)\|$ via using $F^{\mathbb{D}}_{\boldsymbol{p}_j}(\boldsymbol{p}_i)$. To see this, the norm value of the projected embedding is equivalent to the geodesic distance, and the optimization is indeed posed to the metric of points, making the learning procedure more flexible. Specifically, optimization over the kernels makes the embedding function adaptive to the local relations of samples. This is different from the existing works (Fang, Harandi, Lan et al., 2023a; Fang, Harandi et al., 2021a), where the kernels are defined by a fixed embedding function. In other words, the proposed adaptive embedding function can model the local relationship between samples while the fixed one only reveals the global relationships of samples.

4.4. Optimization

In DNNs, we usually employ the kernels as the similarity measure and plug it in the object function, as Fang, Harandi, Lan et al. (2023a), Jayasumana, Ramalingam, and Kumar (2021). Then one should perform the optimization over the objective function, updating the parameters of the network. Instead of minimizing the objective function in the RKHS, the stabilization point of the objective function is required in RKKS (Ong et al., 2004). Specifically, given a sample $\boldsymbol{X}_i$ in a batch, a neural network f, parameterized by θ, first encodes the input data and the proposed indefinite kernels project the embedding to RKKS. The optimization problem is realized by the following:

$$\theta^* = \text{stabilize}_{k(f(\boldsymbol{X}_i|\theta))\in\mathcal{K}} \mathcal{L}\{k(f(\boldsymbol{X}_i|\theta))\}. \tag{23}$$

Theoretically, the optimization over indefinite kernels is the result of stabilization to the loss function, proven in Ong et al. (2004). However, in practice, we adopt the classical minimization scheme to update the parameters in the neural network.

Table 1
Image classification result on CIFAR-10, CIFAR-100 and ImageNet datasets on the ResNet-18 backbone.

Model	Backbone	CIFAR-10	CIFAR-100	ImageNet
Baseline[a]	ResNet-18	90.12 ± 0.15	73.15 ± 0.12	68.58 ± 0.08
Indefinite Lorentz kernels				
iLTk	ResNet-18	91.26 ± 0.11	74.29 ± 0.15	68.77 ± 0.10
iLRk	ResNet-18	**91.98 ± 0.10**	**74.70 ± 0.12**	**69.12 ± 0.09**
iLLk	ResNet-18	91.74 ± 0.15	74.62 ± 0.10	68.96 ± 0.09
Indefinite Poincaré kernels				
iPTk	ResNet-18	91.31 ± 0.13	74.10 ± 0.11	68.82 ± 0.12
iPRk	ResNet-18	91.88 ± 0.15	**74.61 ± 0.11**	69.10 ± 0.13
iPLk	ResNet-18	**92.20 ± 0.07**	74.58 ± 0.12	**69.21 ± 0.10**

[a] Indicates that we use the Hyperbolic Classifier as the baseline.

Table 2
Image classification result on CIFAR-10, CIFAR-100 and ImageNet datasets on the ResNet-50 backbone.

Model	Backbone	CIFAR-10	CIFAR-100	ImageNet
Baseline[a]	ResNet-50	93.12 ± 0.09	74.86 ± 0.12	69.32 ± 0.09
Indefinite Lorentz kernels				
iLTk	ResNet-50	93.29 ± 0.10	75.40 ± 0.12	69.71 ± 0.10
iLRk	ResNet-50	93.96 ± 0.10	75.62 ± 0.12	70.10 ± 0.09
iLLk	ResNet-50	**94.08 ± 0.10**	**75.88 ± 0.11**	**70.16 ± 0.09**
Indefinite Poincaré kernels				
iPTk	ResNet-50	93.89 ± 0.13	75.60 ± 0.10	69.61 ± 0.09
iPRk	ResNet-50	94.60 ± 0.10	75.98 ± 0.15	**70.28 ± 0.11**
iPLk	ResNet-50	**94.73 ± 0.09**	**76.10 ± 0.12**	70.22 ± 0.11

[a] Indicates that we use the Hyperbolic Classifier as the baseline.

5. Experiment

This section presents the experimental evaluation of the proposed kernels. In this study, the proposed kernels act as a similarity measurement, plugged seamlessly into the objective function. Various learning paradigms, jointly optimizing the deep networks and the kernels, are adopted to demonstrate the superiority and generalization of the indefinite kernels. In this regard, the main purpose of this study is to verify the benefits of indefinite kernels in the deep learning framework.

5.1. Image classification

5.1.1. Setting

We first report the results of the image classification task. As the most fundamental learning task, the image classification task learns a classifier to predict the class for a given sample. Usually, it consists of two components, one backbone network to extract feature representation, and another classification layer to make the prediction. In this regard, we employ the ResNet-18 and ResNet-50 (He, Zhang, Ren, & Sun, 2016) as the backbone network. The backbone network and classifier are optimized using cross-entropy loss. We integrate the proposed kernels into the loss function as a method of measuring similarity. Three real world image benchmarks, *i.e.*, CIFAR-10, CIFAR-100 (Krizhevsky, 2009) and ImageNet (Deng, Dong, Li, Li, & Li, 2009), are used in the experiment. Both CIFAR-10 and CIFAR-100 have 50,000 images for training and 10,000 images for evaluation. CIFAR-10 contains 10 classes, with each containing 5000 samples, while CIFAR-100 contains 100 classes, and each class has 500 samples. ImageNet, also known as ILSVRC, is a large-scale dataset for the image classification task. It contains 1000 classes. The training images and validation images are 1.28 million and 50 K. We report the Top-1 accuracy across three benchmarks. In this study, the Hyperbolic Classifier (Ganea et al., 2018) is used as the baseline.

Table 3
Few-shot classification results on the Conv-4 backbone with 95% confidence interval.

Model	Backbone	*tiered*-ImageNet		FC100	
		5-way 1-shot	5-way 5-shot	5-way 1-shot	5-way 5-shot
Baseline[a]	Conv-4	54.44 ± 0.23	71.96 ± 0.20	37.59 ± 0.19	51.76 ± 0.19
Indefinite Lorentz kernels					
iLTk	Conv-4	56.42 ± 0.22	74.58 ± 0.20	38.21 ± 0.19	52.64 ± 0.19
iLRk	Conv-4	**58.69 ± 0.21**	**77.02 ± 0.18**	**39.12 ± 0.17**	**55.10 ± 0.18**
iLLk	Conv-4	58.38 ± 0.21	76.61 ± 0.18	38.96 ± 0.17	54.76 ± 0.18
Indefinite Poincaré kernels					
iPTk	Conv-4	57.62 ± 0.20	75.42 ± 0.20	38.43 ± 0.18	53.61 ± 0.19
iPRk	Conv-4	**59.12 ± 0.20**	**77.24 ± 0.18**	**39.41 ± 0.18**	**55.18 ± 0.18**
iPLk	Conv-4	58.64 ± 0.20	76.48 ± 0.20	38.94 ± 0.18	54.76 ± 0.18

[a] Indicates that we use the Hyperbolic ProtoNet as the baseline.

5.1.2. Result

Tables 1 and 2 illustrate the classification accuracy across three datasets on two backbone networks. It shows that the proposed indefinite kernels improve the visual recognition performance for the baselines. In the ResNet-18 backbone, the best indefinite Lorentz kernel, *i.e.*, iLRk, improves the baseline by 1.86%, 1.55% and 0.54%, respectively, and the best indefinite Poincaré kernel, *i.e.*, iPLk, improves the baseline by 2.08%, 1.43% and 0.63% respectively. While in the ResNet-50 backbone, the best indefinite Lorentz kernel, *i.e.*, iLLk, improves the baseline by 0.96%, 1.02% and 0.84%, respectively, and the best indefinite Poincaré kernel, *i.e.*, iPLk, improves the baseline by 1.61%, 1.24% and 0.90% respectively.

This study of image classification on both backbone networks indicates that (1)-the Laplace kernel attains the overall best performance among the proposed kernels, and (2)-the indefinite Poincaré kernel is superior to the indefinite Lorentz kernel. To statistically evaluate the performance rankings, we employ the Friedman test (Demšar, 2006), where the iPLk is consistently ranked first among all proposals. All the studies above align with the observation that the CV community prefers to employ the Poincaré ball as the model of hyperbolic space.

5.2. Few-shot learning

5.2.1. Setting

Few-shot learning (FSL) aims to train a neural network that can realize the fast adaptation to unknown tasks, given only a few samples of each task (Snell, Swersky, & Zemel, 2017; Sung et al., 2018; Vinyals, Blundell, Lillicrap, Kavukcuoglu, & Wierstra, 2016). The recent advance enjoys the paradigm of episode learning to train a meta learner. Specifically, this paradigm is known as N-way K-shot classification, where each episode consists of N classes of sample with each class having K samples. In this study, we follow the general practice (*i.e.*, 5-way 1-shot and 5-way 5-shot) to evaluate the model and adopt the prototypical network (ProtoNet) (Snell et al., 2017) to train the neural network. Following the practice in Fang, Harandi et al. (2021a), we plug the proposed kernels in the loss function to optimize the network. Two commonly used neural networks, *i.e.*, Conv-4 (Snell et al., 2017) and ResNet-18 (He et al., 2016), are used as backbones, and the comparison is made across two popular FSL benchmarks, *i.e.*, *tiered*-ImageNet (Ren et al., 2018) and Few-shot-CIFAR100 (FC100) (Oreshkin, Rodríguez López, & Lacoste, 2018). *tiered*-ImageNet is a subset of ImageNet with broader classes (*i.e.*, 608 classes in total). The *tiered*-ImageNet contains 351 classes for training, 97 classes for validation, and 160 classes for testing. FC100, which is based on the CIFAR-100, is proposed for the FSL task recently. It also contains three data splits, *i.e.* training split, validation split, and test split, with each having 60, 20, and 20 classes, respectively. In this study, the Hyperbolic ProtoNet (Khrulkov et al., 2020) is employed as the baseline.

5.2.2. Result

The results in Tables 3 and 4 show that both indefinite Lorentz kernels and indefinite Poincaré kernels can bring performance gain over two baselines, and the iLRk and iPRk achieve the consist best performance across two backbones. Specifically, in the *tiered*-ImageNet, iLRk outperforms the baseline by 4.25%/5.06% over the Conv-4 backbone and 3.80%/3.32% over the ResNet-18 backbone, and in the FC100, iLRk outperforms the baseline 1.53%/3.34% over the Conv-4 backbone and 3.96%/4.52% over the ResNet-18 backbone, w.r.t. the 5-way 1-shot/5-way 5-shot learning scenarios. While for the best indefinite Poincaré kernel, *i.e.*, iPRk, it attains the performance gain by 4.68%/5.28% over the Conv-4 backbone and 4.10%/3.30% over the ResNet-18 backbone w.r.t. the 5-way 1-shot/5-way 5-shot in the *tiered*-ImageNet. In the FC100, iLRk outperforms the baseline 1.82%/3.42% over the Conv-4 backbone and 4.50%/4.4.68% over the ResNet-18 backbone, w.r.t. the 5-way 1-shot/5-way 5-shot learning scenarios.

In addition to the performance gain, the similar observation that the indefinite Poincaré kernels are superior to the indefinite Lorentz kernels is also made in this study.

5.3. Zero-shot learning

5.3.1. Setting

Zero-shot learning (ZSL) aims to identify objects that are unseen during the training phase (Akata, Perronnin, Harchaoui, & Schmid, 2015; Xian et al., 2016). The training paradigm of ZSL can be formulated as a cross-modality matching problem. Specifically, the training stage aligns the embedding space for the visual feature and semantic feature, and such aligned embedding space can be used to group the unseen visual feature and semantic feature. In this study, we begin by projecting the visual and semantic embeddings into hyperbolic space. Following the approach outlined in Fang, Harandi et al. (2021a), we align these embedding spaces using the proposed kernels. Four datasets, *i.e.*, SUN (Patterson & Hays, 2012), CUB (Wah, Branson, Welinder, Perona, & Belongie, 2011), AWA1 (Lampert, Nickisch, & Harmeling, 2013) and AWA2 (Akata et al., 2015) are adopted to evaluate our algorithm. The visual features of all datasets are extracted from the ImageNet pre-trained ResNet-101 and the dimensions are 2048. SUN is a fine-grained dataset and contains 717 classes with 14,340 images in total. Those 717 classes are annotated with 102 attributes. CUB, another fine-grained dataset, contains 11,788 images of 200 different species of birds, annotated with 312 attributes. The AWA1 is a coarse-grained dataset with animal images. It has 30,475 images with 50 classes, which are annotated by 85 attributes. Similar to AWA1, AWA2 consists of 37,322 images with the same animal classes and attributes as AWA1. We report the top-1 mean class accuracy for both the unseen classes (U) and the seen classes (S). The harmonic mean (HM) score is also used to calculate the overall performance. In this study, we follow the practice of cross-entropy alignment in Fang, Zhou, Roy, Petersson, and Harandi (2019) as the baseline.

Table 4
Few-shot classification results on the ResNet-18 backbone with 95% confidence interval.

Model	Backbone	*tiered*-ImageNet		FC100	
		5-way 1-shot	5-way 5-shot	5-way 1-shot	5-way 5-shot
Baseline[a]	ResNet-18	62.28 ± 0.23	74.50 ± 0.21	40.64 ± 0.20	52.50 ± 0.30
Indefinite Lorentz kernels					
iLTk	ResNet-18	64.12 ± 0.23	75.63 ± 0.21	42.94 ± 0.20	55.74 ± 0.30
iLRk	ResNet-18	**66.08 ± 0.22**	**77.82 ± 0.20**	**44.60 ± 0.20**	**57.02 ± 0.27**
iLLk	ResNet-18	65.43 ± 0.22	76.21 ± 0.20	43.08 ± 0.20	56.17 ± 0.30
Indefinite Poincaré kernels					
iPTk	ResNet-18	64.36 ± 0.22	75.72 ± 0.21	42.84 ± 0.21	55.60 ± 0.30
iPRk	ResNet-18	**66.38 ± 0.22**	**77.80 ± 0.20**	**45.14 ± 0.20**	**57.18 ± 0.26**
iPLk	ResNet-18	65.38 ± 0.22	77.36 ± 0.20	44.76 ± 0.21	56.89 ± 0.30

[a] Indicates that we use the Hyperbolic ProtoNet as the baseline.

Table 5
Zero-shot recognition on SUN, CUB, AWA1 and AWA2 datasets.

Model	SUN			CUB			AWA1			AWA2		
	U	S	HM	U	S	HM	U	S	HM	U	S	HM
Baseline[a]	22.8 ± 1.1	38.0 ± 1.4	28.5 ± 1.3	18.6 ± 1.4	44.6 ± 1.8	26.3 ± 1.6	29.8 ± 1.2	76.4 ± 2.6	42.9 ± 1.8	25.5 ± 1.5	76.4 ± 2.1	38.2 ± 1.7
Indefinite Lorentz kernels												
iLTk	35.4 ± 2.3	42.1 ± 1.5	38.5 ± 1.9	40.8 ± 1.6	56.7 ± 2.4	47.5 ± 1.9	52.8 ± 1.1	84.9 ± 2.4	65.1 ± 1.5	36.8 ± 1.7	87.6 ± 1.9	51.8 ± 1.8
iLRk	36.1 ± 1.8	**44.5 ± 1.3**	39.9 ± 1.5	**44.2 ± 1.0**	55.6 ± 1.6	49.2 ± 1.2	57.4 ± 1.6	80.9 ± 1.8	67.2 ± 1.6	47.3 ± 1.4	86.4 ± 2.0	61.1 ± 1.6
iLLk	**37.6 ± 1.6**	44.3 ± 1.1	**40.7 ± 1.3**	43.7 ± 1.2	**56.8 ± 1.8**	**49.4 ± 1.4**	**62.8 ± 2.1**	**88.4 ± 2.4**	**73.4 ± 2.2**	**50.9 ± 1.6**	**91.2 ± 2.1**	**65.3 ± 1.8**
Indefinite Poincaré kernels												
iPTk	34.8 ± 0.9	43.3 ± 1.2	38.6 ± 1.0	41.3 ± 1.1	58.1 ± 1.4	48.3 ± 1.2	53.4 ± 1.4	84.6 ± 1.6	65.5 ± 1.5	38.2 ± 1.2	87.4 ± 1.8	53.2 ± 1.4
iPRk	37.1 ± 1.4	44.9 ± 1.2	40.6 ± 1.3	43.8 ± 0.8	57.4 ± 1.2	49.7 ± 1.0	59.4 ± 1.2	**86.0 ± 1.7**	70.3 ± 1.4	44.2 ± 1.0	**90.4 ± 1.6**	59.4 ± 1.2
iPLk	**38.4 ± 1.0**	**45.1 ± 1.2**	**41.5 ± 1.1**	**45.6 ± 1.4**	**58.2 ± 0.9**	**51.1 ± 1.1**	**60.8 ± 1.4**	85.3 ± 1.3	**71.0 ± 1.3**	**52.6 ± 2.4**	90.1 ± 1.1	**66.4 ± 1.5**

[a] Indicates that we follow the cross-entropy alignment as the baseline.

Table 6
Person re-ID results on Market-1501 and DukeMTMC-reID datasets with the embedding dimension of #32. The value of "_" indicates the result below the baseline network.

Model	Dim	Market-1501		DukeMTMC-reID	
		R-1	mAP	R-1	mAP
Baseline[a]	#32	75.9 ± 0.34	51.9 ± 0.30	62.2 ± 0.28	39.1 ± 0.21
Indefinite Lorentz kernels					
iLTk	#32	75.2 ± 0.40	50.6 ± 0.34	63.7 ± 0.25	40.9 ± 0.30
iLRk	#32	76.4 ± 0.33	54.8 ± 0.26	**66.4 ± 0.24**	**45.8 ± 0.37**
iLLk	#32	**78.6 ± 0.22**	**56.4 ± 0.21**	66.2 ± 0.20	44.7 ± 0.24
Indefinite Poincaré kernels					
iPTk	#32	75.6 ± 0.46	50.3 ± 0.42	62.7 ± 0.37	39.8 ± 0.32
iPRk	#32	77.9 ± 0.18	53.9 ± 0.26	**67.2 ± 0.12**	**45.4 ± 0.19**
iPLk	#32	**78.8 ± 0.27**	**56.9 ± 0.32**	66.8 ± 0.29	45.2 ± 0.20

[a] Indicates that we use the hyperbolic counterpart of the re-ID network as the baseline.

Table 7
Person re-ID results on Market-1501 and DukeMTMC-reID datasets with the embedding dimension of #64. The value of "_" indicates the result below the baseline network.

Model	Dim	Market-1501		DukeMTMC-reID	
		R-1	mAP	R-1	mAP
Baseline[a]	#64	84.4 ± 0.41	62.7 ± 0.35	70.8 ± 0.40	48.6 ± 0.36
Indefinite Lorentz kernels					
iLTk	#64	83.7 ± 0.62	61.1 ± 0.49	71.2 ± 0.53	48.7 ± 0.46
iLRk	#64	84.6 ± 0.31	65.8 ± 0.26	**74.6 ± 0.37**	**56.7 ± 0.23**
iLLk	#64	**85.9 ± 0.35**	**68.2 ± 0.30**	74.3 ± 0.42	56.6 ± 0.34
Indefinite Poincaré kernels					
iPTk	#64	84.6 ± 0.53	63.1 ± 0.42	71.5 ± 0.46	49.0 ± 0.39
iPRk	#64	85.6 ± 0.29	69.1 ± 0.24	74.9 ± 0.30	56.4 ± 0.19
iPLk	#64	**86.4 ± 0.31**	**69.3 ± 0.24**	**75.2 ± 0.38**	**56.9 ± 0.26**

[a] Indicates that we use the hyperbolic counterpart of the re-ID network as the baseline.

5.3.2. Result

Table 5 shows the results of the zero-shot learning scenarios. It holds similar observations that indefinite Poincaré kernels outperform the indefinite Lorentz kernels on the SUN, CUB, and AWA2 datasets while the indefinite Lorentz kernels attain better results on the AWA1 dataset. The best Poincaré kernel, *i.e.*, iPLk, improves the HM value by 13.0, 24.8, and 28.2 on the SUN, CUB, and AWA2 datasets respectively. On the AWA1 dataset, the maximum improvement of the HM value reads as 30.5, achieved by the iLLk. Additionally, we observe that the indefinite Laplace kernel outperforms other variants, showing its superiority in the task of ZSL.

5.4. Person re-identification

5.4.1. Setting

Person re-identification (re-ID) is an important application in the video/multi-camera surveillance task (Fang et al., 2021; Ye et al., 2021). This task is characterized as: given a query person of interest, the machine is required to retrieve correct person images from a gallery dataset. This is realized by learning an embedding space, where the intra- (or inter-) person variance is minimized (or maximized). The learning process for re-identification tasks is commonly structured as a classification problem, employing neural networks trained with cross-entropy loss. During inference, only the representation layer is active. The proposed kernels are utilized during training, enhancing the network's ability to produce more discriminative representations for person images. Following the work (Khrulkov et al., 2020), ResNet-50, pre-trained on ImageNet, is employed as a backbone network and we also perform experiments across three dimensions, *i.e.*, 32, 64, 128, for the feature representation. Both Market-1501 (Zheng et al., 2015) and DukeMTMC-reID (Ristani, Solera, Zou, Cucchiara, & Tomasi, 2016) pedestrian datasets are used to evaluate our approaches. The Market-1501 dataset consists of 32,668 pedestrian images, captured by 6 disjoint cameras. The person bounding boxes are detected automatically by DPM (Felzenszwalb, Girshick, McAllester, & Ramanan, 2010). This dataset is split into 12,936 images of 751 identities for

Table 8
Person re-ID results on Market-1501 and DukeMTMC-reID datasets with the embedding dimension of #128. The value of "-" indicates the result below the baseline network.

Model	Dim	Market-1501		DukeMTMC-reID	
		R-1	mAP	R-1	mAP
Baseline[a]	#128	87.8 ± 0.26	68.4 ± 0.37	76.5 ± 0.30	55.4 ± 0.32
Indefinite Lorentz kernels					
iLTk	#128	86.4 ± 0.30	68.1 ± 0.42	75.9 ± 0.22	55.5 ± 0.38
iLRk	#128	88.4 ± 0.31	69.9 ± 0.32	**78.4 ± 0.24**	**63.2 ± 0.42**
iLLk	#128	**89.2 ± 0.28**	**72.4 ± 0.30**	77.4 ± 0.28	60.7 ± 0.45
Indefinite Poincaré kernels					
iPTk	#128	87.1 ± 0.30	68.2 ± 0.18	77.2 ± 0.20	57.0 ± 0.16
iPRk	#128	88.7 ± 0.30	70.6 ± 0.16	78.9 ± 0.21	61.4 ± 0.12
iPLk	#128	**89.6 ± 0.32**	**72.3 ± 0.18**	**79.6 ± 0.22**	**64.1 ± 0.14**

[a] Indicates that we use the hyperbolic counterpart of the re-ID network as the baseline.

training and 19,732 of 750 identities for testing. DukeMTMC-reID is collected by 8 non-overlapped cameras and the person bounding boxes are manually annotated. Following the standard training protocol, this dataset is divided into 16,522 and 19,889 images for training and testing, respectively. We use both mean average precision (mAP) and rank-1 accuracy of cumulative matching characteristics (CMC) to evaluate our algorithms. In this study, the hyperbolic counterpart of the re-ID network (Khrulkov et al., 2020) is set as the baseline, and we evaluate the effectiveness of the proposed kernels on three embedding dimensions, *e.g.*, #32, #64, and #128.

5.4.2. Result

Tables 6, 7, and 8 show the empirical study in the re-ID task, varying the embedding dimensions. From those three tables, one can first observe that the proposed kernels can bring performance gain over three embedding dimensions and the indefinite Poincaré kernels outperform the indefinite Lorentz kernels. Specifically, in #32, the Laplace kernels (*i.e.*, iLLk and iPLk) and the RBF kernels (*i.e.*, iLRk and iPRk) attain the overall best performance on the Market-1501 dataset and the DukeMTMC-reID dataset, respectively. While in the #64 and #128, the iPLk illustrates its superiority across two datasets.

5.5. Knowledge distillation

5.5.1. Setting

Knowledge distillation (KD) (Hinton, Vinyals, & Dean, 2014) is a useful learning paradigm to compress the neural network via transferring the knowledge from a pre-trained larger teacher model to a lightweight student model, such that the knowledge can be compressed in the lightweight student model. This paradigm is also extended to other model adaptation tasks, including continual learning (Li & Holem, 2018). In this task, the optimization process is required to minimize the KL divergence of the logits from teacher network and student network. In this regard, we use the kernels to produce the logits of student network and integrate it in the loss function. We report the results on the CIFAR-10 and CIFAR-100 datasets (Krizhevsky, 2009). The details of those two datasets are in Section 5.1. We use the ResNet-20 as a teacher network and a simple 4-layer CNN as a student network. In regard to the KD method, we follow the logit distillation method proposed in Hinton et al. (2014).

5.5.2. Result

The evaluation results in Table 9 again show that the proposed indefinite kernels achieve a better distillation performance. The best indefinite Lorentz kernels and indefinite Poincaré kernels, *i.e.*, iLRk and iPRk, improve the baseline by 3.0%/4.7% and 4.1%/6.2% on the CIFAR-10/CIFAR-100 datasets, respectively. In addition, this study also

Table 9
Knowledge distillation resutls on CIFAR-10/100 datasets.

Model	CIFAR-10	CIFAR-100
Baseline[a]	80.5 ± 0.3	49.9 ± 0.4
Indefinite Lorentz kernels		
iLTk	81.9 ± 0.3	51.8 ± 0.4
iLRk	**83.5 ± 0.1**	54.6 ± 0.2
iLLk	83.1 ± 0.2	**54.7 ± 0.2**
Indefinite Poincaré kernels		
iPTk	82.7 ± 0.2	53.0 ± 0.2
iPRk	**84.6 ± 0.3**	**56.1 ± 0.4**
iPLk	83.8 ± 0.3	55.4 ± 0.3

[a] Indicates that we use the logit distillation method as the baseline.

verifies that the Poincaré ball is a better alternative than the Lorentz model when modeling the hyperbolic space for the visual data. All the observation above indicates that the indefinite kernels developed in the hyperbolic space can improve the representation capacity of the network and can be extended to other challenging vision tasks.

5.6. Further analysis

The kernel methods have been studied recently in the context of CV applications. The works in Fang, Harandi et al. (2021a) and Fang, Harandi, Lan et al. (2023a) propose a series of pd kernels, and the proposed pd kernels can improve the representation power of DNNs. This manuscript develops some indefinite kernels, on the Lorentz model and Poincaré ball, and the initial developments include an indefinite counterpart of the tangent kernel, RBF kernel, and Laplace kernel. In this study, we conduct an empirical study to verify the potential of the indefinite kernels by comparing them to the pd kernels.

5.6.1. Comparison to the PD kernels

We first compare the pd kernels and indefinite kernels on three variants, *e.g.*, tangent kernel, RBF kernel, and Laplace kernel. This study is conducted thoroughly on image classification, few-shot learning, zero-shot learning, and knowledge-distillation tasks. From Fig. 1, one can observe that the proposed indefinite kernels can almost consistently outperform the pd hyperbolic kernels, vividly showing the superiority of the proposal.

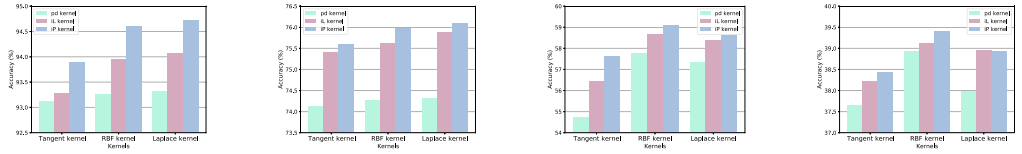
In the AWA1 for zero-shot learning (Fig. 1(f)) and CIFAR-10 for knowledge distillation (Fig. 1(g)) tasks, even though indefinite Lorentz kernels are inferior to the pd kernels, the indefinite Poincaré kernels can outperform the pd kernels considerably, again showing the good practice of the indefinite kernels proposed in this paper.

5.6.2. Comparison to the multiple kernel learning scheme

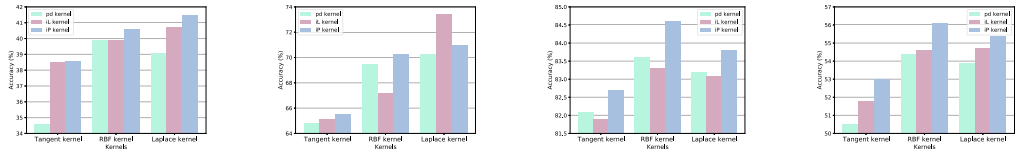
In Fang, Harandi, Lan et al. (2023a), the multiple kernel learning scheme is proposed in the hyperbolic space, named Poincaré radial kernel, as one of the best kernel methods in the hyperbolic space. To verify the effectiveness of the indefinite kernels, we then compare the performance of the Poincaré radial kernel to the proposed indefinite kernels on image classification, few-shot learning, zero-shot learning, and knowledge-distillation tasks. As suggested in Fig. 2, one can observe that both the best indefinite Lorentz kernel and the best indefinite Poincaré kernel proposed in this manuscript can outperform the Poincaré radial kernel, again showing the superiority of the indefinite kernels.

5.6.3. Analysis on the curvature of hyperbolic space

The above studies show that the proposed indefinite kernels can improve the representation power of the DNNs and achieve superior performance than the pd hyperbolic kernels, even suppressing the advanced multiple kernel learning. In this study, we evaluate the

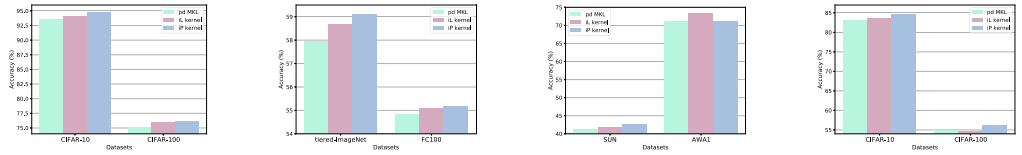


(a) Classification (CIFAR-10) (b) Classification (CIFAR-100) (c) FSL (*tiered*-ImageNet) (d) FSL (FC100)



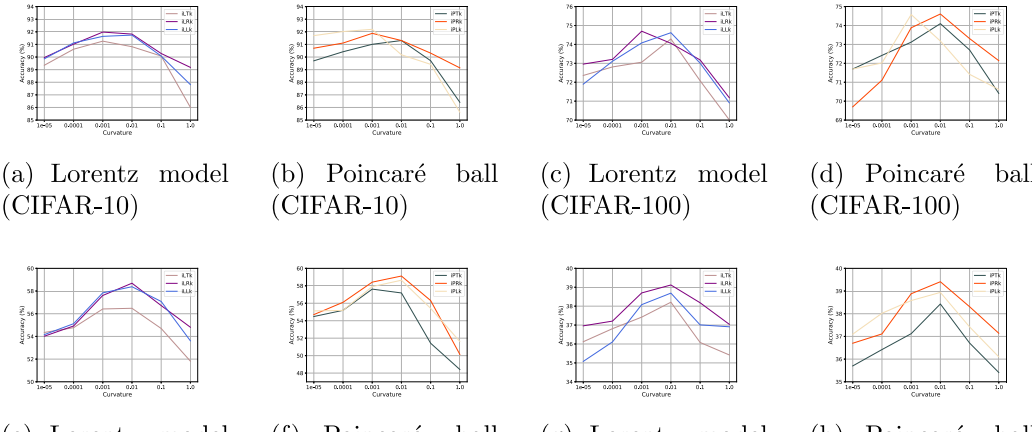
(e) ZSL (SUN) (f) ZSL (AWA1) (g) KD (CIFAR-10) (h) KD (CIFAR-100)

Fig. 1. Comparison to the pd kernels on image classification, few-shot learning, zero-shot learning and knowledge distillation tasks.



(a) Classification (b) FSL (c) ZSL (d) KD

Fig. 2. Comparison to the multiple kernel learning scheme on image classification, few-shot learning, zero-shot learning and knowledge distillation tasks.



(a) Lorentz model (CIFAR-10) (b) Poincaré ball (CIFAR-10) (c) Lorentz model (CIFAR-100) (d) Poincaré ball (CIFAR-100)

(e) Lorentz model (*tiered*-ImageNet) (f) Poincaré ball (*tiered*-ImageNet) (g) Lorentz model (FC100) (h) Poincaré ball (FC100)

Fig. 3. Analysis on the curvature of hyperbolic space for both Lorentz model and Poincaré ball.

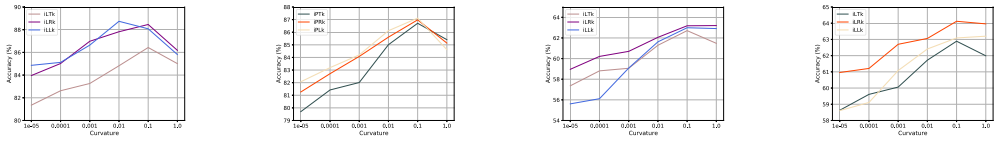
effectiveness of the curvature for both the Lorentz model and Poincaré ball, as shown in Fig. 3.

This analysis is performed on the image classification and few-shot learning tasks. In the image classification task, we evaluate the curvature on both CIFAR-10 and CIFAR-100 datasets (see Figs. 3(a)–3(d)). It indicates that in both the Lorentz model and Poincaré ball, the best performance of the indefinite kernels can be achieved when the curvature is around 0.001–0.01. In the few-shot learning task, the study is performed on the *tiered*-ImageNet and FC100 datasets, as shown in Figs. 3(e)–3(h). It also reveals that the proposed indefinite kernels attain the best performance when the curvature value is around 0.001–0.01.

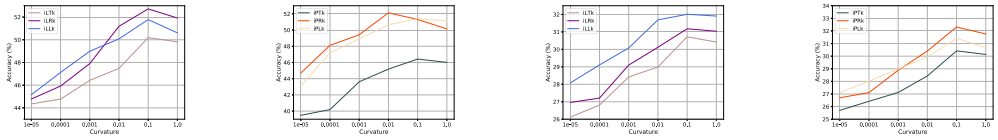
The analysis above poses a similar observation. However, it raises a question of when a larger curvature value with the proposed indefinite kernels can achieve a superior performance. In this initial study shown in Fig. 3, we set the dimension of hyperbolic embedding as 1024, and we attempt to use a lower dimension (*i.e.*, 128) to evaluate the effectiveness of curvature. The study shown in Fig. 4 answers the question that a larger value of curvature works better in a lower dimension hyperbolic space, while the overall performance in a lower dimension will be decreased significantly. A conclusion hence can be made that both the curvature value and embedding dimension should be considered in DNNs for a good practice of hyperbolic embeddings.

5.6.4. *Analysis on the robustness of the proposed kernels*

We further perform the study to analyze the robustness of the proposed kernels. Specifically, we conduct the experiments in the presence of outliers in the form of mislabeled images. In doing so, we manually

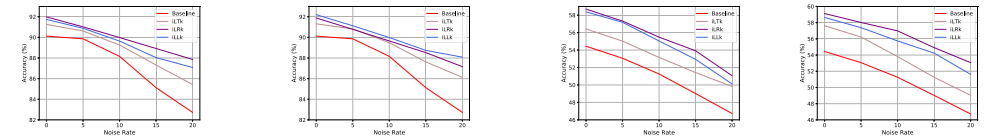


(a) Lorentz model (CIFAR-10) (b) Poincaré ball (CIFAR-10) (c) Lorentz model (CIFAR-100) (d) Poincaré ball (CIFAR-100)



(e) Lorentz model (*tiered*-ImageNet) (f) Poincaré ball (*tiered*-ImageNet) (g) Lorentz model (FC100) (h) Poincaré ball (FC100)

Fig. 4. Analysis on the curvature of low dimensional hyperbolic space for both Lorentz model and Poincaré ball.



(a) Lorentz model (CIFAR-10) (b) Poincaré ball (CIFAR-10) (c) Lorentz model (*tiered*-ImageNet) (d) Poincaré ball (*tiered*-ImageNet)

Fig. 5. Robust analysis of the proposed kernels for both Lorentz model and Poincaré ball.

set the noise rate of mislabeled images in the training, and the results are illustrated in Fig. 5. One can observe that integrating the proposed kernels can bring robustness to the learning process and gain show the superiority of the proposal.

5.7. Discussion

In this part, we would like to discuss whether the proposed kernel can be used in other learning fields. In the classification task, we integrate the proposed kernels in the cross-entropy loss. Specifically, the proposed kernels measure the similarity between two samples in a high-dimensional (or even infinite-dimensional) Kreĭn spaces, which realizes the same role as the inner product in the Euclidean space. We further plug the kernels in the loss function to replace the inner product. Since the loss function of few-shot learning, zero-shot learning *etc*. shares a similar formulation of the cross-entropy loss, *e.g*., prototype loss (Snell et al., 2017), cross-entropy loss (Li, Zhu, & Gong, 2019), knowledge distillation loss (Hinton et al., 2014), *etc*. and the kernels can be plugged to those loss functions seamlessly. Along with the applications used in our paper, one can further adopt the proposed kernels in other learning scenarios, *e.g*., graph learning, natural language processing, *etc*.

6. Conclusion

This paper studies the indefinite kernels in the hyperbolic space, such that the hyperbolic embedding can naturally benefit from the representation power by the indefinite kernels. Specifically, we propose adaptive embedding functions in the Lorentz model and Poincaré ball, respectively, and construct a series of indefinite Lorentz kernels and indefinite Poincaré kernels. The potential of the proposed kernels is evaluated across image classification, few-shot learning, zero-shot learning, person re-identification, knowledge distillation, *etc*. We believe our work has shown the effectiveness of the indefinite kernels, which will positively bring the direction in the hyperbolic deep learning field. Future works include developing more advanced indefinite kernels and exploiting the proposed kernels in the NLP and GNN fields.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China under Grant 62306070 and in part by the Southeast University Start-Up Grant for New Faculty under Grant 4009002309. Furthermore, the work was also supported by the Big Data Computing Center of Southeast University.

Data availability

Data will be made available on request.

References

Absil, P.-A., Mahony, R., & Sepulchre, R. (2007). Optimization algorithms on matrix manifolds. Princeton University Press.

Akata, Z., Perronnin, F., Harchaoui, Z., & Schmid, C. (2015). Label-embedding for image classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1425–1438.

Alpay, D. (2002). The Schur algorithm, reproducing kernel spaces and system theory. *The Mathematical Gazette*, *86*, 573, URL https://api.semanticscholar.org/CorpusID:117254570.

Bdeir, A., Schwethelm, K., & Landwehr, N. (2023). Hyperbolic geometry in computer vision: A novel framework for convolutional neural networks. arXiv:arXiv:2303.15919. arXiv:2303.15919 [cs.CV].

Bognár, J. (1974). Indefinite inner product spaces. Springer.

Cannon, J. W., Floyd, W. J., Kenyon, R., & Parry, W. R. (1997). Hyperbolic geometry. MSRI Publications.

Cho, H., DeMeo, B., Peng, J., & Berger, B. (2019). Large-margin classification in hyperbolic space. In *The twenty-second international conference on artificial intelligence and statistics* (pp. 1832–1840).

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 1–30.

Deng, J., Dong, W., Li, R. S. L.-J., Li, K., & Li, F.-F. (2009). ImageNet: A large-scale hierarchical image database. In *IEEE conference on computer vision and pattern recognition* (pp. 248–255).

Fang, P., Harandi, M., Lan, Z., & Petersson, L. (2023a). Poincaré kernels for hyperbolic representations. *International Journal of Computer Vision*.

Fang, P., Harandi, M., Le, T., & Phung, D. (2023b). Hyperbolic geometry in computer vision: A survey. arXiv:arXiv:2304.10764. arXiv:2304.10764 [cs.CV].

Fang, P., Harandi, M., & Petersson, L. (2021a). Kernel methods in hyperbolic spaces. In *IEEE/CVF international conference on computer vision* (pp. 10665–10674).

Fang, P., Zhou, J., Roy, S. K., Ji, P., Petersson, L., & Harandi, M. (2021). Attention in attention networks for person retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 4626–4641.

Fang, P., Zhou, J., Roy, S. K., Petersson, L., & Harandi, M. (2019). Bilinear attention networks for person retrieval. In *IEEE/CVF international conference on computer vision* (pp. 8030–8039).

Felzenszwalb, P. F., Girshick, R. B., McAllester, D., & Ramanan, D. (2010). Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1627–1645.

Feragen, A., Lauze, F., & ren Hauberg, S. (2015). Geodesic exponential kernels: when curvature and linearity conflict. In *IEEE conference on computer vision and pattern recognition* (pp. 3032–3042).

Ganea, O.-E., Bécigneul, G., & Hofmann, T. (2018). Hyperbolic neural networks. In *Advances in neural information processing systems* (pp. 5345–5355).

Gulcehre, C., Denil, M., Malinowski, M., Razavi, A., Pascanu, R., Hermann, K. M., et al. (2019). Hyperbolic attention networks. In *International conference on learning representations* (pp. 1–11).

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *IEEE conference on computer vision and pattern recognition* (pp. 770—-778).

Hinton, G., Vinyals, O., & Dean, J. (2014). Distilling the knowledge in a neural network. In *Advances in neural information processing systems deep learning workshop*.

Hofmann, T., Scholkopf, B., & Smola, A. J. (2008). Kernel methods in machine learning. *The Annals of Statistics*, 1171–1220.

Hong, J., Fang, P., Li, W., Han, J., Petersson, L., & Harandi, M. (2023). Curved geometric networks for visual anomaly recognition. *IEEE Transactions on Neural Networks and Learning Systems*, 1–14.

Huang, X., Suykens, J. A. K., Wang, S., Hornegger, J., & Maier, A. (2018). Classification with truncated L1 distance kernel. *IEEE Transactions on Neural Networks and Learning Systems*, *29*(5), 2025–2030.

Jayasumana, S., Hartley, R., Salzmann, M., Li, H., & Harandi, M. (2013). Kernel methods on the Riemannian manifold of symmetric positive definite matrices. In *IEEE conference on computer vision and pattern recognition* (pp. 73–80).

Jayasumana, S., Ramalingam, S., & Kumar, S. (2021). Kernelized classification in deep networks. *arXiv:2012.09607v2*.

Khrulkov, V., Mirvakhabova, L., Ustinova, E., Oseledets, I., & Lempitsky, V. (2020). Hyperbolic image embeddings. In *IEEE/CVF conference on computer vision and pattern recognition* (pp. 6418–6428).

Krizhevsky, A. (2009). Learning multiple layers of features from tiny images. In *Technical report*.

Lampert, C. H., Nickisch, H., & Harmeling, S. (2013). Attribute-based classification for zero-shot visual object categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 453–465.

Li, Z., & Holem, D. (2018). Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *40*(12), 2935–2947.

Li, W., Zhu, X., & Gong, S. (2019). Scalable person re-identification by harmonious attention. *International Journal of Computer Vision*, 1635–1653.

Linial, N., London, E., & Rabinovich, Y. (1995). The geometry of graphs and some of its algorithmic applications. *Combinatorica*, *15*, 215–245.

Liu, S., Chen, J., Pan, L., Ngo, C.-W., Chua, T.-S., & Jiang, Y.-G. (2020). Hyperbolic visual embedding learning for zero-shot recognition. In *IEEE/CVF conference on computer vision and pattern recognition* (pp. 9273–9281).

Liu, F., Huang, X., Chen, Y., & Suykens, J. (2021). Fast learning in reproducing kernel Krein spaces via signed measures. In *The 24th international conference on artificial intelligence and statistics* (pp. 388–396).

Liu, F., Huang, X., Shi, L., Yang, J., & Suykens, J. A. K. (2019a). A double-variational Bayesian framework in random Fourier features for indefinite kernels. *IEEE Transactions on Neural Networks and Learning Systems*, *31*(8), 2965–2979.

Liu, Q., Nickel, M., & Kiela, D. (2019b). Hyperbolic graph neural networks. In *Advances in neural information processing systems* (pp. 8230–8241).

Mathieu, E., Le Lan, C., Maddison, C. J., Tomioka, R., & Teh, Y. W. (2019). Continuous hierarchical representations with poincaré variational auto-encoders. *32*, In *Advances in neural information processing systems (neurIPS)* (pp. 12565–12576).

Nagano, Y., Yamaguchi, S., Fujita, Y., & Koyama, M. (2019). A wrapped normal distribution on hyperbolic space for gradient-based learning. In *International conference on machine learning*.

Ong, C. S., Mary, X., Canu, S., & Smola, A. J. (2004). Learning with non-positive kernels. In *The 21st international conference on machine learning* (pp. 1–8).

Oreshkin, B., Rodríguez López, P., & Lacoste, A. (2018). TADAM: Task dependent adaptive metric for improved few-shot learning. In *Advances in neural information processing systems* (pp. 721–731).

Patterson, G., & Hays, J. (2012). SUN attribute database: Discovering, annotating, and recognizing scene attributes. In *IEEE conference on computer vision and pattern recognition* (pp. 2751–2758).

Peng, W., Varanka, T., Mostafa, A., Shi, H., & Zhao, G. (2021). Hyperbolic deep neural networks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *44*, 10023–10044.

Ren, M., Triantafillou, E., Ravi, S., Snell, J., Swersky, K., Tenenbaum, J. B., et al. (2018). Meta-learning for semi-supervised few-shot classification. In *International conference on learning representations* (pp. 1–11).

Ristani, E., Solera, F., Zou, R., Cucchiara, R., & Tomasi, C. (2016). Performance measures and a data set for multi-target, multi-camera tracking. In *European conference on computer vision workshop on benchmarking multi-target tracking* (pp. 17–35).

Schleif, F.-M., & Tino, P. (2015). Indefinite proximity learning: A review. *Neural Computation*, *27*(10), 2039–2096.

Smola, A., Óvári, Z., & Williamson, R. C. (2000). Regularization with dot-product kernels. In *Advances in neural information processing systems (neurIPS)* (pp. 308–314).

Snell, J., Swersky, K., & Zemel, R. (2017). Prototypical networks for few-shot learning. In *Advances in neural information processing systems* (pp. 4077–4087).

Sung, F., Yang, Y., Zhang, L., Xiang, T., Torr, P. H., & Hospedales, T. M. (2018). Learning to compare: Relation network for few-shot learning. In *IEEE conference on computer vision and pattern recognition* (pp. 1199–1208).

Vinyals, O., Blundell, C., Lillicrap, T., Kavukcuoglu, K., & Wierstra, D. (2016). Matching networks for one shot learning. In *Advances in neural information processing systems* (pp. 3630–3638).

Wah, C., Branson, S., Welinder, P., Perona, P., & Belongie, S. (2011). The caltech-UCSD birds-200–2011 dataset. In *Tech. Rep. CNS-TR-2011-001*. California Institute of Technology.

Xian, Y., Akata, Z., Sharma, G., Nguyen, Q., Hein, M., & Schiele, B. (2016). Latent embeddings for zero-shot classification. In *IEEE conference on computer vision and pattern recognition* (pp. 69–77).

Yang, M., Fang, P., & Xue, H. (2023). Expanding the hyperbolic kernels: A curvature-aware isometric embedding view. In *International joint conference on artificial intelligence*.

Ye, M., Shen, J., Lin, G., Xiang, T., Shao, L., & Hoi, S. C. H. (2021). Deep learning for person re-identification: A survey and outlook. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2872–2893.

Zheng, L., Shen, L., Tian, L., Wang, S., Wang, J., & Tian, Q. (2015). Scalable person re-identification: A benchmark. In *IEEE international conference on computer vision* (pp. 1116–1124).