De l'espace euclidien à l'espace hyperbolique : repenser la classification hiérarchique des images de scènes de télédétection

Manal Hamzaoui

COLLEGE MATHSTIC
DOCTORAL BRETAGNE
BRETAGNE OCEANE

Université Bretagne Sud
ubs

THÈSE DE DOCTORAT DE

L'UNIVERSITÉ DE BRETAGNE SUD

ÉCOLE DOCTORALE N° 644
Mathématiques et Sciences et Technologies de l'Information et de la Communication en Bretagne Océane
Spécialité : *Informatique*

Par
Manal HAMZAOUI

From Euclidean to Hyperbolic Space: Rethinking Hierarchical Classification of Remote Sensing Scene Images

Thèse présentée et soutenue à Vannes, le 30 Mai 2023
Unité de recherche : UMR 6074 IRISA
Thèse N° : 663

Rapporteurs avant soutenance :

Erchan APTOULA — Associate Professor, Sabanci University
Céline HUDELOT — Professor, CentraleSupélec

Composition du Jury :

Président :	Valérie GOUET-BRUNET	Research director (DR1), Institut Géographique National (IGN)
Examinateurs :	Valérie GOUET-BRUNET	Research director (DR1), Institut Géographique National (IGN)
	Benjamin PERRET	Professor, ESIEE - LIGM, Université Gustave Eiffel
	Thomas CORPETTI	Research director, HDR, CNRS, LETG-Rennes
Dir. de thèse :	Sébastien LEFÈVRE	Professor, Université Bretagne Sud
Encadr. de thèse :	Laetitia CHAPEL	Associate Professor, HDR, Université Bretagne Sud
Encadr. de thèse :	Minh-Tan PHAM	Assistant professor, Université Bretagne Sud

Acknowledgement

And just like that, the journey has come to an end! I've been eagerly awaiting this moment to write down these words, though I've also dreading it, fearing that my words wouldn't suffice to convey how thankful I am for the support I've received over these three years of thesis work (okay, and a few extra months).

First of all, I want to express my gratitude to my supervisors, Laetitia, Minh-Tan, and Sébastien, as my PhD journey owes much to your unwavering guidance and assistance. Despite your numerous commitments, you've consistently dedicated time to provide me with your expertise, and I'm truly appreciative of your availability and meticulousness. This thesis would lack its essential dimension without the insightful comments you have provided. I feel fortunate to have had the privilege of your supervision.

I extend my gratitude to my thesis reviewers, Erchan Aptoula and Céline Hudelot, for readily agreeing to evaluate my thesis without any hesitation. Their feedback has been of remarkable quality, encompassing insightful criticism and novel research avenues. I also wish to express my appreciation to the esteemed members of the jury, Benjamin Perret and Thomas Corpetti, whose notable research contributions I deeply respect. Furthermore, I am indebted to Valérie Gouet-Brunet, who chaired my thesis jury with a commendable blend of professionalism and kindness. Presenting and defending my thesis before such a distinguished panel is a source of deep pride.

I wish to convey my sincere appreciation to the members of the Obelix team at IRISA. I would also like to extend my gratitude to Anne and Mario, whose substantial contributions, both in administrative and technical capacities, have been of immense assistance to me.

Beyond the scientific aspect of my thesis, the social aspect holds great importance. I was lucky to enjoy wonderful moments with colleagues and friends from the IRISA laboratory: Jamal, Monica, Iris, Marion, Nan, Lucie, Claire, Anne, Behzad, Elyes, Ân, Mansour, Corentin, Guillaume, Paul, Renan, Jean-Christophe, and Brendan. While I might have unintentionally omitted a few, I want to express my gratitude to all those who contributed to making this thesis such an enjoyable journey.

I want to express my appreciation to my friends who provided unwavering encourage-

ment, particularly Raounak and Djedjiga. I am also deeply thankful to Kaouter for her unwavering presence and exceptional assistance.

I am immensely grateful to my entire family, who played an invaluable role throughout this thesis journey. My parents, Saïd and Zahia, deserve a special mention for their unwavering encouragement to always give my best. Their steadfast support and dedication to their children's well-being have been an endless source of inspiration. To my elder sister Khadidja, I extend heartfelt thanks for her role as a caring protector, for the unwavering encouragement, and the unwavering support she provided at every step of this journey. Lastly, to my brothers Mohammed Nadjib and Abderahim, I thank you from the bottom of my heart for your constant support and deep affection. Your presence and support have been the strong pillars that made this thesis experience a success.

Résumé étendu

Introduction

Contexte : L'observation de la Terre (OT) consiste à étudier et analyser la Terre et son environnement à l'aide de données de télédétection (Tang et al., 2021). Cette pratique permet de mesurer et observer en détail les structures de la surface terrestre, afin de comprendre et surveiller les systèmes et phénomènes terrestres. Elle fournit également des informations pour diverses applications, telles que la gestion des ressources naturelles (Shahbazi et al., 2014), la surveillance du climat et des conditions météorologiques (Calbo & Sabburg, 2008), l'intervention en cas de catastrophe (Schumann et al., 2018) et l'aménagement du territoire (Van Westen et al., 2008).

Les avancées technologiques dans le domaine de l'OT ont rendu disponibles une grande variété de types de données, tels que les images multi/hyper-spectrales (Gerhards et al., 2019) et les images radar à synthèse d'ouverture (SAR) (Li et al., 2021b), toutes en haute résolution. Les images à haute résolution (HRRS) peuvent être collectées quotidiennement à partir de diverses sources, notamment des drones, des capteurs aéroportés et principalement des satellites (Dutta & Das, 2023), ce qui engendre des téraoctets de données rendant difficile l'annotation précise de toutes les images. La gestion de ces volumes d'images HRRS massifs est devenue une tâche critique et nécessaire pour l'observation intelligente de la Terre. Par conséquent, il est d'une importance extrême d'analyser et de comprendre le contenu sémantique des images HRRS volumineuses et complexes, qui présentent une texture nette et des informations spatiales riches.

Plusieurs types d'analyse d'images HRRS existent, notamment la détection d'objets (Shin et al., 2020), la détection des changements (Li et al., 2021b) et la classification d'images (Cheng et al., 2017; Cheng et al., 2020; Dutta & Das, 2023), qui est considérée comme l'une des tâches les plus importantes de l'analyse d'images HRRS. La classification d'images peut être effectuée à différents niveaux de granularité, à savoir au niveau du pixel (également appelée segmentation sémantique), au niveau de l'objet et au niveau de la scène. La classification au niveau de la scène fournit des informations sémantiques, ce qui permet

l'extraction de caractéristiques de niveau supérieur (Cheng et al., 2017; Cheng et al., 2020). Ici, la "scène" fait référence à une zone d'image découpée dans une image de télédétection à grande échelle qui contient des informations sémantiques pertinentes sur la surface terrestre (par exemple, *zone résidentielle*, *zone industrielle* et *zone commerciale*) (Cheng et al., 2020). Par conséquent, la classification des scènes suscite un intérêt croissant en télédétection et constitue un domaine de recherche actif.

Problématique et motivations : La classification des images de scènes de télédétection (RSISC) consiste à attribuer automatiquement une étiquette sémantique spécifique, telle que *airport*, *beach*, *farmland* ou *residential area*, à chaque image de télédétection en fonction de son contenu (Cheng et al., 2020). Plusieurs approches ont été proposées pour résoudre ce problème, impliquant généralement deux étapes : l'extraction des caractéristiques visuelles des images de scènes, suivie d'un algorithme d'apprentissage automatique pour effectuer la classification. La classification des images de scènes étant généralement effectuée dans un espace de caractéristiques, la construction d'une méthode précise de classification dépend fortement de la qualité de la représentation des caractéristiques (Cheng et al., 2017). Les méthodes récentes de classification des scènes d'images de télédétection se basent sur l'apprentissage profond et ont montré une capacité impressionnante de représentation des caractéristiques, ce qui a permis d'améliorer considérablement les performances. Cependant, elles souffrent de certaines limitations dues à la nature des données de télédétection (Cheng et al., 2020; Wang et al., 2022b).

Les images de scènes comprennent généralement non seulement les éléments principaux de la scène, mais aussi des objets identifiables ou du contexte. Ainsi, les images de scènes sont composées de divers objets, qui fournissent un modèle spatial en tant que zone fonctionnelle qui est sémantiquement cohérente et qui reflète souvent le monde réel d'une manière visuelle identifiable par les humains (Wang et al., 2019). De plus, une variation significative peut être observée à l'intérieur des classes, tandis que des images de scène appartenant à des catégories différentes peuvent parfois être très similaires en termes de contenu visuel et donc difficiles à distinguer, même pour les humains. En effet, les mêmes objets peuvent apparaître dans des images de scènes appartenant à des catégories différentes.

La classification des images de scènes de télédétection est souvent considérée comme une tâche difficile en raison de ces divers facteurs. Pour surmonter ces obstacles, la plupart des approches proposées prennent en compte les relations inter- et intra-classes lors

de la construction de l'extracteur de caractéristiques et de l'algorithme de classification. Cependant, ces approches sont souvent conçues comme des classifieurs plats, qui considèrent toutes les classes de scènes comme également distinctes, ce qui ignore les informations sémantiques hiérarchiques potentielles entre elles (la relation étiquette-étiquette). En conséquence, ces approches peuvent conduire à une confusion entre des classes non liées sémantiquement. Néanmoins, en considérant ces difficultés d'un autre point de vue, il serait possible d'obtenir un contexte sémantique plus global que la classe à grain fin. Ainsi, l'utilisation d'informations sémantiques hiérarchiques reflétant l'interaction entre les classes pourrait améliorer les performances des classifieurs de scènes et rendre les prédictions du modèle plus cohérentes d'un point de vue sémantique.. Cette information hiérarchique est généralement disponible explicitement via la hiérarchie de classes, ou implicitement dans les données.

- ***Information hiérarchique explicite :*** La hiérarchie de classes est une organisation de classes à plusieurs niveaux dans laquelle les classes de scène sont les étiquettes à grain fin au bas de la hiérarchie de classes, qui sont ensuite regroupées en fonction des informations sémantiques partagées à des niveaux plus grossiers. Les étiquettes à gros grain englobent donc une ou plusieurs étiquettes à grain fin.

  La hiérarchie des classes est généralement disponible, telles que *Corine Land Cover* (CLC) (Bossard et al., 2000) et L'*European Nature Information System* (EUNIS) (Davies et al., 2004), ou facile à construire (manuellement ou automatiquement).

- ***Information hiérarchique implicite :*** Bien que la hiérarchie des classes ne soit pas prise en compte pour décrire les relations entre les classes de scènes, les informations hiérarchiques sont néanmoins implicitement présentes entre les images de scènes. Cette observation est soutenue par un concept appelé *Gromov* $\delta-$*hyperbolicity* (Gromov, 1987), appelé $\delta-$*hyperbolicity* pour plus de simplicité, qui nous permet de mesurer le degré d'information hiérarchique d'un ensemble de données. En pratique, pour quantifier cette information hiérarchique, nous calculons généralement la métrique invariante à l'échelle δ_{rel} (la $\delta-$*hyperbolicité* relative) qui prend des valeurs dans $[0, 1]$, plus elle est proche de zéro, plus l'information hiérarchique est forte (Khrulkov et al., 2020). En outre, une faible valeur de δ_{rel} indique que l'espace d'intégration de données a une géométrie hyperbolique sous-jacente et que cet espace hyperbolique conviendrait en tant qu'espace d'intégration (Tifrea et al., 2019).

Afin de valider l'hypothèse concernant l'hyperbolicité des ensembles de données visuelles, nous calculons la métrique invariante à l'échelle δ_{rel} de cinq ensembles de données de scènes de télédétection. Nous observons que les valeurs δ_{rel} dérivées des ensembles de données d'images de scènes sont nettement plus proches de 0 que de 1, ce qui se traduit par un degré d'hyperbolicité assez élevé, suggérant ainsi que les représentations hyperboliques des images de scènes peuvent être bénéfiques pour la tâche de classification.

Objectifs Cette thèse vise à examiner comment les informations hiérarchiques, qu'elles soient explicites ou implicites dans les données de télédétection, notamment entre les différentes classes de scènes, peuvent impacter les méthodes d'extraction de caractéristiques et de classification. L'objectif est donc de répondre aux questions de recherche suivantes :

- La hiérarchie des classes peut-elle être utile pour la classification de scènes de télédétection ?

- L'espace hyperbolique est-il plus approprié pour représenter les données de télédétection que l'espace euclidien, en particulier les images de scènes ?

Au cours de mon doctorat, j'ai contribué aux deux aspects mentionnés dessous : 1- l'introduction explicite d'informations hiérarchiques lors de l'apprentissage d'un réseau profond, 2- l'intégration des images de scènes dans l'espace hyperbolique qui accentue la hiérarchie sous-jacente entre les classes de scènes.

Contributions de la thèse

Exploitation de la hiérarchie de classes via une fonction de perte spécifique pour l'analyse de scènes de télédétection

Cette partie de la thèse se focalise sur le premier type d'information hiérarchique, à savoir l'information explicite fournie par une hiérarchie de classes qui reflète les relations sémantiques entre les classes de scènes. Des études récentes se sont intéressées à cette direction de recherche et ont proposé diverses approches principalement associées aux trois lignes de recherche :

- *Architectures hiérarchiques :* les approches de cette catégorie modifient l'architecture du modèle original en fonction de la hiérarchie de classes afin d'apprendre à reconnaître les classes à différents niveaux.

- *Encodage des étiquettes :* les méthodes d'encodage des étiquettes (*label-embedding*) convertissent l'espace discret des étiquettes en un espace continu en se basant sur les relations entre les étiquettes données par la hiérarchie de classes.

- *Fonction de perte hiérarchique :* cette catégorie modifie la fonction de perte en donnant plus de poids à des catégories spécifiques dans la hiérarchie de classes.

Dans cette partie de la thèse, nous proposons d'introduire la hiérarchie de classe via une fonction de perte spécifique pour la classification d'images de scènes de télédétection.

Dans la première section de ce chapitre, nous procédons à une vérification initiale qui vise à montrer les bénéfices potentiels de l'incorporation explicite de l'information hiérarchique dans la construction des caractéristiques. Nous avons donc ajusté la recherche présentée dans (Yu et al., 2020), qui proposait un auto-encodeur variationnel guidé (VAE) pour intégrer des médicaments dont les étiquettes étaient disposées suivant une hiérarchie de classes, à notre contexte. Plus précisément, nous utilisons le VAE pour intégrer des images de scènes, tout en restreignant son espace latent à l'aide d'une perte de classement local (*soft local ranking loss*), qui est paramétrée par la hiérarchie de classe. Nous évaluons la qualité de l'espace latent du modèle, et donc la qualité des intégrations qui en résultent, ainsi que la capacité à discriminer entre les classes à l'aide d'un classifieur simple de type $1-$NN. Les résultats de nos expériences montrent que les informations hiérarchiques sur les classes peuvent être une source d'information utile pour améliorer les performances du classifieur.

Dans la deuxième section, nous proposons un réseau prototypique hiérarchique pour la classification d'images de scènes avec peu d'exemples (*few-shot*). Plus précisément, nous augmentons le réseau prototypique traditionnel (Snell et al., 2017) en établissant des prototypes à chaque niveau de la hiérarchie de classes, plutôt qu'uniquement au niveau des nœuds feuilles. Les informations relatives à la hiérarchie de classes sont alors introduites par le biais de ces prototypes hiérarchiques qui seront impliqués dans une somme pondérée de la perte d'entropie croisée sur les différents niveaux de la hiérarchie de classes. Les résultats expérimentaux montrent les avantages de l'utilisation de la hiérarchie des classes pour résoudre le problème du RSISC à quelques coups. Notre réseau prototypique

hiérarchique a permis donc de régulariser l'espace latent, ce qui permet d'obtenir de meilleures performances par rapport à son équivalent traditionnel.

Géométrie hyperbolique : application à l'analyse d'images de scènes de télédétection

Dans cette deuxième partie de la thèse, nous nous concentrons sur les approches qui traitent l'information hiérarchique implicite dans les données de télédétection en opérant dans un espace hyperbolique. Ces espaces ont démontré leur pertinence pour représenter des données hiérarchiques ou des données avec une hiérarchie sous-jacente. L'objectif de cette partie est donc d'étudier le potentiel des représentations hyperboliques dans le contexte des données de télédétection, avec un accent particulier sur les images de scènes.

De même que dans la section précédente, nous examinons deux contextes dans cette étude : un contexte non supervisé et un contexte *few-shot*. Dans le cas non supervisé, nous adoptons le framework VAE pour intégrer des images de scènes. Plus précisément, nous utilisons l'extension de la distribution normale à l'espace hyperbolique pour construire notre VAE hyperbolique. Cela implique de conserver l'ensemble du réseau VAE dans l'espace euclidien en généralisant uniquement son espace latent, qui est conforme à une distribution normale (Nagano et al., 2019). Dans le contexte *few-shot*, nous utilisons le réseau prototypique pour l'intégration d'images de scènes. La généralisation de ce dernier est réalisée par l'ajout d'une couche qui permet la passage des caractéristiques euclidiennes générées par l'extracteur de caractéristiques à l'espace hyperbolique (Khrulkov et al., 2020). Comme il s'agit dans les deux cas d'architectures hybrides euclidiennes-hyperboliques qui présentent un risque de disparition du gradient, nous utilisons la technique de seuillage des caractéristiques euclidiennes pour contourner ce problème et donc assurer la stabilité numérique des deux modèles.

Dans des contextes non supervisés et avec peu d'images, nous montrons la supériorité des intégrations hyperboliques d'images de scènes de télédétection par rapport à leurs équivalents euclidiens. Néanmoins, le choix des hyper-paramètres tels que la courbure hyperbolique est très important. Assurer la stabilité numérique dans l'espace hyperbolique reste un défi majeur. Cependant, le seuillage des caractéristiques est une solution simple mais efficace pour contourner ce problème.

En résumé, cette étude constitue une ouverture vers l'application de l'espace hyperbolique aux images de télédétection. Bien que difficiles, les propriétés des images de

télédétection s'alignent bien avec les propriétés géométriques de l'espace hyperbolique, ce qui ouvre une perspective prometteuse pour les futures recherches dans ce domaine.

Publications

Le travail présenté dans ce manuscrit est basé sur les publications suivantes :

Article de journal

(1) Hamzaoui, M., Chapel, L., Pham, M. T., & Lefèvre, S., *Hyperbolic Prototypical Network for Few Shot Remote Sensing Scene Classification*, Pattern Recognition Letters, En cours d'évaluation.

Articles de conférence

(1) Hamzaoui, M., Chapel, L., Pham, M. T., & Lefèvre, S., (2023), *Hyperbolic variational auto-encoder for remote sensing scene embeddings*, IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Accepté pour une présentation orale.

(2) Hamzaoui, M., Chapel, L., Pham, M. T., & Lefèvre, S., (2022), *A hierarchical prototypical network for few-shot remote sensing scene classification*, The 3rd International Conference on Pattern Recognition and Artificial Intelligence (ICPRAI), Paris, France, June 1-3, 2022. [Prix du meilleur article]

(3) Hamzaoui, M., Chapel, L., Pham, M. T., & Lefèvre, S. (2021). *Hyperbolic Variational Auto-Encoder for Remote Sensing Scene Classification*. In Journées Francophones des Jeunes Chercheurs en Vision par Ordinateur (ORASIS).

Contents

List of Figures

List of Tables

Chapter 1

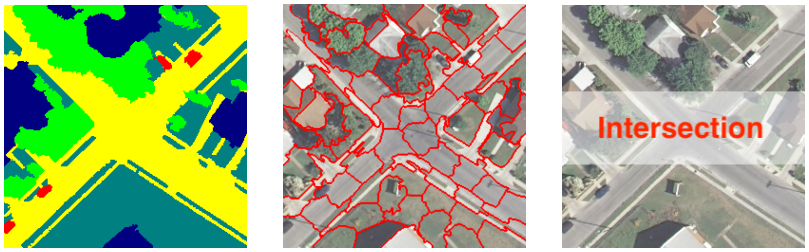
Introduction

1.1 Context: Remote sensing scene classification

Earth observation (EO) is the study and analysis of the Earth and its environment using remote sensing (RS) data (Tang et al., 2021). EO therefore measures and observes the detailed structures of the Earth's surface, in order to understand and monitor Earth systems and processes, as well as to provide information for a variety of applications such as natural resource management (Shahbazi et al., 2014), climate and weather monitoring (Calbo & Sabburg, 2008), disaster response (Schumann et al., 2018) and land-use planning (Van Westen et al., 2008).

With EO technologies continually developing, a variety of data types (e.g. multi/hyper-spectral (Gerhards et al., 2019) and synthetic aperture radar (SAR) (Li et al., 2021b)) of high-resolution RS images of the Earth's surface are widely available. High-resolution RS (HRRS) images can be collected every day via a range of sources such as drones, airborne sensors and mainly satellites (Dutta & Das, 2023), resulting in terabytes of data, making it almost impossible to accurately analyse every produced image. Effectively managing these enormous volumes of HRRS images is therefore becoming an urgent and required task for EO. Accordingly, analysing and understanding the semantic content of huge and complex HRRS images, which have clear texture and rich spatial information, is extremely important.

Several HRRS image analysis methods are available in the literature, including but not limited to object detection (Shin et al., 2020), change detection (Li et al., 2021b), and image classification (Cheng et al., 2017; Cheng et al., 2020; Dutta & Das, 2023). The latter is considered to be one of the foremost tasks. Image classification can be performed at various levels of granularity, namely pixel-level (also known as semantic segmentation), object-level, and scene-level classification (Fig. 1.1). Scene-level classification provides semantic



(a) (b) (c)

Figure 1.1: Three levels of remote sensing image classification: (a) Pixel-level classification aims to assign a class label to each pixel. (b) Object-level classification aims at recognising objects in remote sensing images. (c) Scene-level classification aims to categorise each remote sensing image patch into a semantic class.

information, enabling the extraction of higher-level features (Cheng et al., 2017; Cheng et al., 2020). The "scene" here refers to an image patch cropped out of a large-scale remote sensing image which includes relevant semantic information about the earth's surface (e.g., *dense residential area*, *medium residential area*, and *sparse residential area*) (Cheng et al., 2020). Consequently, scene classification has received increasing attention in RS and constitutes an active research topic.

1.1.1 Remote sensing scene classification

Remote sensing image scene classification (RSISC) aims to automatically assign a specific semantic label (e.g. *airport*, *beach*, *farmland* or *residential area*) to each remote sensing image based on its content (Cheng et al., 2020). As scene classification is usually performed in a feature space, the construction of an accurate scene classification method depends strongly on the efficiency of the feature representation (Cheng et al., 2017). Several methods have been proposed to carry out this task, which can be divided into three main categories from the perspective of the features used: methods based on manual feature extraction, methods based on unsupervised feature extractors, and methods based on deep learning.

Handcrafted-feature-based methods Most early RSISC approaches are based on low-level or handcrafted features, e.g., colour histograms (Swain & Ballard, 1991), texture descriptors (Ojala et al., 2002), scale-invariant feature transform (SIFT) (Lowe, 2004) and histogram of oriented gradients (HOG) (Dalal & Triggs, 2005). These approaches mainly rely on engineering skills and domain expertise to retrieve basic features from the scene image such as colour, texture, shape, spatial and spectral information, or their combination

resulting in human-engineering descriptors. These descriptors are then used for scene classification. However, with the increasing volume and complexity of RS data, the ability of human-engineered descriptors is becoming limited or even impoverished (Cheng et al., 2017).

Unsupervised-feature-learning-based methods Learning features automatically from unlabelled scene images has been considered a more practical approach and has become an attractive alternative to human-engineering features. These emerging approaches are referred to as medium-level or unsupervised feature learning. They aim to learn a set of basis functions which encode the handcrafted features or raw pixel intensity values into a set of learned features (Cheng et al., 2017; Cheng et al., 2020). As such, a wide variety of scene classification approaches based on unsupervised learning have been proposed, such as principal component analysis (PCA) (Chaib et al., 2016), k-means clustering (Zhao et al., 2014), sparse coding (Cheriyadat, 2014) and auto-encoders (Zhang et al., 2015). Automatically learned features derived from the scene images are more discriminating and yield better classification performance than the manually designed features, yet they can no longer satisfy the needs of a high-resolution scene classification (Cheng et al., 2017).

Deep-feature-learning-based methods Deep learning-based methods have shown impressive feature representation capability which has significantly improved the performance of remote sensing image scene classification. Most of these scene classification deep-algorithms can be organised into four main categories: methods based on auto-encoders (AE), methods based on convolutional neural networks (CNNs), methods based on generative adversarial networks (GANs) and methods based on vision transformers (ViTs) (Wu et al., 2020).

Auto-encoder based approaches are algorithms learned generally in an unsupervised manner. Although they are deep networks, they are limited in their ability to learn discriminative features, as they are generally learned in an unsupervised manner, relying solely on the visual features of the image without utilising the scene labels.

Convolutional neural networks (CNNs), such as AlexNet (Krizhevsky et al., 2012), VGGNet (Simonyan & Zisserman, 2015), and ResNet (He et al., 2016), are powerful feature extractors that are primarily learned in a supervised manner. In this approach, the semantic information provided by the category labels is utilised to obtain good feature representations, commonly known as high-level features, and thus ensure the best

discrimination between classes. Some CNNs-based approaches, such as (Marmanis et al., 2016; Yuan et al., 2019), use pre-trained models on ImageNet solely for feature extraction. They then learn a new CNN classifier from scratch, which takes the extracted features as input. However, achieving outstanding performance through training from scratch requires a large labelled dataset as a training set, which is time consuming, expensive and may require domain expertise. Unfortunately, in the RS community, there is not yet a large labelled HRRS dataset at a comparable scale to ImageNet (contains over 14 million images) which satisfies the training requirements for CNN-based methods (Han et al., 2020). Therefore, a possible option is to fine-tune pre-trained models on ImageNet for the target datasets, which have been adopted by a large number of studies as (Bazi et al., 2019; Castelluccio et al., 2015). Although fine-tuning pre-trained CNNs can achieve remarkable performance, there may be some limitations with these approaches, such as learned features that may not fully fit the characteristics of target dataset. Alternatively, some studies such as (Chen et al., 2018) have opted to train shallow CNNs from scratch, which, showed promising results. The recent Million-AID dataset (Long et al., 2021), which includes one million images, may be a good data source for these line of approaches.

Compared to CNN-based approaches, RSISC methods based on generative adversarial networks (GANs) (Goodfellow et al., 2020) are less frequently reported in the RS literature. These methods generally employ GANs as a data augmentation technique to generate supplementary labelled samples, thus increasing the size of labelled RS datasets that will be employed to train the CNN classifiers. For instance, (Han et al., 2020) proposed a supervised Wasserstein Generative Adversarial Network (SWGAN) to generate synthetic samples that are similar to real RS images. These synthetic samples are then combined with real samples to train a deep neural network for scene classification. To perform a classification task with a generative approaches, a classifier network is required. Although these methods are learned in an unsupervised manner, they still require some labelled data to guide the generation.

More recently, around the beginning of the thesis, methods based on vision transformers (ViTs) (Dosovitskiy et al., 2021; Wu et al., 2020) have emerged in the computer vision community. These methods have demonstrated outstanding performance when compared to state-of-the-art CNNs, which has received significant attention from the remote sensing community. Thus, several promising approaches have been proposed to solve the problem of RS scene classification, such as (Deng et al., 2022; Lv et al., 2022; Sha & Li, 2022; Wang et al., 2022a).

In conclusion, RSISC deep learning approaches generally require a large set of labelled data as a training set in order to achieve outstanding performance. More recent strategies, such as semi-supervised learning (Castillo-Navarro et al., 2022; Miao et al., 2022), self-supervised learning (Berg et al., 2022; Zhao et al., 2020) and few-shot learning (Li et al., 2021c; Snell et al., 2017), have been considered to handle limited annotation samples.

Although these deep learning strategies are considered to be effective and provide high level features resulting in better classification performance compared to low and medium level features, they nevertheless suffer from some limitations due to the nature of the remote sensing data (Cheng et al., 2020; Wang et al., 2022b). The current main challenges of remote sensing image scene classification are discussed in the next section.

1.1.2 Main challenges

Generally, scene images contain not only a single object referring to a scene class, but also additional objects relevant to the scene context (Wang et al., 2019). Thus, scene images are composed of various objects (for example, Figure 1.1(b) shows that in a scene of *bridge*, besides the bridge, we can also find water and land), providing a spatial model as a functional area that is semantically consistent and often reflects the real world in a visual and human-identifiable way (Wang et al., 2019). Furthermore, a significant intra-class variation can be observed as shown in Figure 1.2(a) and (b) where two images of the same *bridge* class are very different visually. Moreover, scene images belonging to different categories are sometimes very similar in terms of visual content, and therefore difficult to distinguish, even by humans. Indeed, the same objects may occur in scene images falling into different categories, such as buildings in *medium residential* (Figure 1.2(c)) and *dense residential* (Figure 1.2(d)) or a road in a *runway* and a *freeway*.

These aspects make the classification of remote sensing scene images rather a challenging task. Therefore, the majority of the proposed approaches generally consider the inter- and intra-class relationships among scene images when constructing the feature extractor and the classification algorithm. However, looking at these aspects from another perspective could potentially result in a more global semantic context than the fine-grained class. For instance, images from *circular farmland* and *rectangular farmland* classes fall into a coarser *cultivated land* class, as well as images from *bridge*, *freeway* and *intersection* fall into a coarser *transportation* class. This suggests that hierarchical information between scene images may exist and that fine-grained scene labels may belong to a hierarchical organisation with multiple levels of granularity, as illustrated in Figure 1.3.

(a) (b) (c) (d)

Figure 1.2: Inter-class similarity and intra-class variance in remote sensing scene data. (a) and (b) variation of the bridge class images. Visual similarity between (c) medium residential and (d) dense residential.

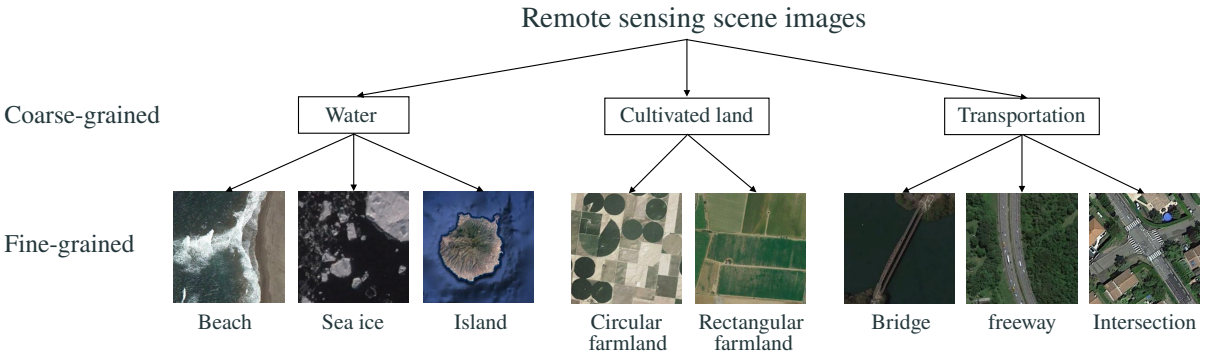


Figure 1.3: Example of a hierarchical organisation of remote sensing scene labels at multiple levels of granularity. A coarse-grained label is composed of several fine-grained labels.

Few approaches have considered this hierarchical information when solving the remote sensing scene classification, despite its relevance in better understanding the remote sensing data. In the following section, we present the interest of this hierarchical information as well as its possible forms.

1.2 Motivation: Hierarchical information for remote sensing scene analysis

Remote sensing images are complex in nature and usually exhibit a hierarchical structure. However, remote sensing image scene classification methods generally focus on the inter-class and intra-class information via the image-label relationship. They are usually designed as *flat* classifiers, which treat all non-target scene classes with the same importance, ignoring the potential hierarchical semantic information between them (the label-label relationship). As a result, confusing a *medium residential* image with a *dense residential* image has the same severity as confusing it with an *airport* or other semantically unrelated classes, which is inappropriate in terms of semantic understanding. Therefore, the use of hierarchical semantic information that reflects the interaction between classes could improve the performance of scene classifiers and, in addition, make the model predictions more semantically coherent.

The hierarchical information, which reflects the interactions between classes, can be represented via a class hierarchy (as explained below). It is expected to improve the accuracy and the efficiency of RSISC models in several ways:

- Improving classification accuracy: hierarchical information about the class organisation regularises the feature space of the classifier, allowing coarse-grained categories to be better distinguished and thus improving the classification accuracy at both coarser and finer levels.

- Reducing mistake severity: a class hierarchy can further reduce the severity of misclassifications by favouring those that share a similar semantic context over those that are semantically distant.

- Fostering knowledge transfer: a class hierarchy can facilitate the knowledge transfer from source classes to target classes sharing common semantic context, allowing better generalisation of the classifier. This is applicable not only when source and

target classes belong to the same dataset, as shown in (Garg et al., 2022; Li et al., 2019), but also when they belong to different datasets, as in (Wang et al., 2020).

Moreover, remote sensing scene images are naturally hierarchical, as we will show later. It would be of interest to consider this hierarchical nature when designing classification models.

Explicit class hierarchy

The class hierarchy is a multi-level class organisation where the scene classes are the fine-grained labels at the bottom of the class hierarchy which are then aggregated according to the shared semantic information in coarser levels. The coarse-grained labels thus comprise one or more fine-grained labels.

The class hierarchy is usually available, such as the well-known WordNet hierarchy (Miller, 1998) for natural images, or easily constructed (either manually or automatically via text embedding and clustering algorithms (Li et al., 2019)). In addition to the WordNet hierarchy, there are numerous domain-specific hierarchies, particularly in the remote sensing community, such as the Corine Land Cover (CLC) (Bossard et al., 2000) and the European Nature Information System (EUNIS) (Davies et al., 2004).

Furthermore, several datasets of high-resolution public remote sensing scene images (such as UCMerced (Yang & Newsam, 2010), WHU-RS19 (Xia et al., 2010), NWPU-RESISC45 (Cheng et al., 2017), AID (Xia et al., 2017) and PatternNet (Zhou et al., 2018)) have been introduced by different groups to learn and evaluate different scene classification methods. However, in theses datasets, the scene image is only associated with its most fine-grained class. To obtain coarser classes, it is necessary to resort to hierarchies such as CLC and EUNIS. Nevertheless, these hierarchies do not fit well to scene datasets, thus encouraging some studies to rather establish their own class organisation according to the scene classes of the considered dataset, such as PatternNet (Liu et al., 2020b) and NWPU-RESISC45 (Sen & Keles, 2022a; Zeng et al., 2022). More recently, (Long et al., 2021) provided Million-AID, a new remote sensing scene classification dataset which scene classes are organised into a hierarchy.

In this thesis, we evaluate our different approaches on the NWPU-RESISC45 database as it is widely used to assess scene classification algorithms. Furthermore, some studies such as (Sen & Keles, 2022a; Zeng et al., 2022) provide a hierarchical organisation of its classes.

NWPU-RESISC45 dataset The NWPU-RESISC45 (Cheng et al., 2017) dataset is a widely used benchmark for remote sensing image scene classification. It consists of 31 500 images of 256×256 pixels; the spatial resolution varies from approximately 30 to 0.2 m per pixel. It covers 45 scene categories, each with 700 RGB images, which can be organised hierarchically. Following (Liu et al., 2020b) and (Sen & Keles, 2022a), in which the authors propose a hierarchical organisation of the scene classes of PatternNet and NWPU-RESISC45 datasets, respectively, we construct a tree-like arrangement of these scene classes which reflects their semantic relationships. We note that the leaf level of the constructed class hierarchy corresponds to the original scene classes of the dataset. The category tree of the dataset classes is summarised in Figure 1.4.

Implicit hierarchy: δ-hyperbolicity of remote sensing scene images

Although the class hierarchy is not considered to describe the relationships between scene categories, hierarchical information is nevertheless implicitly present among scene images. This is supported by a concept called the Gromov $\delta-$hyperbolicity (Gromov, 1987), referred to as $\delta-$hyperbolicity for convenience, which enables us to measure the strength of the hierarchical information in a dataset. In practice, to quantify this information, we usually compute the scale-invariant metric δ_{rel} (the relative $\delta-$hyperbolicity) which takes values in $[0, 1]$, the closer to zero the stronger the hierarchical information (Khrulkov et al., 2020). Furthermore, a low δ_{rel} value indicates that the data embedding space has an underlying hyperbolic geometry and that hyperbolic space would be suitable as an embedding space (Tifrea et al., 2019).

In the remainder of this section, we provide some formal definitions of $\delta-$hyperbolicity and then show that RS scene images have intrinsic hierarchical relationships that could be interpreted by choosing the right settings.

Gromov $\delta-$hyperbolicity The $\delta-$hyperbolicity provides a measure of how closely the structure of a metric space $\mathcal{X}$, equipped with the distance function d, resembles that of a tree. Its calculation requires first computing the Gromov product for points $x, y \in \mathcal{X}$ with respect to $z \in \mathcal{X}$:

$$(x, y)_z = \frac{1}{2} \left(d(z, x) + d(z, y) - d(x, y) \right) \tag{1.1}$$

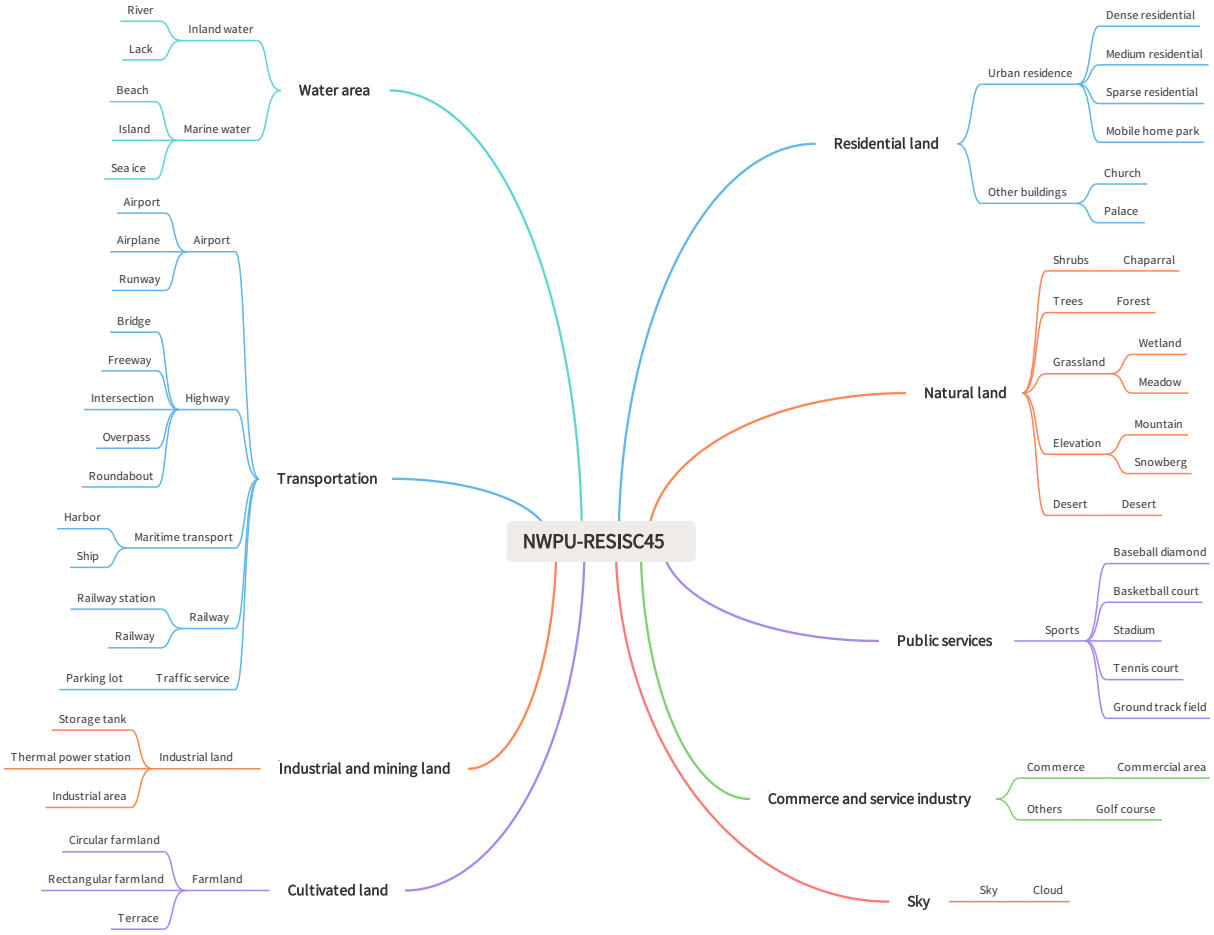


Figure 1.4: Our NWPU-RESISC45 class hierarchy. All scene classes are hierarchically organised in a four-level structure, with the root node placed at the first level. At the second level, the 9 primary scene categories – residential land, natural land, public services, commerce and service industry, sky, water area, transportation, industrial and mining land, and cultivated land – are represented by 9 nodes that aggregate 20 parent nodes at the third level. These 20 parent nodes further aggregate 45 leaf nodes at the fourth level.

The metric space $(\mathcal{X}, d)$ is then $\delta-$hyperbolic if there exists a $\delta > 0$ that fulfils the four-point condition, which is defined for $x, y, z, w \in \mathcal{X}$ as:

$$(x, z)_w \geq \min\{(x, y)_w, (y, z)_w\} - \delta \tag{1.2}$$

In practice, the δ value is defined as the largest coefficient in the matrix $(M \otimes M) - M$ where M is the matrix of pairwise Gromov products (using eq.(1.1)) and $\otimes$ denotes the

min-max matrix product defined as:

$$(A \otimes B)_{ij} = \max_k \min \{A_{ik}, B_{kj}\} \tag{1.3}$$

For example, a tree is $0-$hyperbolic, the Euclidean space $\mathbb{R}^n$, n is the space dimension, is not $\delta-$hyperbolic ($\delta = \infty$) while the Poincaré ball $\mathbb{B}^n$ (Nickel & Kiela, 2017) which is a hyperbolic space is $\delta-$hyperbolic with $\delta = \log(1 + \sqrt{2}) \approx 0.88$ (Khrulkov et al., 2020).

Hyperbolicity of remote sensing scene images In order to validate the hypothesis regarding the hyperbolicity of visual datasets, (Khrulkov et al., 2020) calculated the scale-invariant metric δ_{rel}, also known as relative $\delta-$hyperbolicity, of many natural image datasets such as CIFAR10/100 (Krizhevsky & Hinton, 2009), CUB (Wah et al., 2011) and MiniImageNet (Ravi & Larochelle, 2017), which revealed high degrees of hyperbolicity as reported in Table 1.1 (low δ_{rel} values, the closer to zero the better). δ_{rel} is defined as $\delta_{rel}(M) = \frac{2\delta(M)}{diam(M)}$ where M is the matrix pairwise Gromov products (eq. (1.1)) and $diam(M)$ denotes the maximal pairwise distance (the set diameter).

Table 1.1: The relative delta δ_{rel} values calculated for different natural image datasets. Results are averaged across 10 sub-samples of size 1000. The standard deviation for all the experiments did not exceed 0.02 (source (Khrulkov et al., 2020)).

Encoder	**Datasets**			
	CIFAR10	CIFAR100	CUB	MiniImageNet
Inception v3	0.25	0.23	0.23	0.21
ResNet34	0.26	0.25	0.25	0.21
VGG19	0.23	0.22	0.23	0.17

As we assume the hyperbolicity of the RS data, we adopt the procedure described in (Khrulkov et al., 2020) and evaluate δ_{rel} for image scene embeddings of various RS scene datasets extracted by various CNNs pre-trained on the ImageNet dataset. In particular, we consider VGG16 (Simonyan & Zisserman, 2015), ResNet18 (He et al., 2016), GoogleNet (Szegedy et al., 2015), DenseNet (Huang et al., 2017) and SqueezeNet (Iandola et al., 2016). Table 1.2 highlights the obtained δ_{rel} values for the five RS datasets including UCMerced, WHU-RS19, NWPU-RESISC45, AID and PatternNet. We observe that the δ_{rel} values derived from the scene image datasets are closer to 0 than to 1 which results in a rather high degree of hyperbolicity, thus suggesting that hyperbolic representations of scene images can benefit the classification task.

Table 1.2: The relative delta δ_{rel} values calculated for different remote sensing image scene datasets. For each dataset, we measured the Euclidean distance between the features produced by various standard feature extractors pre-trained on ImageNet. Values of δ_{rel} closer to 0 indicate a stronger hyperbolicity of a dataset. Results are averaged across 10 sub-samples of size 1500.

Dataset	VGG16	ResNet18	GoogleNet	DenseNet	SqueezeNet
UCMerced	0.23 ± 0.01	0.26 ± 0.01	0.25 ± 0.01	0.25 ± 0.02	0.28 ± 0.03
WHU-RS19	0.22 ± 0.01	0.27 ± 0.01	0.25 ± 0.02	0.24 ± 0.01	0.31 ± 0.02
NWPU-RESISC45	0.23 ± 0.01	0.28 ± 0.01	0.24 ± 0.01	0.25 ± 0.01	0.31 ± 0.03
AID	0.23 ± 0.01	0.27 ± 0.01	0.23 ± 0.01	0.26 ± 0.01	0.31 ± 0.02
PatternNet	0.20 ± 0.01	0.27 ± 0.01	0.25 ± 0.02	0.25 ± 0.01	0.28 ± 0.02

1.3 Objectives of the Thesis

The focus of this thesis is to investigate the impact of hierarchical information which may be either explicitly or implicitly available in remote sensing data, in particular between scene classes, on feature extraction and classification methods. In this perspective, we thus address the following research questions:

- **Can class hierarchy be relevant for remote sensing scene classification?**
- **Is the hyperbolic space more suited to represent remote sensing data than the Euclidean space, in particular for scene images?**

During my PhD, I contributed to the two aspects mentioned above: 1- the explicit introduction of hierarchical information during the learning of a deep network, 2- the embedding of the scene images into the hyperbolic space which further highlights the underlying hierarchy among scene classes.

1.4 Outline of the thesis

This manuscript is organised as follows:

- Chapter 2 provides a review of relevant studies carried out within the machine learning community that focus on leveraging hierarchical information about the data during the learning process, whether it is explicit or implicit. Additionally, a review of related researches conducted in the remote sensing community is presented. Moreover, several concepts that will be employed throughout the thesis, such as distance and mean calculations in the hyperbolic space, and various evaluation metrics are also defined.

- Chapter 3 focuses on the first type of hierarchical information, namely explicit information conveyed through a class hierarchy that reflects the semantic relations between scene classes. In this context, we propose to introduce this information in two different settings. In the first one, we present the label-driven VAE for scene embedding, which utilises the soft local ranking loss to incorporate class hierarchy information. We then evaluate the performance of the approach by assessing the quality of the VAE latent space. In the second setting, we tackle few-shot learning by proposing a hierarchical prototypical network, which establishes prototypes at every level of the class hierarchy. Then, the class hierarchy information is incorporated through hierarchical prototypes. The hierarchical prototypical network provides regularisation of the latent space and outperforms the classical counterpart.

- Chapter 4 is devoted to the implicit hierarchical information that can potentially exist among scene images. In this respect, we propose to adopt hyperbolic space as an embedding space, as it has been shown (Nickel & Kiela, 2017) to be better suited than Euclidean space for embedding data with an underlying hierarchy. We adopt this hyperbolic space within the same frameworks as in Chapter 3, namely the variational auto-encoder and the prototypical network. Both frameworks were evaluated on a remote sensing scene image classification task and demonstrated the superiority of the hyperbolic space with respect to the Euclidean space.

- We summarise our contributions and discuss potential avenues for future research in Chapter 5, thereby concluding this thesis.

1.5 List of publications

The work presented in this manuscript has led to the following publications:

Journal article

(1) Hamzaoui, M., Chapel, L., Pham, M. T., & Lefèvre, S., *Hyperbolic Prototypical Network for Few Shot Remote Sensing Scene Classification*, Pattern Recognition Letters, Under review.

Conference articles

(1) Hamzaoui, M., Chapel, L., Pham, M. T., & Lefèvre, S., (2023), *Hyperbolic variational auto-encoder for remote sensing scene embeddings*, IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Accepted for oral presentation.

(2) Hamzaoui, M., Chapel, L., Pham, M. T., & Lefèvre, S., (2022), *A hierarchical prototypical network for few-shot remote sensing scene classification*, The 3rd International Conference on Pattern Recognition and Artificial Intelligence (ICPRAI), Paris, France, June 1-3, 2022. [Best paper award]

(3) Hamzaoui, M., Chapel, L., Pham, M. T., & Lefèvre, S. (2021). *Hyperbolic Variational Auto-Encoder for Remote Sensing Scene Classification*. In Journées Francophones des Jeunes Chercheurs en Vision par Ordinateur (ORASIS).

Chapter 2

State-of-the-art: Learning with a class hierarchy in machine learning

In this chapter, we review recent and popular ML approaches that have considered hierarchical information during the learning process, either explicitly when the class hierarchy is available or implicitly through hyperbolic space. We first introduce approaches which explicitly incorporate the class hierarchy in Section 2.1 through the network, the label-embedding or the loss function. Then, in Section 2.2, we provide some basics of hyperbolic geometry and some relevant studies that are helpful in our research. Finally, in Section 2.3, we present studies in remote sensing scene analysis that have incorporated hierarchical information into the learning process. Section 2.4 describes relevant metrics used to evaluate hierarchical approaches.

2.1 Learning strategies with a predefined class hierarchy

Hierarchical information tends to be available as prior knowledge in the form of coarse or most abstract labels with subsequent levels representing more specific categories (fine-grained labels) (Garg et al., 2022). These different classes are organised into a hierarchy of classes, usually a tree or a DAG (Direct Acyclic Graph). Exploiting this hierarchical knowledge has been proven to be effective for many machine learning tasks, such as text classification (Xu & Du, 2020), object recognition (Marszalek & Schmid, 2007), retrieval (Barz & Denzler, 2019; Ramzi et al., 2022), and mistake severity reduction (Bertinetto et al., 2020). Approaches which explicitly consider the class hierarchy during learning are mainly related to three lines of research:

- Hierarchical architectures: approaches of this category change the original model's architecture according to the class hierarchy to learn to recognise the classes at different levels.
- Label-embedding: label-embedding methods convert the discrete label space into a continuous space based on the label relationships given by the class hierarchy.
- Hierarchical losses: this category alters the loss function by giving more weight to specific categories in the class hierarchy.

In this section, we will focus on methods which explicitly incorporate the information given by the class hierarchy to tackle the image classification/retrieval problems. Therefore, we give a brief review of the three categories identified above as well as examples which we consider relevant for a better understanding of each category.

2.1.1 Hierarchical networks

These methods attempt to incorporate the class hierarchy into the classifier architecture without necessarily modifying the loss function. The main idea is to create a classification tree whose internal nodes capture the most easily discriminated and relevant concepts (general classes), which can be transferred to lower level nodes that are more difficult to identify. Numerous studies have been conducted in this regard, including the work by (Zhu & Bain, 2017). In this study, the authors introduced a CNN called B-CNN (Branch Convolutional Neural Network) which incorporates prior knowledge of hierarchical category relations. By going through the network layers, the B-CNN produces, at predefined levels, several predictions ordered from coarse to fine, reflecting the hierarchical structure of the target classes. The H-CNN (Hierarchical Convolutional Neural Network) (Seo & Shin, 2019), which is another type of branching CNN, was later proposed as a derivative of the B-CNN model to predict a class hierarchy for a fashion image data set. Indeed, H-CNN can be considered as a particular implementation of B-CNN, customised for fashion images, and characterised by three prediction branches. H-CNN and B-CNN share similar architecture and training algorithm, and hence, may also encounter similar training challenges, such as the relevance of pre-selected indices matching prediction blocks to hierarchical class levels - i.e., the placement of the prediction branches per granularity level is not obvious. (Kolisnik et al., 2021) extended the B-CNN model by introducing a Condition-CNN, which incorporates conditional probability to capture class relationships for hierarchical

classification. In contrast to the B-CNN approach, which incorporates prediction branches along the feature extractor blocks, the Condition-CNN defines prediction branches in parallel after the common feature extraction step.

The effectiveness of hierarchical neural networks (HNNs) highlights the potential of using hierarchical architectures in solving complex tasks. However, these approaches primarily focus on enhancing the prediction accuracy of fine-grained classes (Mayouf & de Saint-Cyr, 2022). Our aim, on the other hand, is to ensure hierarchical consistency and enhance accuracy for both coarser-class and fine-grained class predictions, which differs from the primary objective of the aforementioned approaches. Furthermore, the HNN architecture is often domain-dependent (Goyal et al., 2021; Taoufiq et al., 2020), requiring careful design and proper branch point selection to ensure that the HNN is able to effectively capture the hierarchical relationships within the particular domain. This may require extensive experimentation and tuning to determine the optimal branch point placement to achieve the best possible HNN performance.

2.1.2 Label-embedding

An alternative approach to incorporate the class hierarchy is through label-embedding methods, which can effectively facilitate knowledge sharing among classes. In contrast to the previous category, label-embedding approaches do not necessarily require specific architecture modifications. Such methods define each class as a soft embedding vector instead of the typical one-hot vector by using a mapping function to associate classes with representations to encode information about the inter-class relationships.

(Frome et al., 2013) proposed a deep visual-semantic embedding Model (DeViSE) which is a visual object recognition approach involving semantic class information. They suggested to map the target classes to a unit hyper-sphere by embedding them using a word2vec model pre-trained on Wikipedia. Another hierarchy-based method was suggested by (Barz & Denzler, 2019), in which they incorporate the hierarchical relationships of classes into the hyper-sphere in order to learn semantically discriminative features for hierarchical retrieval. They thus introduced an embedding algorithm such that all inter-class distances represent similarities derived from the height of the lowest common ancestor (LCA) given a class hierarchy. In the same direction, the authors in (Bertinetto et al., 2020) have recently proposed an interesting approach to incorporate the class hierarchy into the training of a deep classifier in a standard supervised setup. This approach, known as *soft labels*, applies a mapping function to encode the class-relationship information,

producing a categorical distribution over the classes. The class-relationship information is derived from inter-class distances based on LCA (Lowest Common Ancestor). The standard cross-entropy loss is employed to train the classifier, however, the soft label is employed instead of the one-hot label.

Although label-embedding is considered a promising approach for capturing correlations across hierarchical classes, in this thesis, we attempt to incorporate the class hierarchy via the hierarchical loss approaches that will be described below to avoid being dependent on the embedding strategy.

2.1.3 Hierarchical losses

The remaining direction to leverage the class hierarchy is through the loss function. Modifying the optimised loss function is a reasonable and interesting approach to incorporate the class hierarchy when training the model. Numerous studies on hierarchical classification/retrieval have adopted this strategy. In these methods, the loss function is parameterised by the class hierarchy, thus classifying an image into a different but semantically close category results in a lower loss than classifying it into a semantically distant category, implying a higher penalty when predicting a more distant relative of the true class.

(Bertinetto et al., 2020) have recently proposed a hierarchical loss which adapts the well-known cross-entropy loss. The hierarchical cross-entropy loss (HXE) considers the class hierarchy through the definition of a conditional distribution along the path connecting each class node to the root node of the class hierarchy. (Ramzi et al., 2022) modified the average precision measure by introducing the class hierarchy and proposed HAPPIER, a hierarchical average precision learning approach for image retrieval. This approach is based on the hierarchical rank, H-rank, which introduces a soft penalty for instances that do not share the same fine-grained class as the query yet share more general semantics. Some studies adopted pair-based losses, such as hierarchical triplet loss (Ge et al., 2018). The authors suggested that the class hierarchy could be used to sample more reasonable hard triplets as well as controlling the degree of aggregation/separation between samples of different classes. Similarly, (Yu et al., 2020) used the class hierarchy to parameterise the sampling strategy of the local ranking loss function which is typically utilised for aligning preferences (Goyal et al., 2021). The authors in (Yang et al., 2022c) introduced a novel approach to enhance the retrieval performance of deep metric models. Specifically, they proposed a hierarchical proxy-based loss (HPL) that leverages the hierarchical relationships

among classes. The method involves training finer-level proxies, which are proxies of finer classes, as part of the embedding network parameters. The authors then extended this idea to learn higher-level proxies, which are proxies of coarser classes. If a class hierarchy is available, the higher-level proxies are learned following the same approach as the finer proxies. Alternatively, pseudo super-classes are created by clustering the lower level proxies using an unsupervised clustering algorithm (such as k-means), and the cluster centroids are used as higher-level proxies. The model is then optimised using a weighted proxy-loss over the different levels of the class hierarchy.

Hierarchical loss-based approaches are straightforward strategies to utilise the class hierarchy in the context of hierarchical classification tasks. These approaches enable a *flat* network to perform tasks that require the support of a class hierarchy. However, it is worth noting that designing an effective hierarchical loss function can be challenging. The choice of loss function, as well as its hyper-parameters, can greatly impact the performance of the model. Careful consideration needs to be given to the design of the loss function to ensure that it adequately captures the hierarchical relationships among classes. In this thesis, we attempt to leverage the class hierarchy through the loss function.

2.2 Hyperbolic geometry for data embedding

As discussed earlier, the label information can be expressed explicitly as a hierarchical arrangement of image classes which reflect the semantic relationships between the different categories. This information can be explicitly taken into account according to different approaches as presented above. Alternatively, the relationships between image labels may not be explicitly provided by a class hierarchy, yet this information still exists. Recent studies (Nickel & Kiela, 2017; Peng et al., 2022) have suggested working in a non-Euclidean space, specifically hyperbolic space, which has been shown to be well suited to hierarchical data or data with an underlying hierarchy.

In this section, we will focus on hyperbolic space. We will provide some mathematical background and review several machine learning studies in which these spaces have been used to represent hierarchical data or data with an underlying hierarchy, as well as a brief overview of strategies used for optimisation in these spaces.

2.2.1 Hyperbolic geometry

Different curvatures of Riemannian manifolds yield different geometries: Euclidean, which has zero curvature, Elliptic, which has constant positive curvature, and Hyperbolic, which has constant negative curvature (Figure 2.1); curvature essentially measures how far manifolds deviate from *flat* Euclidean space (Peng et al., 2022).

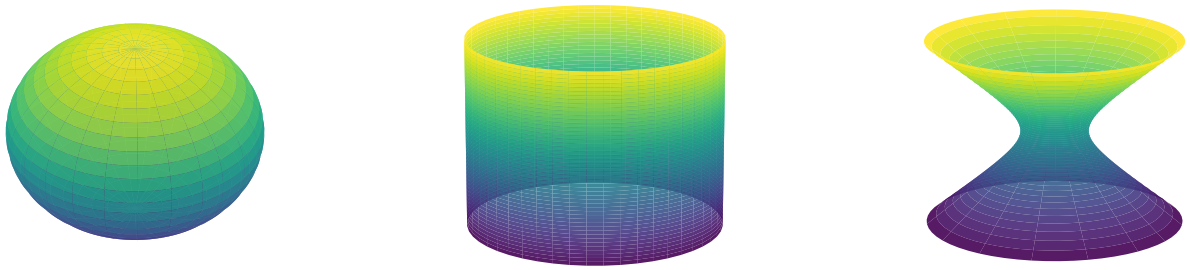


Figure 2.1: Illustration of spherical, Euclidean and hyperbolic spaces. From left to right: a surface of positive curvature (sphere), a surface of zero curvature (cylinder), and a surface of negative curvature (hyperboloid).

In this part, we will focus on the hyperbolic space. There are five models of hyperbolic space, each with a unique set of properties (Peng et al., 2022). These models are isometric, which makes it easy to switch from one model to another. Among these, we are interested in the two following models: the Poincaré Ball model and the Lorentz model which have been most used in recent machine learning studies involving hyperbolic geometry (Peng et al., 2022). The Poincaré Ball model represents the infinite hyperbolic space in a finite ball. Its volume increases exponentially in proportion to its radius, leading to two main strengths that make it well-suited for dealing with hierarchical data or data with an underlying hierarchy. First, this exponential growth property closely matches the growth rate of the tree data, resulting in a space with minimal distortion that fits hierarchies particularly well, unlike the Euclidean space. The second point to note is that it is capable of producing remarkably high-quality representations at low-dimensional embedding space. This makes it particularly advantageous in situations where memory and storage resources are limited (Yang et al., 2022b). Apart from its benefits in representing data, it is very useful for visualisation. The Lorentz model (or hyperboloid model) refers to the upper half (positive sheet) of a two-sheet hyperboloid. This model provides a relatively simple geodesic formula and is therefore useful for computation. We briefly review the two models and refer to the reference texts (Peng et al., 2022) for more details on the fundamentals of hyperbolic geometry and these two models.

Poincaré Ball model

The Poincaré Ball model (Ganea et al., 2018; Nickel & Kiela, 2017) $(\mathbb{B}^d, g^{\mathbb{B}})$ is a Riemannian manifold defined by the open $d-$dimensional ball of radius $\frac{1}{\sqrt{c}}$, $\mathbb{B}_c^d = \{x \in \mathbb{R}^d : c\|x\|^2 < 1\}$, where $k = -c$ $(c > 0)$ is the negative curvature of the Hyperbolic space which measures how far the manifold deviates from the flat Euclidean space, $\|.\|$ is the Euclidean norm. The Poincaré Ball model is endowed with the Riemannian metric tensor $g^{\mathbb{B}_c^d}(x) = \lambda_x^{c\,2} g^{\mathbb{E}}$, where $x \in \mathbb{B}_c^d$, $\lambda_x = \frac{1}{1-c\|x\|^2}$ is the conformal factor and $g^{\mathbb{E}} = \mathbb{I}^d$ denotes the Euclidean metric tensor. The figure 2.2 shows an example of two–dimensional Poincaré Ball model as well as some operations in this space.

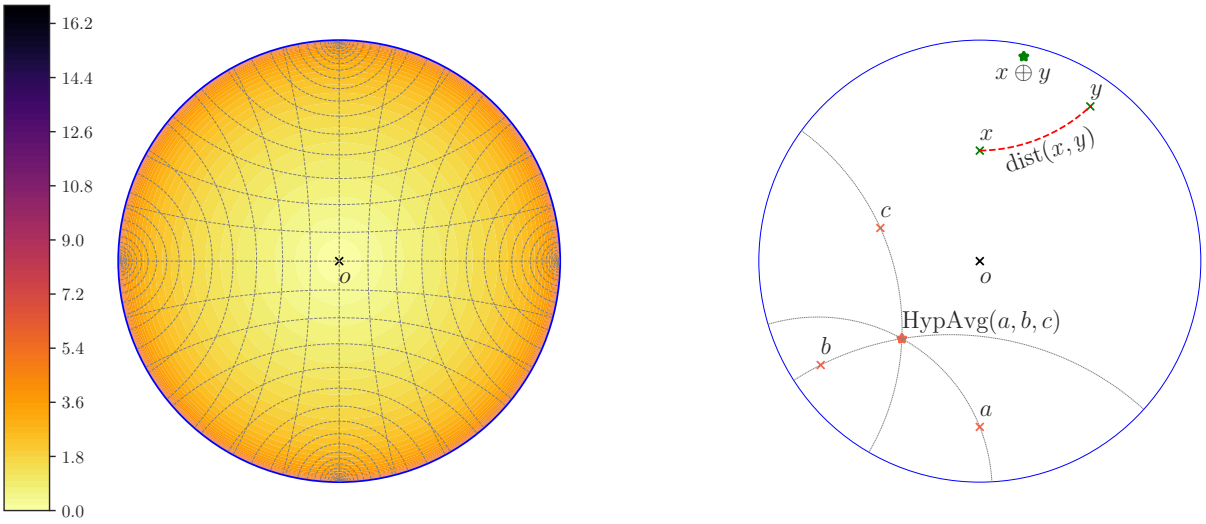


Figure 2.2: Two–dimensional Poincaré Ball model with negative curvature $k = -1$. **(left)** Grid of Geodesics and the geodesic distance to the origin. **(right)** Point $x \oplus y$ represents the Möbius sum of points x and y. *HypAvg* stands for the hyperbolic average of the points a, b and c. The dashed red line represents the shortest distance between x and y.

Distance The shortest path between two points $x, y \in \mathbb{B}_c^d$ is given by the geodesic distance defined as:

$$d_{\mathbb{B}}(x, y) = \frac{2}{\sqrt{c}} \operatorname{arctanh}\left(\sqrt{c}\ \|-x \oplus_c y\|\right) \tag{2.1}$$

where $\oplus$ is Möbius addition defined as follows:

$$x \oplus_c y = \frac{\left(1 + 2c\,\langle x, y\rangle + c\,\|y\|^2\right) x + \left(1 - c\,\|x\|^2\right) y}{1 + 2c\,\langle x, y\rangle + c^2\,\|x\|^2\,\|y\|^2} \tag{2.2}$$

where $\|.\|$, $\langle ., . \rangle$ are the norm and the scalar product in the Euclidean space, respectively.

Tangent space The tangent space $\mathcal{T}_x\mathbb{B}_c^d$ is a $d-$dimensional vector space which corresponds to a first order linear approximation of $\mathbb{B}_c^d$ around x.

Exponential and logarithmic maps Working in hyperbolic space is not easy: it requires generalising basic operations, such as vector addition, matrix-vector multiplication and vector translation to these spaces, which is not trivial or sometimes even impossible (Ganea et al., 2018). A simple and straightforward way to accomplish this is to move the data from a hyperbolic space to a tangent space, a local Euclidean space in which the operations are constructed as in Euclidean space (Ganea et al., 2018). To switch respectively from and to the hyperbolic space, we first need to define a bijective function from $\mathbb{R}^d$ to $\mathbb{B}_c^d$. This function maps vectors from the Euclidean space to the hyperbolic space, and vice versa.

Formally, the exponential map at x ($exp_x^c : \mathcal{T}_x\mathbb{B}_c^d \cong \mathbb{R}^d \to \mathbb{B}_c^d$) maps an Euclidean tangent vector $v \in \mathcal{T}_x\mathbb{B}_c^d$ onto $\mathbb{B}_c^d$ and it is defined as:

$$\exp_x^c(v) = x \oplus_c \left(\tanh\left(\sqrt{c} \frac{\lambda_x^c \|v\|}{2} \right) \frac{v}{\sqrt{c}\|v\|} \right) \tag{2.3}$$

The logarithmic map at x ($log_x^c : \mathbb{B}_c^d \to \mathcal{T}_x\mathbb{B}_c^d$) has an inverse role and maps points $u \in \mathbb{B}_c^d$ to the tangent space at x $\mathcal{T}_x\mathbb{B}_c^d$ following:

$$\log_x^c(u) = \frac{2}{\sqrt{c}\lambda_x^c} \operatorname{arctanh}\left(\sqrt{c} \|-x \oplus_c u\| \right) \frac{-x \oplus_c u}{\|-x \oplus_c u\|} \tag{2.4}$$

In practice, we use the exponential and logarithmic maps at the origin, denoted by $\exp_0^c$ and $\log_0^c$ respectively, to move from Euclidean space to the Poincaré Ball.

Lorentz model

The Lorentz model (Chen et al., 2022b; Nickel & Kiela, 2018) also known as the Hyperboloid model is one of the typical hyperbolic models that refers to the upper sheet of a two-sheet $d-$dimensional hyperboloid (Figure 2.3). Formally, the Lorentz model is a Riemannian manifold with negative curvature $k = -c$ ($c > 0$) defined as $\mathcal{L}_c^d = (\mathcal{H}_c^d, g_l)$, where g_l is the Riemannian metric tensor and $\mathcal{H}_c^d$ denotes the upper sheet of a two sheet d-dimensional

hyperboloid:

$$\mathcal{H}_c^d = \{x \in \mathbb{R}^{d+1} : c\langle x, x\rangle_{\mathcal{L}} = -1,\ x_0 > 0\} \tag{2.5}$$

where $\langle ., . \rangle_{\mathcal{L}}$ is the Lorentzian inner product, also known as the metric tensor, defined as:

$$\langle x, y\rangle_{\mathcal{L}} = -x_0 y_0 + \sum_{i=1}^{d} x_i y_i \tag{2.6}$$

for points $x, y \in \mathbb{R}^{d+1}$.

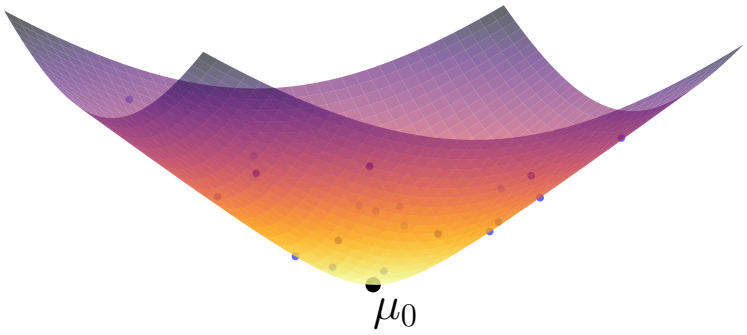


Figure 2.3: Two–dimensional Lorentz model with negative curvature $k = -1$, $\mu_0 = [1, 0, 0]$ is the origin of hyperbolic space.

We note that for any point $x = (x_0, x') \in \mathbb{R}^{d+1}$

$$x \in \mathcal{H}_c^d \Leftrightarrow x_0 = \sqrt{\frac{1}{c} + ||x'||^2} \tag{2.7}$$

The origin of hyperbolic space is referred as a vector $\mu_0 = \left[\frac{1}{\sqrt{c}}, 0, \ldots, 0\right] \in \mathcal{H}_c^d$.

Geodesic distance The shortest path between two points $x, y \in \mathcal{H}_c^d$ is given by a relatively simple formula of the geodesic distance and is defined as:

$$d_{\mathcal{L}}^c(x, y) = \frac{1}{\sqrt{c}} \operatorname{arcosh}(-c\ \langle x, y\rangle_{\mathcal{L}}) \tag{2.8}$$

Tangent space The tangent space at $x \in \mathcal{H}_c^d$ ($\mathcal{T}_x\mathcal{H}_c^d$) can be described as a subspace of $\mathbb{R}^{d+1}$. It is represented by a set of points $v \in \mathbb{R}^{d+1}$ satisfying the orthogonality relation with respect to the Lorentzian product:

$$\mathcal{T}_x\mathcal{H}_c^d = \{v \in \mathbb{R}^{d+1} \mid \langle v, x\rangle_{\mathcal{L}} = 0\} \tag{2.9}$$

Note that $\mathcal{T}_{\mu_0}\mathcal{H}^d_c$, the tangent space at the origin, consists of points $u \in \mathbb{R}^{d+1}$ with $v_0 = 0$ and $\|v\|_{\mathcal{L}} = \sqrt{\langle v, v\rangle_{\mathcal{L}}} = \|v\|$.

Exponential and logarithmic maps Exponential map projects a tangent space vector $v \in \mathcal{T}_x\mathcal{H}^d_c$ onto the hyperbolic space $\mathcal{H}^d_c$ (see Figure 2.4 for illustration). It is defined locally and only projects a small neighbourhood of the tangent space origin x onto its neighbourhood in the hyperbolic space. The exponential map of the Lorentz model is then given by:

$$\begin{aligned} \exp^c_x &: \mathcal{T}_x\mathcal{H}^d_c \to \mathcal{H}^d_c \\ exp^c_x(v) &= \cosh\left(\sqrt{c}\,\|v\|_{\mathcal{L}}\right) x + \frac{1}{\sqrt{c}}\,\sinh\left(\sqrt{c}\,\|v\|_{\mathcal{L}}\right)\,\frac{v}{\|v\|_{\mathcal{L}}} \end{aligned} \tag{2.10}$$

The logarithmic map, also known as the inverse exponential map, is defined for $u, x \in \mathcal{H}^d_c$ as:

$$\begin{aligned} \log^c_x &: \mathcal{H}^d_c \to \mathcal{T}_x\mathcal{H}^d_c \\ \log^c_x(u) &= \left(\exp^c_x\right)^{-1}(u) = d^c_{\mathcal{L}}(x,u)\frac{u + c\,\langle x, u\rangle_{\mathcal{L}}\,x}{\|u + c\,\langle x, u\rangle_{\mathcal{L}}\,x\|} \end{aligned} \tag{2.11}$$

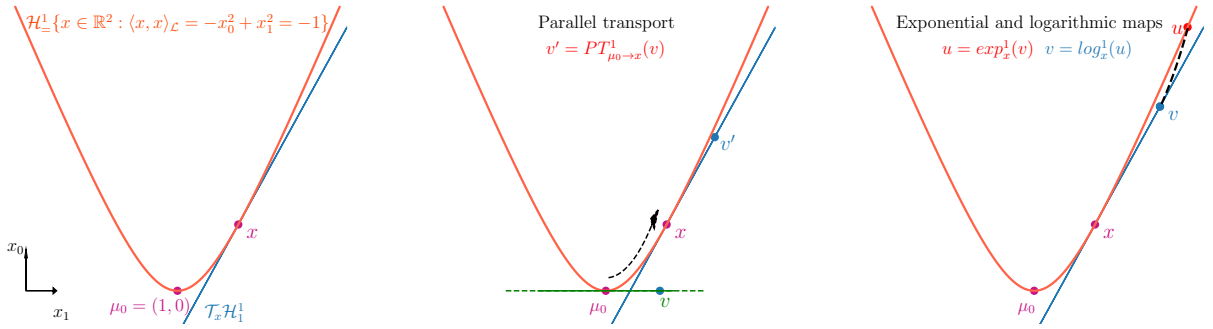


Figure 2.4: From left to right: **(Left)** The one-dimensional Lorentz model $\mathcal{H}^1_1$ (orange) and the tangent space at x $\mathcal{T}_x\mathcal{H}^1_1$ (blue). **(Centre)** Parallel transport that carries $v \in \mathcal{T}_{\mu_0}\mathcal{H}^1_1$ (green) to $v' \in \mathcal{T}_{\S}\mathcal{H}^1_1$ (blue). **(Right)** Exponential map projects the $v \in \mathcal{T}_x\mathcal{H}^1_1$ (blue) to $u \in \mathcal{H}^1_1$(red).

Parallel transport For any couple of points $x, y \in \mathcal{H}^d_c$, parallel transport from x to y ($\mathrm{PT}^c_{x\to y}$) is a map that carries a vector $v \in \mathcal{T}_x\mathcal{H}^d_c$ along the geodesic to their corresponding

vector $v' \in \mathcal{T}_y\mathcal{H}^d_c$ while preserving its metric tensor i.e. $\langle \mathrm{PT}^c_{x\to y}(v), \mathrm{PT}^c_{x\to y}(v')\rangle_{\mathcal{L}} = \langle v, v'\rangle_{\mathcal{L}}$ (see Figure 2.4 for illustration). For the Lorentz model, this map is given by:

$$\mathrm{PT}^c_{x\to y}(v) = v - \frac{\langle \log^c_x(y), v\rangle_{\mathcal{L}}}{d^c_{\mathcal{L}}(x,y)^2}\left(\log^c_x(y) + \log^c_y(x)\right) \tag{2.12}$$

The inverse parallel transport $(\mathrm{PT}^c_{x\to y})^{-1}$ simply carries back the vector in $\mathcal{T}_y\mathcal{H}^d_c$ to $\mathcal{T}_x\mathcal{H}^d_c$ along the geodesic and is defined as:

$$v = (\mathrm{PT}^c_{x\to y})^{-1}(v') = \mathrm{PT}^c_{y\to x}(v') \tag{2.13}$$

Switching between Poincaré Ball and Lorentz models The five models of hyperbolic geometry are isometric, which makes it easy to switch from one model to another. We can thus move from and to the Poincaré Ball model to the Lorentz model via the following equations:

$$\Pi_{\mathcal{H}^d_c\to\mathbb{B}^d_c}(x_0,\ldots,x_d) = \frac{\sqrt{c}}{1+\sqrt{c}x_0}(x_1,\ldots,x_d), \quad x = [x_0,\ldots,x_d]^T \in \mathcal{H}^d_c \tag{2.14}$$

$$\Pi_{\mathbb{B}^d_c\to\mathcal{H}^d_c}(x_1,\ldots,x_d) = \frac{(1+\|x\|^2, 2x_1,\ldots,2x_d)}{\sqrt{c}\,(1-\|x\|^2)}, \quad x = [x_1,\ldots,x_d]^T \in \mathbb{B}^d_c \tag{2.15}$$

Mean in the hyperbolic space

The mean calculation, simple yet valuable, is one of the key operations in machine learning approaches. The path to extend this weighted mean calculation to hyperbolic space is much less obvious. The averaging cannot be accomplished simply by averaging the vectors since this does not guarantee that the resulting mean is always on the manifold. Theoretically, we can generalise the mean calculation to the hyperbolic space thanks to one of three approaches: the Fréchet mean method (Fréchet, 1948), the tangent space aggregation (Chami et al., 2019) and the Einstein midpoint method (Peng et al., 2022).

An equivalent of the Euclidean mean in hyperbolic space is the Fréchet mean (Fréchet, 1948), which, however, has no closed-form solution. Solving the Fréchet mean currently requires an iterative computation which significantly slows down learning and inference. Alternatively, tangent aggregation is a simpler method to compute the average in hyperbolic

space, which is in fact the same as the simple average in Euclidean space. However, rather than approximating the mean in tangent space, the hyperbolic mean can be calculated using the Einstein midpoint which is an extension of the mean operation to hyperbolic space, having the simplest form with Klein[1] coordinates:

$$\text{HypAve}(x_1, \ldots, x_n) = \sum_{i=1}^{n} \gamma_i x_i / \sum_{i=1}^{n} \gamma_i \tag{2.16}$$

where x_i are embeddings in Klein model $\mathbb{K}_c^d$ and $\gamma_i = \frac{1}{\sqrt{1-c\|x_i\|^2}}$ are the Lorentz factors.

We can therefore easily compute the hyperbolic mean by simply projecting to and from the Klein model to various hyperbolic space models as all are isomorphic. The transfer to and from the the Klein model to the Poincaré Ball model is thus carried out via the two matching functions:

$$\Pi_{\mathbb{B}_c^d \to \mathbb{K}_c^d}(x) = \frac{2x}{1 + c\,\|x\|}, \quad x \in \mathbb{B}_c^d \tag{2.17}$$

$$\Pi_{\mathbb{K}_c^d \to \mathbb{B}_c^d}(x) = \frac{x}{1 + \sqrt{1 - c\,\|x\|^2}}, \quad x \in \mathbb{K}_c^d \tag{2.18}$$

Similarly, the transfer to and from the Klein model to the Lorentz model is accomplished by:

$$\Pi_{\mathcal{H}_c^d \to \mathbb{K}_c^d}(x_0, \ldots, x_d) = \left(\frac{x_1, \ldots, x_d}{\sqrt{c} x_0}\right), \quad x = [x_0, \ldots, x_d]^T \in \mathcal{H}_c^d \tag{2.19}$$

$$\Pi_{\mathbb{K}_c^d \to \mathcal{H}_c^d}(x_1, \ldots, x_d) = \frac{1}{\sqrt{1 - c\|x\|^2}} \left(\frac{1}{c}, x_1, \ldots, x_d\right), \quad x = [x_1, \ldots, x_d]^T \in \mathbb{K}_c^d \tag{2.20}$$

2.2.2 Hyperbolic geometry in machine learning

Although hyperbolic space has been known since the 19th century, it has rarely been used in machine learning (ML) despite its attractive theoretical properties. Traditionally, ML researches have focused mainly on approaches operating in Euclidean space, with less interest in other spaces, regardless of the nature of the data being manipulated and their specificities. However, this trend has begun to change following the publication of (Nickel & Kiela, 2017) (as depicted in Figure 2.5). The authors propose to embed data with a

1. The Klein model is one of the five models of hyperbolic geometry. Similarly to the Poincaré Ball model, it is defined on the set $\mathbb{K}_c^d = \{x \in \mathbb{R}^d : c\,\|x\|^2 < 1\}$, however, with a different metric.

latent hierarchy, more specifically graphs, in Poincaré model rather than Euclidean space. They provided evidence to support the superiority of this space in learning high-quality embeddings.

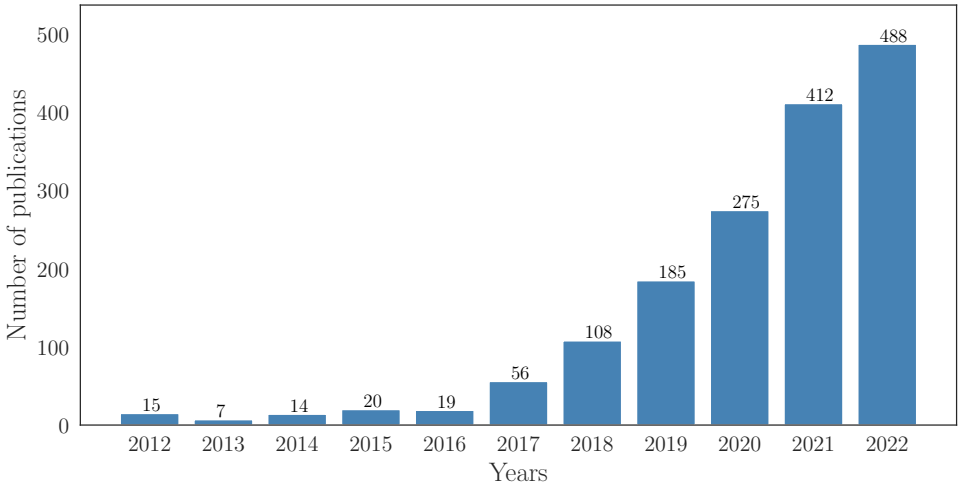


Figure 2.5: Number of publications in machine learning from 2012 to 2022. Data from Google scholar advanced search: "hyperbolic space" and "machine learning"

This paper drew the attention of the ML community to the hyperbolic space and significantly advanced researches which seek to better represent data with a latent hierarchy in several applications such as word embedding (Tifrea et al., 2019), text classification (Zhu et al., 2020), text generation (Dai et al., 2021), node classification and link prediction (Chami et al., 2019). As such, many recent works use the hyperbolic space to learn data representations, and various machine learning methods have been adapted to this framework. Among them, we can mention the hyperbolic SVM (Cho et al., 2019) or the hyperbolic neural network (Ganea et al., 2018). Other studies have provided a generalisation of normal distributions on hyperbolic space that can be used to build and learn a probabilistic model like Variational Auto-Encoder (VAE) (Mathieu et al., 2019; Nagano et al., 2019). The proposed hyperbolic VAEs (H-VAE) are among the earliest studies dealing with images in a hyperbolic space. They were used to embed images in a hyperbolic latent space and then infer the underlying hierarchical structure. These methods have been validated on MNIST and Atari 2600 Breakout datasets, showing that the H-VAE is able to retrieve their hierarchical nature. However, those datasets are simple and do not reflect complex scenario as in real-world images. Moreover, the MNIST dataset is not hierarchical whereas real-world images can show hierarchical structures, either within the image (Cui et al., 2015), or between the images, when a hierarchy of classes is available (Dhall et al., 2020;

Khrulkov et al., 2020; Liu et al., 2020b). Given this setting, an interesting study (Yu et al., 2020) was suggesting to rather guide the VAE learning in order to drive the construction of its latent space such that it reflects a given class hierarchy.

Inspired from these two studies, the computer vision community was no exception and followed the trend of the machine learning community, as hierarchical information can also be found in image data, although this remains in progress and has not yet been widely applied. The study proposed in (Khrulkov et al., 2020) is considered one of the pioneering papers in the image community. The authors provide a methodology for image embedding in hyperbolic space and have evaluated their method when solving the person identification and a few-shot image classification tasks, showing the benefits of these spaces and their superiority. Further studies have targeted other applications such as image classification (Atigh et al., 2021; Ermolov et al., 2022; Guo et al., 2022b) and semantic segmentation. However, some studies have introduced additional information such as the class hierarchy to drive the learning process and further benefit from the geometric properties of the hyperbolic space, tackling challenges such as object detection (Lang et al., 2022), semantic segmentation (Li et al., 2022a) or zero-shot classification (Liu et al., 2020a).

2.2.3 Optimisation in the hyperbolic space

In most machine learning applications, learning is performed in a Euclidean space, mainly because of its convenient mathematical properties, especially those required for constructing deep neural networks, such as vector structures or closed forms for distance calculation. However, this is rather less obvious when dealing with the hyperbolic space, considering the difficulty of expanding the hyperbolic counterparts of deep neural architectures. Optimisation in these Riemannian spaces is not trivial either and is challenging, as is the construction of neural networks. Considering these challenges, the development of Riemannian models and optimisation techniques in this hyperbolic space is rather not as efficient as in Euclidean space, which results in relatively limited resources to be used by the computer vision community, until very recently.

A pioneering study was presented in (Ganea et al., 2018) which provides a framework for reasonably generalising classical Euclidean deep learning tools to hyperbolic space. Accordingly, numerous recent researches have suggested alternative approaches to generalise deep learning operations to the hyperbolic space, as this represents a major step towards hyperbolic deep neural networks. Therefore, we can consider two possible approaches

to generalise neural networks to the hyperbolic space. The first, which is simple and straightforward, requires transferring the data from hyperbolic space to tangent space and then applying traditional operations as used in Euclidean networks. The second alternative, which is less obvious, is to construct the network directly in hyperbolic space. In this regard, we point to some studies that started to explore this aspect: linear layers (Shimizu et al., 2021), softmax layer (Ganea et al., 2018) and activation functions (Ganea et al., 2018).

Early hyperbolic neural networks (Ganea et al., 2018) adopted the Poincaré Ball model due to its ability to have an infinite space in a finite one. The Poincaré Ball's Radius, defined as $\frac{1}{\sqrt{c}}$, results in a finite space when $c > 0$. However, it's worth noting that while the boundary of the ball doesn't belong to the hyperbolic space, it represents points that are infinitely distant. This hyperbolic space expands exponentially as the radius of the ball increases, making it an ideal choice for representing trees where the number of children increases exponentially as they move away from the root of the tree. This property allows for an effective representation of hierarchical data. However, this exponential growth property sometimes leads to numerical instability when the hyperbolic embeddings get too close to the boundaries of the Poincaré model, resulting in values that are unrepresentable in floating point arithmetic and lead to undefined distances (NaN values). To avoid this problem, some studies such as (Chami et al., 2019; Nagano et al., 2019) have rather adopted the Lorentz model which provides a simpler distance formula.

As for the optimisation of these hyperbolic networks, the study in (Bonnabel, 2013) represents one of the pioneers of stochastic optimisation in Riemannian manifolds in which they introduce the Riemannian stochastic gradient descent (RSGD) optimisation. Further studies have followed the trend of hyperbolic space and have proposed a generalisation of other optimisers to Riemannian manifolds such as RADAM and RAMSGRAD (Sakai & Iiduka, 2022). However, the gradient sometimes vanishes during the optimisation, which is due to the hybrid architecture of the networks that connect Euclidean features to hyperbolic layers (Guo et al., 2022b). A potential solution to this issue could be to adopt fully hyperbolic networks as was done in (Chen et al., 2022b), although this is not yet possible for all varieties of architectures. Indeed, the authors in (Guo et al., 2022b) have suggested that Euclidean features should be clipped before moving to hyperbolic layers, which seems to avoid the vanishing gradient problem.

Through this brief review, we can observe that neural networks are not yet very well established in the hyperbolic space and further work remains to be accomplished in

order to attain a more standard framework as it is in Euclidean space. However, none of this changes the merits of these spaces. As for optimisation in this hyperbolic space, a standard Manifold interface for Riemannian optimisation is provided by the Geoopt framework (Kochurov et al., 2020) which is developed in Pytorch.

2.3 Leveraging class hierarchy for remote sensing scene analysis

In the last decade, deep learning (DL) methods have induced a significant revolution across diverse research areas, with a particular emphasis on computer vision tasks, including, but not limited to, object detection (Lang et al., 2022), image classification (Zhao et al., 2020), semantic segmentation (Castillo-Navarro et al., 2022) and image generation (Han et al., 2020). Typically, these methodologies follow a two-step process, whereby the initial step involves the automated extraction of meaningful patterns from the images. Subsequently, in the second step, these learned representations are applied to downstream tasks that could be supervised, such as classification and object detection, or unsupervised, such as compression, generation, or clustering.

Within the specific context of remote sensing, various tasks such as ship classification (Chen & Qian, 2022), hierarchical object detection (Shin et al., 2020), tree spices classification (Lei et al., 2022), and urban building classification (Taoufiq et al., 2020) have utilised class hierarchies. Typically, such class hierarchies are incorporated into the network architecture, such as in the case of HierarchyNet (Taoufiq et al., 2020), which was inspired from the B-CNN (Zhu & Bain, 2017) and introduced a novel hierarchical network for urban building classification. Alternatively, a combination of hierarchical network and loss function can be used, as demonstrated in (Chen & Qian, 2022). The hierarchical network architecture in this approach includes two output channels. The first channel is organised based on a hierarchy and exclusion (HEX) graph, which models the class hierarchy and encodes semantic relations between classes. The corresponding probabilistic classification loss for this channel reflects the hierarchical structure of the HEX graph. While the second output channel is dedicated to the finest-grained classes, and its multi-class cross-entropy loss is designed to improve the classifier's discriminative power for these classes.

Regarding RS scene analysis, although there exist some hierarchical solutions in the literature, the hierarchy is usually defined based on the clustering of similar features (Sen & Keles, 2022b) such as (Goel et al., 2019; Liu et al., 2019). Newer studies, such as (Guo

et al., 2022a; Liu et al., 2020b; Sen & Keles, 2022b; Zeng et al., 2022), which focus on scene classification and retrieval tasks, have considered a predefined class hierarchy and explicitly incorporated it during the learning process. To the best of our knowledge, these methods appear to belong solely to the first category.

In (Liu et al., 2020b), the authors introduced a triplet network designed for RS scene retrieval. The proposed network utilises the class hierarchy to choose appropriate triplets, consisting of an anchor, a positive sample, and a negative sample, as inputs. Notably, the positive sample does not necessarily need to belong to the same class as the anchor, but rather should be semantically closer to it than the negative sample. Furthermore, the authors leveraged the class hierarchy to parameterise the loss function and enable an adaptive "pull-push" mechanisms. As for (Sen & Keles, 2022b), they described a hierarchical network for scene classification. They introduced a coarser classifier to predict the high-level class category, and fine classifiers were defined for each coarser class in the hierarchy. The selection of the finer classifier to use was based on the output of the coarser classifier, and finer predictions were made using the selected finer classifier. Likewise, the hierarchical network introduced in (Zeng et al., 2022) aimed to tackle the scene classification task. They proposed a single classifier per level, which were learned in parallel. The network learned fine-grained features, which were utilised as inputs for both the fine classifier and coarser classifier, in addition to coarse-grained features. Similarly, (Guo et al., 2022a) introduced a multiple granularity semantic learning network (MGSN), which is a hierarchical network consisting of multiple independent branches, each corresponding to a level in the class hierarchy. The purpose of the MGSN is to leverage various levels of semantic information about scenes and guide the network in learning global and local features simultaneously. However, the authors treat RS scene image classification as a multi-label classification task and opt to learn the different granularity semantics independently and in parallel.

Incorporating class hierarchy into the learning process enables the data embeddings to incorporate valuable information from both class-specific and semantically related classes. This facilitates information transfer between semantically similar classes, leading to improved accuracy values at coarser levels, while simultaneously reducing the severity of mistakes. Furthermore, hierarchical information about the classes can be particularly valuable in situations where there is limited labelled data available for training. By leveraging the hierarchical structure of the classes, the model can effectively transfer knowledge from related classes to those with few or no labelled samples, improving its ability to make accurate predictions even with limited training data (Li et al., 2019; Liu

et al., 2020a).

As far as hyperbolic space is concerned, to our knowledge, it has only been referenced in two studies within the RS community, namely (Li et al., 2022b; Sun et al., 2022). In (Sun et al., 2022), the authors focused on reducing the dimensionality of hyperspectral images (HSI) through an unsupervised approach for selecting more consistent bands. HSI have numerous spectral bands, and not all of them are informative or contribute to the desired classification or analysis task. As such, (Sun et al., 2022) proposed a novel hyperbolic clustering-based band hierarchy (HCBH) approach which aims to select a subset of informative bands for HSI analysis. The HCBH method uses hyperbolic clustering to construct a hierarchy of bands based on their similarity in the hyperbolic space. The resulting hierarchy of bands is then used to select a subset of representative bands based on an adaptive hyperbolic distance of each band to the "origin" of the Poincaré Ball, suggesting that bands closer to the origin are more prominent in their group and capture the most significant spectral information in the HSI. While authors in (Li et al., 2022b) proposed a hybrid attention-enhanced neural network (HAENet) to perform semantic segmentation of RS images. A key component of the HAENet is the similarity hybrid attention module (SHAM) which fuses position-specific attention maps from both Euclidean and hyperbolic spaces.

2.4 Hierarchical evaluation metrics

The performance of classification algorithms can be evaluated according to several criteria. Therefore, different measures usually assess different properties derived from the classification algorithm. This section presents an overview of the evaluation metrics employed to assess the performance of hierarchical classification approaches in the literature. Some of them will be used to evaluate the methods proposed in this thesis.

Accuracy

Accuracy is the most widely used metric to evaluate the effectiveness of classification algorithms. It is calculated as the percentage of correctly predicted samples, i.e. predictions $\hat{y} \in \hat{Y}$ identical to the ground truth labels $y \in \hat{Y}$.

$$Acc = \frac{1}{|Y|} \sum_{(y,\hat{y})} 1\left(\hat{y} = y\right) \tag{2.21}$$

where $x \to 1(x)$ is the indicator function defined as: $1(\hat{y} = y) = 1$ if $\hat{y} = y$ else 0.

Naturally, it is a *flat* measure which does not consider the class hierarchy. However, it can be easily derived into a hierarchical metric by computing the accuracy at each level of the class hierarchy when it is available.

Hierarchical distance of mistake

As opposed to the previous metric which measures the correctness of the classification algorithm, the hierarchical distance of mistake (HDM) (Bertinetto et al., 2020) rather assesses the severity of the error. This metric is defined as the distance $d_{\mathcal{H}}$ between the true class y and the predicted class $\hat{y}$ in the class hierarchy $\mathcal{H}$, which is represented by the height of their Lowest Common Ancestor (LCA). However, we only consider misclassified entries $\mathcal{N} = \{(y, \hat{y}), y \neq \hat{y}\}$, i.e. when the predicted class $\hat{y}$ is different from the ground truth class y. The smaller the value of this measure the better. Formally, it is defined as:

$$\text{HDM} = \frac{1}{|\mathcal{N}|} \sum_{(y,\hat{y}) \in \mathcal{N}} d_{\mathcal{H}}(y, \hat{y}) \tag{2.22}$$

Hierarchical precision

Hierarchical precision (P_H) (Liu et al., 2021) is an augmented version of the precision which uses the class hierarchy. Unlike pair-based measures such as accuracy, this measure operates on the full sets of predicted and true classes, including their ancestors. It is computed as:

$$P_H = \frac{\left|\hat{Y}_{aug} \cap Y_{aug}\right|}{\left|\hat{Y}_{aug}\right|} \tag{2.23}$$

$|.|$ is the cardinality of a set. $\hat{Y}_{aug}$ and Y_{aug} are the augmented sets of $\hat{y}$ and y which refer to the predicted and ground truth classes, respectively, that may be expanded by including their ancestors in the class hierarchy $\mathcal{H}$. In other words, they represent the sets of classes along the path from the predicted class $\hat{y}$, resp. ground truth class y, to the root class in the class hierarchy $\mathcal{H}$ and are defined as:

$$Y_{aug} = \text{Ancestors}_{\mathcal{H}}(y) \cup \{y\}$$

$$\hat{Y}_{aug} = \text{Ancestors}_{\mathcal{H}}(\hat{y}) \cup \{\hat{y}\}$$

$\text{Ancestors}_{\mathcal{H}}(y)$ returns, for y as a leaf, all non-leaf nodes in the class hierarchy $\mathcal{H}$ that lie along the path from y to the root node, the root node itself is excluded from this set. The higher P_H, the better.

Hierarchical F-score

The hierarchical F-score (Liu et al., 2021) is an adaptation of the F-Score, which is based on the hierarchical recall R_H and hierarchical precision P_H, as previously defined. It can be expressed as:

$$F_H = \frac{2 \times P_H \times R_H}{P_H + R_H} \tag{2.24}$$

with R_H defined as:

$$R_H = \frac{\left|\hat{Y}_{aug} \cap Y_{aug}\right|}{|Y_{aug}|} \tag{2.25}$$

Lowest common ancestor

Lowest Common Ancestor (LCA) (Liu et al., 2021) is a hierarchical evaluation metric derived from F_H. The main change is in the calculation of the augmented sets. Rather than having classes from the leaf node (predicted class or ground truth class) up to the root node, we consider only classes up to the closest ancestor class found between the true class and the predicted class in the class hierarchy $\mathcal{H}$. The higher the LCA value, the better.

Tree-induced error

The Tree-Induced Error (TIE) (Liu et al., 2021) is a metric which measures the severity of a misclassification, similar to HDM. However, TIE considers all cases, whether the predicted class is different from or the same as the ground truth class. This metric is defined by the distance between the true class y and the predicted class $\hat{y}$ in the class hierarchy $\mathcal{H}$; the distance is given by the number of edges connecting the true class to the predicted class in $\mathcal{H}$. The smaller the value of this measure the better. Formally, it is defined as:

$$\text{TIE} = |Edges_{\mathcal{H}}(y, \hat{y})| \tag{2.26}$$

2.5 Conclusion

This chapter has outlined some approaches which allow us exploiting hierarchical information when learning data embeddings for ML tasks such as classification and retrieval. The first category of approaches explicitly considers the class hierarchy, and can be related to three lines of research: defining a hierarchical loss function, altering the network architecture or the label-embedding space. To the best of our knowledge, most studies carried out in the remote sensing community for scene analysis fall into this first category. The second direction implicitly exploit hierarchical information by considering the hyperbolic space as an embeddings space. The hyperbolic space has attracted a lot of attention recently and has proven beneficial for data representation in various applications, especially when the data has a hidden or underlying hierarchical structure. In the two following chapters, we address both lines of research. In Chapter 3, we will discuss explicit methods for incorporating class hierarchy when analysing remote sensing scene images. In particular, we will consider the incorporation of class hierarchy via the loss function. In chapter 4, we investigate the relevance of hyperbolic space for remote sensing scene image analysis. In both cases, we will consider the two following learning frameworks: VAE-based embedding and few-shot learning.

Chapter 3

Leveraging class hierarchy via loss functions

The content of this chapter is mainly built upon the research presented in the two articles: "*Hyperbolic Variational Auto-Encoder for Remote Sensing Scene Classification*" (Hamzaoui et al., 2021) and "*A Hierarchical Prototypical Network for Few-Shot Remote Sensing Scene Classification*" (Hamzaoui et al., 2022).

3.1 Introduction

In computer vision, the issue of mistake severity has been discussed in several works (Bertinetto et al., 2020), especially after the release of the ImageNet dataset (Deng et al., 2009), whose classes are organised into a class hierarchy according to the WordNet ontology (Miller, 1998). Nevertheless, this issue has only recently regained attention after having been largely neglected since the advent of the deep learning. A recent and noteworthy study presented in (Bertinetto et al., 2020) incorporates the class hierarchy through two distinct approaches: a label-embedding-based approach and a loss-based one. The objective is to train a classifier that is capable of mitigating the severity of classification errors.

As discussed in Chapter 1, RS data are naturally hierarchical, it would therefore be interesting to consider this information when learning the scene classifier. Some recent studies, such as (Liu et al., 2020b; Sen & Keles, 2022b; Zeng et al., 2022), have benefited from this nature by explicitly introducing the class hierarchy into the learning process in order to enhance the performance and ensure more meaningful predictions even when they are incorrect. We share an interest in this research direction and intend in this chapter to explicitly incorporate the semantic information provided by the class hierarchy. Specifically,

we propose to introduce the class hierarchy via the loss function to learn a more meaningful scene classifier.

The first section of this chapter is an initial check which aims to demonstrate the potential benefits of explicitly incorporating hierarchical information in feature construction. As such, we customised the research presented in (Yu et al., 2020), which proposed a guided variational auto-encoder (VAE) to embed drugs whose labels were arranged in a class hierarchy, to suit our context. To be specific, we employ the VAE to embed scene images, while restricting its latent space using a soft local ranking loss, which is parameterised by the class hierarchy. This approach shows that adding the class hierarchy information can improve classification performance. Consequently, in the second section, we address another challenging task, namely the few-shot learning problem. Here we adopt a prototypical network as a framework to encode the scene images. The class hierarchy is then leveraged to derive prototypes at different levels of the class hierarchy, which will then be fed to a weighted sum of *cross-entropy* losses in order to optimise the network and guide feature learning.

3.2 Label-driven variational auto-encoder learning

In order to determine whether the incorporation of hierarchical information affects the quality of remote sensing representations, particularly the performance of RS scene image classifier, we conducted an initial investigation to validate its potential benefits.

In this section, we expand upon the methodology presented in (Yu et al., 2020), which employs a simple data embedding technique using VAE, to embed RS scene images. To incorporate the class hierarchy, they utilised the soft local ranking loss, which guides the construction of the VAE latent space. We evaluate the effectiveness of this framework and thus the benefits of incorporating the class hierarchy by analysing the quality of the embeddings through a classification task.

3.2.1 Variational auto-encoder

Variational Auto-Encoder (VAE) (Kingma & Welling, 2014) is a probabilistic generative model relevant to representation learning in which we aim to learn good representations, such as interpretable representations or representations that give a better generalisation (Mathieu et al., 2019). A VAE model is composed of two components: an encoder that

embeds observations x^i into a low dimensional latent space $z \in Z$, and a decoder generating observations $\hat{x}^i$ out of this latent space. Formally, the VAE consists of a probabilistic decoder defined as a likelihood function $p_\theta(x^i|z)$ and parameterised by θ which generates data $\hat{x}^i$ given the latent variable z as well as a posterior distribution $q_\phi(z|x^i)$ that can be considered as a probabilistic encoder parameterised by ϕ. The parameters ϕ and θ are learned simultaneously by maximising the evidence lower bound (ELBO) which is defined for each observation x^i by:

$$\log p_\theta(x^i) \geq \mathbb{E}_{z\sim q_\phi(z|x^i)}[\log p_\theta(x^i|z)] - D_{KL}(q_\phi(z|x^i)||p_\theta(z)) \tag{3.1}$$

where the first term after the inequality encourages the decoder to learn to reconstruct the observation x^i, and the second is a regularisation term that promotes latent space representations to follow a predefined distribution, $\mathbb{E}$ and D_{KL} being respectively the expectation and the Kullback–Leibler (KL) divergence. Usually, $p_\theta(z)$ is chosen as a standard Normal distribution with mean zero and variance one $\mathcal{N}(0, \mathrm{I})$ where I is the identity matrix.

In practice, we approximate the reconstruction term by sampling using a Monte Carlo estimator:

$$\mathbb{E}_{z\sim q_\phi(z|x^i)}[\log p_\theta(x^i|z)] \approx \frac{1}{L}\sum_{l=1}^{L} \log p_\theta(x^i|z^{(i,l)}) \tag{3.2}$$

where L is the number of samples per data point x^i, $z^{(i,l)} = g_\phi(\epsilon^{(i,l)}, x^i) = \mu_\phi^i + \sigma_\phi^i \odot \epsilon^{(i,l)}$ is the reparameterisation trick, $\odot$ indicates an element-wise product and $\epsilon^{(i,l)} \sim \mathcal{N}(0, I)$ is a random noise vector. μ_ϕ^i and σ_ϕ^i are outputs of the encoder, representing respectively the mean and the standard deviation of the target distribution.

The regularisation term D_{KL} (Odaibo, 2019) encourages the approximate posterior $q_\phi(z|x^i)$ to be close to the prior $p_\theta(z)$ and is defined as:

$$\begin{aligned} D_{KL}\left(q_\phi\left(z|x^i\right) \parallel p_\theta(z)\right) &= \mathbb{E}_{q_\phi}\left[\log \frac{q_\phi\left(z|x^i\right)}{p_\theta(z)}\right] \\ &= -\frac{1}{2}\sum_{j=1}^{d}\left[1 + \log\left({\sigma_{j,\phi}^i}^2\right) - {\sigma_{j,\phi}^i}^2 - {\mu_{j,\phi}^i}^2\right] \end{aligned} \tag{3.3}$$

where d is the dimension of z, $\mu_{j,\phi}^i$ and $\sigma_{j,\phi}^i$ denote the j^{th} element of the encoder output.

The objective function to be maximised during training is therefore given by:

$$\mathcal{L}(\theta, \phi; x^i) = \frac{1}{L} \sum_{l=1}^{L} \log p_\theta(x^i | z^{(i,l)}) + \frac{1}{2} \sum_{j=1}^{d} \left[1 + \log \left({\sigma^i_{j,\phi}}^2 \right) - {\sigma^i_{j,\phi}}^2 - {\mu^i_{j,\phi}}^2 \right] \tag{3.4}$$

and the loss function to be optimised is simply taken as the negative of $\mathcal{L}(\theta, \phi; x^i)$.

3.2.2 Label-driven VAE

VAE only considers visual information when learning image embeddings. Here, we detail the incorporation of the hierarchical class structure into the VAE learning process so as to supervise and guide the construction of the latent space Z (see Figure 3.1). To do this, and following (Yu et al., 2020), we use a class hierarchy-based pairwise similarity measurement between images, which aims to bring semantically similar images closer together and distancing them from those that are less similar.

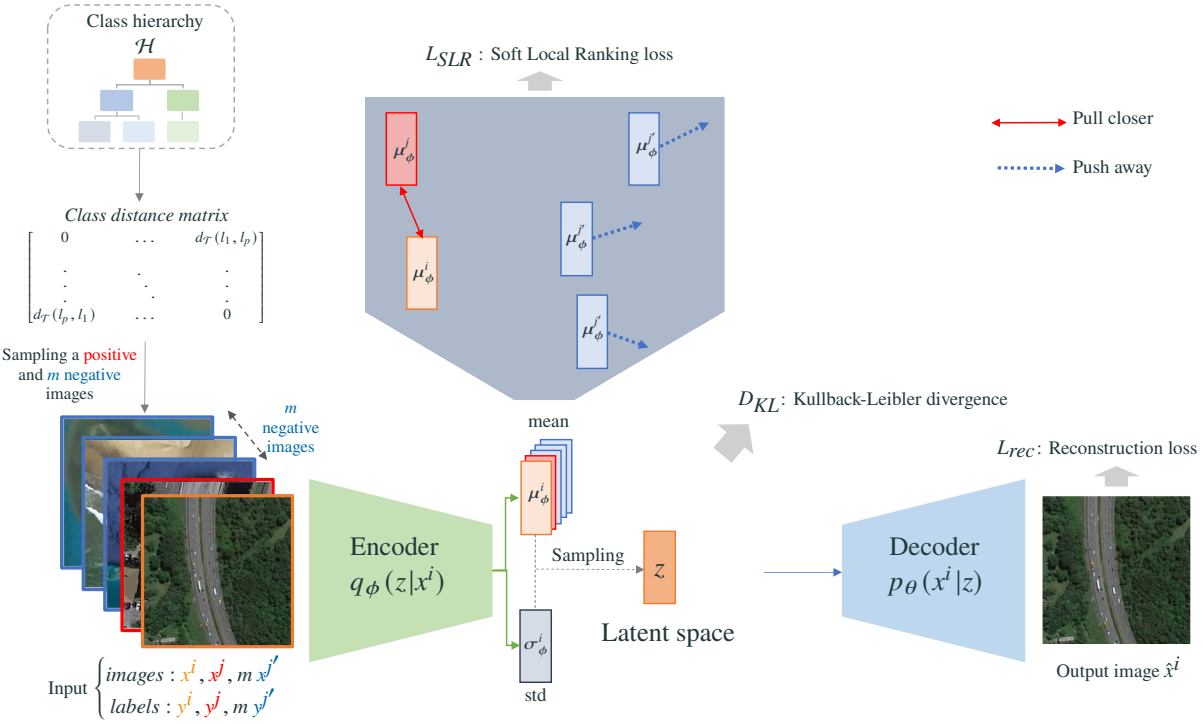


Figure 3.1: Overall framework of the proposed hierarchical VAE

We therefore drive the construction of our latent space Z by optimising the Soft Local Ranking (SLR) loss defined as :

$$\mathcal{L}_{\text{SLR}}(x^i, \mathcal{T}; \phi) = -\sum_{i,j} \log \Pr(x^i, x^j; \phi),$$

where

$$\Pr(x^i, x^j; \phi) = \frac{e^{-d(\mu_\phi^i, \mu_\phi^j)}}{\sum_{j' \in \mathcal{N}(i,j)} e^{-d(\mu_\phi^i, \mu_\phi^{j'})}}, \tag{3.5}$$

where μ_ϕ^i is the mean of the input image x^i, $d(\mu_\phi^i, \mu_\phi^j)$ is the Euclidean distance between μ_ϕ^i and μ_ϕ^j.

$\mathcal{N}(i,j)$ is the set referencing images semantically less similar to x^i than x^j including x^j, which is given by $\mathcal{N}(i,j) = \left\{j' : d_{\mathcal{T}}\left(y^i, y^{j'}\right) > d_{\mathcal{T}}\left(y^i, y^j\right)\right\} \cup \{j\}$ where $d_{\mathcal{T}}\left(^i, y^j\right)$ is the path-length between y^i and y^j, labels of images x^i and x^j respectively, in the class hierarchy $\mathcal{H}$.

We then formulate our label-driven VAE for scene image embedding as:

$$\arg\max_{\phi,\theta} \left((1-\gamma)\mathcal{L}_{\text{ELBO}}\left(x, \phi; \theta\right) - \gamma\ \mathcal{L}_{\text{SLR}}\left(x; \mathcal{T}, \phi\right)\right), \tag{3.6}$$

The first term is the VAE objective which embeds the scene images based on their visual similarity, while the second term is the SLR objective detailed above.

The label-driven VAE objective can thus be detailed as:

$$\arg\max_{\phi,\theta} \left((1-\gamma)\left(\mathbb{E}_{z\sim q_\phi(z|x^i)}\left[\log p_\theta(x^i|z)\right] - \beta\ D_{KL}\left(q_\phi(z|x^{(i)})||p_\theta(z)\right)\right) - \gamma\ \mathcal{L}_{\text{SLR}}\left(x, \mathcal{T}; \phi\right)\right) \tag{3.7}$$

where x^i are scene images, ϕ and θ are VAE parameters, β and γ are the scaling hyper-parameters controlling the weight relative to the KL divergence and SLR during training.

3.2.3 Experimental study

This study focuses on the relevance of explicitly considering, via a loss function, the hierarchical information among classes in the context of remote sensing, rather than competing with the most recent scene embedding approaches. The objective is therefore to investigate whether the guided VAE, referred to as VAE+SLR, outperforms the baseline *flat* VAE as well as a simple CNN classifier trained with a few labelled data.

In this perspective, for both the VAE+SLR and the VAE, we adopt a simple VAE

architecture with regard to those used recently in the remote sensing community (Cheng et al., 2017; Cheng et al., 2020; Dutta & Das, 2023). As for the classifier, for a fair evaluation, we also adopt a simple architecture by using the same encoder of VAE as the feature extractor, followed by two fully connected layers for classification.

We evaluate the quality of the resulting VAE embeddings and the ability to discriminate classes using the simple 1−NN classifier.

Experimental setup

Dataset The experiments are conducted on a subset of the NWPU-RESISC45 (Cheng et al., 2017) remote sensing scene dataset (Section 1.2). All 45 classes are considered, for each, we randomly select 100 images for the training set, 50 images for the validation set and 80 images for the test set. Within the training set, we consider only 10% of the images in each class as labelled .

Implementation details For both the VAE and the VAE+SLR, we choose the same following architecture. Both the encoder and the decoder are composed of 5 convolutional layers and a linear layer, each convolutional layer is followed by a batch normalisation layer and a Leaky ReLU activation, except for the decoder last convolutional layer which is followed by a tanh activation. The input size of the encoder network is set to 64×64. As the architecture we are using is not commonly applied in RS, we do not have prior knowledge of the optimal latent space dimension d for the embedding z. Therefore, in our experiments, we explore different values of d, including 8, 16, 32, 64, and 128.

The Adam optimiser (Kingma & Ba, 2015) acts as optimiser with a constant learning rate of $1e^{-3}$. The models are trained with mini-batches of size 64 for 1500 epochs with an early stopping of 50 epochs and 10 negative samples to optimise the SLR term.

The ELBO term is approximated by Monte Carlo (MC) estimation with $L = 1$ (Eq. (3.2)). β scaling hyper-parameter weight of the KL divergence was chosen experimentally and set to $5e^{-5}$ while γ, the scaling hyper-parameter weight of the SLR loss, is set to 0.1.

As for the classifier, we adapt the dimension of the first fully connected layer to match the dimension of the VAE latent space, while the dimension of the second fully connected layer corresponds to the number of classes in the dataset, which is 45.

Results and discussion

We evaluate the quality of the resulting embeddings of both VAEs and the ability to discriminate between classes using the simple 1−NN classifier. We also provide the classification results of a simple CNN classifier trained only on the labelled images available for the Label-driven VAE (VAE+SLR). Experiments are conducted on a subset of the NWPU-RESISC45 dataset and reported in Table 3.1, results are averaged over 3 runs.

Table 3.1: 1-NN classification results computed on the test set of the NWPU-RESISC45 dataset at different levels of the class hierarchy: overall acc represents the classification accuracy at the leaves (level 4) and thus the NWPU-RESISC45 classes; L3-acc and L2-acc give the accuracy at level 3 and level 2, respectively (the higher, the better); HDM is the hierarchical distance of mistakes (the smaller the better). Results are averaged over 3 runs.

	Metric	Latent Space Dimension d				
		8	16	32	64	128
CNN classifier	Overall acc	8.11 ± 0.10	10.58 ± 0.07	13.72 ± 0.52	14.49 ± 0.40	15.76 ± 0.40
	L3-acc	14.05 ± 0.84	16.55 ± 0.77	19.89 ± 0.90	21.47 ± 0.53	23.58 ± 1.00
	L2-acc	24.27 ± 1.24	27.14 ± 1.05	29.52 ± 0.66	32.71 ± 0.37	33.76 ± 1.44
	HDM	2.71 ± 0.01	2.71 ± 0.01	2.71 ± 0.01	2.70 ± 0.01	2.71 ± 0.01
VAE	Overall acc	12.00 ± 0.15	13.96 ± 0.56	13.21 ± 0.21	12.08 ± 0.10	12.39 ± 0.32
	L3-acc	$18.38 \pm 0,42$	20.64 ± 0.41	19.33 ± 0.13	17.93 ± 0.57	17.88 ± 0.24
	L2-acc	$28.34 \pm 0,57$	31.49 ± 0.38	30.97 ± 0.16	29.96 ± 0.47	29.95 ± 0.49
	HDM	2.74 ± 0.01	2.72 ± 0.01	2.72 ± 0.01	2.73 ± 0.01	2.74 ± 0.00
VAE+SLR	Overall acc	$\mathbf{13.24 \pm 0.49}$	$\mathbf{14.04 \pm 0.21}$	$\mathbf{15.97 \pm 0.41}$	$\mathbf{15.58 \pm 0.38}$	$\mathbf{16.97 \pm 0.63}$
	L3-acc	$\mathbf{20.68 \pm 0.74}$	$\mathbf{21.53 \pm 0.46}$	$\mathbf{23.48 \pm 0.46}$	$\mathbf{22.83 \pm 0.52}$	$\mathbf{24.16 \pm 0.40}$
	L2-acc	$\mathbf{31.64 \pm 0.60}$	$\mathbf{32.07 \pm 0.65}$	$\mathbf{33.80 \pm 0.44}$	$\mathbf{33.20 \pm 0.40}$	$\mathbf{35.09 \pm 0.25}$
	HDM	$\mathbf{2.70 \pm 0.01}$	$\mathbf{2.70 \pm 0.01}$	$\mathbf{2.70 \pm 0.01}$	2.71 ± 0.02	$\mathbf{2.70 \pm 0.01}$

When comparing VAE+SLR against the *flat* VAE and the *flat* classifier, we observe that VAE+SLR outperforms both models in terms of classification accuracy across different levels and dimensions. Although our VAE+SLR model's performance falls significantly short of the state-of-the-art (Miao et al., 2022), we anticipated it would outperform the two other models, owing to its ability to utilise both labelled and unlabelled images. The unsupervised component of the VAE allowed for learning the underlying data distribution from the unlabelled images. Meanwhile, the labelled images were utilised to improve the organisation of the latent space while ensuring that the class hierarchy constraints were met. Our SLR term is specifically designed to encourage a more structured and informative latent space that aligns with prior knowledge about the data, i.e., the class hierarchy. This, in turn, led to improved regularisation of the latent space and enhanced classification performance.

We evaluate the performance of various approaches over different dimensions. For the CNN classifier, we modified the dimension of the layer preceding the softmax layer to

ensure a fair comparison. By using cross-validation, we determined that the baseline VAE attains its peak performance on the validation set when the optimal latent dimension is set to 16. This dimension was also optimal on the test set, although other dimensions showed relatively similar performances. In contrast, the performance of the CNN classifier is strongly impacted by the dimension, with a overall accuracy gain of nearly twice observed in dimension 128 as compared to dimension 8. Our guided VAE, on the other hand, leveraged the strengths of both the VAE and the classifier, demonstrating promising performance at small latent dimensions derived from the VAE. Additionally, the labelled data further improved the model's performance at higher dimensions.

Regarding the misclassified images which are organised according to the class hierarchy, we observe that VAE+SLR slightly improves the misclassification of the VAE. This implies that incorporating the class hierarchy to guide the construction of the VAE latent space has enabled a more effective reorganisation, thereby reducing the severity of mistakes. However, there is still a considerable margin for improvement and the first alternative we can think of is to use feature extractors which are already widely used in the RS community.

3.2.4 Conclusion

In this section, we conducted a preliminary study on the potential benefits of introducing the class hierarchy in the learning. In particular, we examined a label-driven VAE, in which the latent space construction is guided by a pairwise loss function that exploits the class hierarchy to select the relevant pairs. We evaluate the hierarchical VAE on the NWPU-RESISC45 dataset and showed that the class hierarchy is an appealing source of information that allows improving the classification performance.

Having established that incorporating the class hierarchy is advantageous in the remote sensing context, we can now proceed to a more challenging task, namely the few-shot RSISC. Here, we incorporate the class hierarchy in order to transfer semantic information from the source (seen) classes to the target (unseen) classes.

3.3 Hierarchical prototypical network for few-shot classification

Inspired by the human ability to learn new abstract concepts from very few (or even one) examples and to generalise quickly to new instances (Shi et al., 2020), Few-shot learning (FSL) was introduced as one of the alternative ways to deal with the "data-hungry" issue. FSL methods can be divided into three categories (Sun et al., 2021): metric learning, meta-learning and transfer learning. Metric learning methods learn a distance function that brings samples from the same category as close as possible in the feature space while pushing samples from other categories as far away as possible. As for meta-learning, also known as learning to learn, it is the most common approach in FSL, which efficiently optimises the model parameters to new tasks. Transfer learning aims at using the knowledge gained from relevant tasks towards new tasks, *e.g.* fine-tuning the pre-trained models is a powerful transfer method.

Recently, the combination of meta-learning and metric learning has been one of the most studied approaches in FSL for natural image classification (Snell et al., 2017; Vinyals et al., 2016) and for remote sensing scene classification (Zhang et al., 2021a). First, based on meta-learning, these approaches construct tasks with few labelled samples, which enhances the generalisation performance of the model for new tasks. Then, the similarity between image features is measured to make predictions. Some of related methods include relation network (Sung et al., 2018), classical matching network (Vinyals et al., 2016) and prototypical network (Snell et al., 2017).

In recent years, several approaches were proposed to tackle the problem of few-shot remote sensing scene classification (FSRSSC). In (Li et al., 2021a), the authors adopted the attention mechanism to delve into the inter-channel and inter-spatial relationships to discover discriminative regions in the remote sensing scene images. The authors in (Cheng et al., 2022) used a Siamese-prototype network with prototype self-calibration and inter-calibration to learn more discriminative prototypes. In (Zhang et al., 2021a), the authors introduced a pre-training step on the base data to provide better initialisation of the feature extractor and performed the few-shot remote sensing scene classification using cosine distance metric. However, to the best our knowledge, the majority of these methods have focused only on visual scene information to improve feature representations without considering semantic knowledge that may exist within these classes. Yet this type of semantic knowledge about classes, which can consist of attributes, word embeddings or

even a knowledge graph (e.g. WordNet (Miller, 1998)), is commonly used in zero-shot learning (ZSL) and increasingly in few-shot natural image classification approaches.

Although incorporating semantic knowledge is not a novelty in ZSL, it has only recently been applied in FSL. (Chen et al., 2019) proposed the TriNet to tackle the "1-shot" task by synthesising the instance features from the semantic space which is given by the label embeddings. In (Yang et al., 2022a), the authors proposed a method called Semantic Guided Attention (SEGA) mechanism which leverages semantic knowledge to guide the visual perception in learning the discriminative visual features of each class. Most of these FSL approaches that introduce semantic knowledge involve the text modality. However, few attention has been paid to knowledge transfer based on the class hierarchy which is either built using text modality as in (Li et al., 2019) or already predefined as in (Liu et al., 2022). In (Li et al., 2019), the authors proposed a hierarchical image recognition approach by performing Softmax optimisation on all levels of the class hierarchy. This allows learning transferable visual features through this class hierarchy which encodes semantic relationships between seen and unseen classes. In (Liu et al., 2022), a class hierarchy was introduced to address the multi-class FSL problem. The authors proposed a "memory-augmented hierarchical-classification network (MahiNet)" model which leverages the hierarchy as prior knowledge to train a coarse-to-fine classifier where each coarse class can cover multiple finer classes.

According to (Liu et al., 2022), FSL with knowledge transfer can be accomplished independently of an additional modality such as text and yields competitive performances, when the class hierarchy is known or easily obtained, which fits well with our research interests. As we mentioned in Chapter 1, the remote sensing classes can be easily arranged in a hierarchical structure following well-known organisations such as Corine Land Cover (CLC), the European Nature Information System (EUNIS) habitat classification scheme or other structures such as done in (Liu et al., 2020b) where they propose a hierarchical organisation of the scene classes of the PatternNet (Zhou et al., 2018) remote sensing scene dataset.

In this section, we build on prototypical networks to define a hierarchical variant: in a nutshell, hierarchical prototypes are attached to each level of the hierarchy, allowing us to first consider high-level aggregated information before making a fine prediction.

3.3.1 Problem formulation

In few-shot classification, we assume that we have two sets, a large labelled training set, referred to as the base set D_{base}, and a test set with few labelled images per class, the novel set D_{novel}. The classes that constitute the base and novel sets, denoted C_{base} and C_{novel} respectively, are disjoint $C_{base} \cap C_{novel} = \emptyset$. To mimic the sparsity of the test data in the training stage, we adopt the $K-$way $N-$shot strategy (an episodic learning strategy) used in various FSL studies (Snell et al., 2017; Vinyals et al., 2016), in which K refers to the number of classes and N (usually set to 1 or 5) is the number of labelled images per classes during a training/testing episode. For each training episode, we randomly sample a subset of K classes out of C_{base} which we denote C_e. We then randomly sample N labelled images from D_{base} for each class $k \in C_e$, resulting in the episode support set $S = \{(x_i, y_i)\}_{i=1}^{K \times N}$, where x_i is an image and $y_i \in C_e$ its corresponding label. Furthermore, for the same K selected classes, we sample N' labelled images for each class $k \in C_e$ to form a set known as the query set $Q = \{(x_i, y_i)\}_{i=1}^{K \times N'}$. A training episode therefore has a total of $K \times (N + N')$ samples. In this training step, the support set S and the query set Q are used to learn the model that projects the input images into the feature space.

The testing step is also carried out with the same episodic strategy where we have an unlabelled query set Q (drawn from D_{novel}) for which we want to predict the class label of each query sample $x_i \in Q$ using the labelled support set S (also drawn from D_{novel}). Fig. 3.2 shows a visualisation of the K-way N-shot episodes.

3.3.2 Prototypical networks

Prototypical networks (Snell et al., 2017) are metric learning-based methods which learn a distance function in order to bring samples within the same category as close as possible in the feature space, while pushing away samples from other categories. They adopt an episodic strategy to train the meta-learner classifier. Given an episode with a support set S and a query set Q, we compute the representations of the images in both sets S and Q using the meta-learner feature extractor f_Φ (a neural network such as CNN) parameterised by Φ. Thereafter, the support set representations are averaged to compute the prototypes p^k for each class $k \in C_e$ as follows:

$$p^k = \frac{1}{N} \sum_{(x_i, y_i) \in S^k} f_\Phi(x_i) \tag{3.8}$$

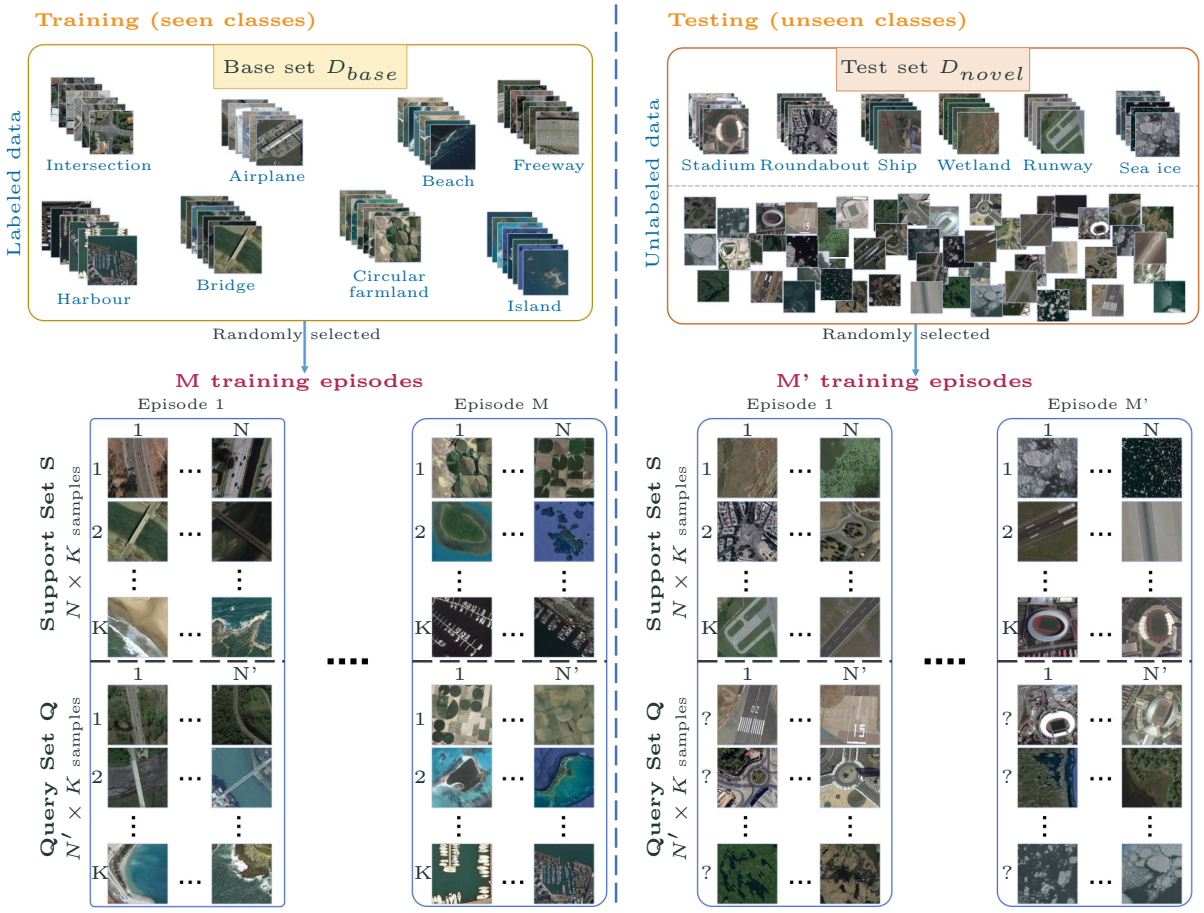


Figure 3.2: Illustration of K-way N-shot classification episodes. The left side shows the M episodes of the training step; each episode consists of $K \times N$ support samples and $K \times N'$ query samples. The testing step is similarly defined on M' episodes, as shown on the right.

where S^k is the subset of the episode support set S that contains the samples of class $k \in C_e$, C_e is the set of classes sampled during episode e.

To optimise the feature extractor f_Φ, we minimise the loss function:

$$\mathcal{L} = -\frac{1}{K \times N'} \sum_{k \in C_e} \sum_{(x_i, y_i) \in Q^k} \log p_\Phi(y_i = k \mid x_i) \tag{3.9}$$

where Q^k is the subset of the episode query set Q composed of samples from class k and $p_\Phi(y_i = k \mid x_i)$ is the probability of predicting a query sample $(x_i, y_i) \in Q$ as class k and is given as:

$$p_\Phi(y_i = k | x_i) = \frac{\exp(-d(f_\Phi(x_i), p^k)/\tau)}{\sum_{k' \in C_e} \exp(-d(f_\Phi(x_i), p^{k'})/\tau)} \tag{3.10}$$

where $d(.)$ is the Euclidean distance (Snell et al., 2017) and τ is the temperature hyper-parameter.

3.3.3 Leveraging the class hierarchy in prototypical network learning

Overall framework

We propose a meta-learning framework whose complete pipeline is illustrated in Fig. 3.3 to solve the few-shot classification problem when a hierarchy that describes the organisation between the classes is available. We train a meta-learner classifier by adopting an episodic training strategy. During training stage, using the support set S, we compute N prototypes $\mathcal{P} = \left\{p^k\right\}_{k \in C_e}$ for each class in the current task (episode) and K_h hierarchical prototypes for their super-classes. The query features are then compared to both the scene and the hierarchical prototypes, allowing us to compute an episodic error at different levels of the class hierarchy $\mathcal{T}$ to be minimised and used to fine-tune the parameters Φ of the feature extractor f_Φ. At testing stage, the parameters Φ of the feature extractor f_Φ are fixed and the meta-learner classifier is evaluated on a set of episodes sampled from the novel classes in D_{novel}.

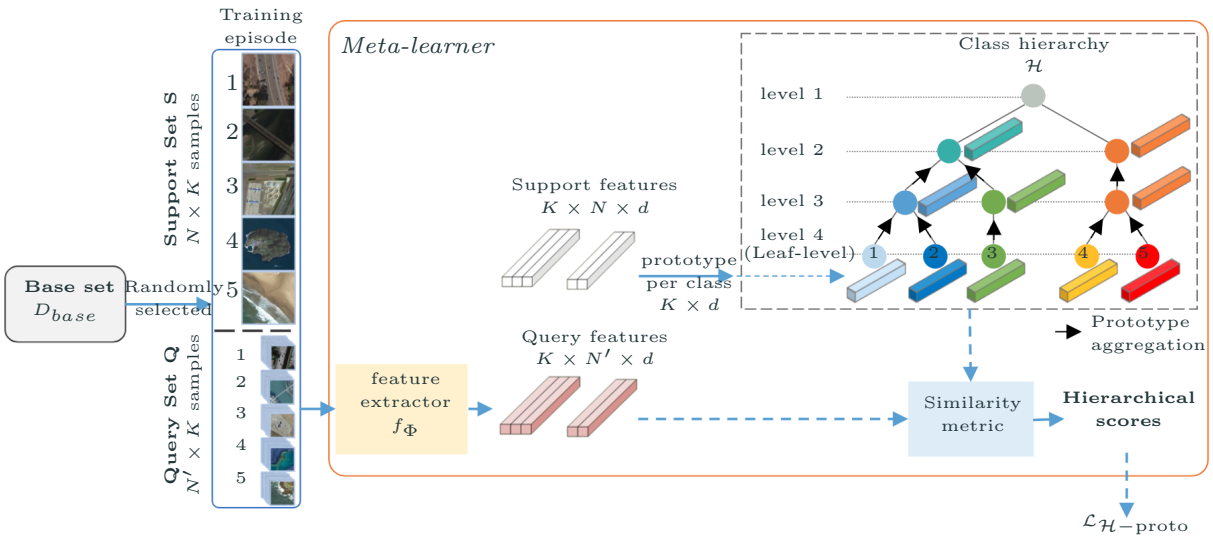


Figure 3.3: Overall framework of the proposed hierarchical prototypical network for few-shot image classification. In this example (one-shot), $K = 5$, $K_h = 5$, $N = 1$, $N' > 1$ (usually set to 15).

Hierarchical prototypical network

Here, we rely on the prototypical networks and introduce the hierarchy knowledge thanks to the definition of hierarchical prototypes. The overall idea is to regularise the latent space by putting closer classes that are in the same branch of the class hierarchy $\mathcal{T}$, and pushing apart classes that have common ancestors in higher levels of the class hierarchy $\mathcal{T}$.

To properly formulate our approach, given an episode, we first compute the prototypes per class which are prototypes at the leaf-level of the class hierarchy $\mathcal{T}$ (following Eq. 3.8). We then compute the hierarchical prototypes by aggregating the leaf-level prototypes according to $\mathcal{T}$.

The prototypes of the super-classes $k \in C_e^l$ (the hierarchical prototypes) at level ($1 < l < L$ with $l = 1$ the root node and $L = \text{height}(\mathcal{T})$) are denoted as $\mathcal{P}_l = \left\{p_l^k\right\}_{k \in C_e^l}$ and computed as the mean of support samples of the super-class sub-tree S_l^k similarly to Eq. 3.8:

$$p_l^k = \frac{1}{|S_l^k|} \sum_{(x_i, y_i) \in S_l^k} f_\Phi(x_i) \tag{3.11}$$

Note that when $l = L$, the prototypes at level l are the prototypes at the lowest level of $\mathcal{T}$ (leaf-level prototypes).

The hierarchical prototypical network outputs a distribution over classes for each query sample $x_q \in Q$ at different levels of $\mathcal{T}$, based on a Softmax over the distances to the prototypes of each level l in $\mathcal{T}$. We then formulate the probability of predicting the query features $f_\Phi(x_q)$ and the prototype p_l^k of its super-class k at level l in $\mathcal{T}$ as formulated in Eq. 3.10 as:

$$p_\Phi(y_i^l = k | x_i) = \frac{\exp(-d(f_\Phi(x_i), p_l^k)/\tau)}{\sum_{k' \in C_e^l} \exp(-d(f_\Phi(x_i), p_l^{k'})/\tau)} \tag{3.12}$$

where y_i^l is the ancestor of y_i at level l, C_e^l represents the super-classes at level l at the current episode.

We therefore optimise a new loss function given as

$$\mathcal{L}_{\mathcal{H}-\text{proto}} = \sum_{l=2}^{L} \lambda_l \mathcal{L}_l \tag{3.13}$$

where $\lambda_l = \frac{\gamma^{l-1}}{\sum_{l'=2}^{L} \gamma^{l'-1}}$, γ is a hyper-parameter that controls the importance of each level in the hierarchy and $\sum_{l=2}^{L} \lambda_l = 1$. $\mathcal{L}_l$ represents the prototypical network loss at level l of the

Table 3.2: NWPU-RESISC45 FSL splits

Split	Categories
Meta-train	chaparral, bridge, commercial area, golf course, dense residential, meadow, forest, airport, freeway, church, harbor, baseball diamond, circular farmland, medium residential, mobile home park, desert, basketball court, lake, beach, cloud, island, airplane, ground track field, industrial area, intersection.
Meta-validation	roundabout, wetland, ship, terrace, storage tank, sparse residential, tennis court, thermal power station, stadium, snowberg, sea ice, runway.
Meta-test	overpass, railway station, mountain, river, parking lot, palace, railway, rectangular farmland.

class hierarchy $\mathcal{T}$.

As such, we can tune the importance of each level of the hierarchy into the learning process: by choosing low values of γ, we put more importance into organising the higher levels of the hierarchy; a value close to one gives the same importance for all the levels; a high value tends to behave like the *flat* cross entropy loss formulation.

3.3.4 Experimental study

Experimental setup

Dataset All models are learned on the NWPU-RESISC45 (Cheng et al., 2017) remote sensing scene dataset. We split the dataset into three disjoint subsets: meta-training D_{base}, meta-validation D_{val}, and meta-test D_{novel} containing 25, 12, and 8 categories, respectively (Table 3.2). We note that the meta-validation set is used for hyper-parameter selection in the meta-training step. The meta-training set is further divided into three subsets: training, validation, and test sets. In our experiments, we follow (Zhang et al., 2021a) and resize all the images to 80×80 pixels to fit our designed feature extractor.

Implementation details

Following recent FSRSSC studies (Li et al., 2021c; Zhang et al., 2021a; Zhang et al., 2021b; Zhang et al., 2021c), we utilise ResNet-12 as a backbone for feature extraction. We also adopt the pre-training strategy as suggested in (Zhang et al., 2021a) to better initialise the meta-learner feature extractor.

We train our meta-learning for 400 epochs, with the best model parameters chosen based on the best overall accuracy on the validation set. In standard deep learning, an epoch implies that the entire train set passes through the deep neural network once. However, in meta-learning, an epoch is a set of episodes randomly sampled from the base set D_{base}, which we set to 500 episodes per epoch. We optimise the model based on the average loss of 2 episodes, i.e. the batch size is set to 2 episodes. We use SGD optimiser to update the network parameters with a momentum set to 0.9 and a weight decay set to 0.0005. The learning rate is fixed to 10^{-3}. After each training epoch, we test our model on a validation set D_{val} by randomly sampling 500 episodes, the network weights with the highest validation overall accuracy are retained as the best results. For the hierarchical hyper-parameter γ, we assigned different values ($\gamma = 1$, $\gamma < 1$ and $\gamma > 1$) in order to observe its impact on the framework performances.

For the meta-testing stage, we conduct a $5-$way $1-$shot and $5-$way $5-$shot classification following the widely used meta-learning protocol. We evaluate the best model on 1000 randomly sampled episodes from the test set D_{novel}. Following the FSL evaluation protocol (Snell et al., 2017), for K-way N-shot episode, we randomly sample 15 images per class to form the query set Q, making a total of $K \times 15$ query images per episode.

As for the meta-training episodes, if at test time a K-way classification and N-shot learning is expected, the training episodes could be composed of K ways and N support samples per class. However, (Snell et al., 2017) observed that it can be highly advantageous to train with a larger number of ways than will be used at test time, while maintaining the same N-shot for training and testing. Furthermore, within our framework, opting for a larger number of ways during training promotes the aggregation of prototypes into hierarchical prototypes and thus boosts the transfer of semantic information between classes via these prototypes. In the default setting (same K-way in training as in the test, $K = 5$), given the distribution of the train classes across the class hierarchy, having only 5 classes for training decreases the chances that two neighbouring classes in the hierarchy occur most often in the same episode. Therefore, considering a higher K-way for training seems to be an appropriate solution. We thus set K to 10 for the meta-training episodes.

Baselines

In both 5-way K-shot configurations, $N = 1$ or 5, We evaluate our approach against the following methods:

ProtoNet ProtoNet refers to the original *flat* prototypical network (Snell et al., 2017) which use the Euclidean distance as a similarity function.

Soft-labels The *soft-label* method (Bertinetto et al., 2020) was introduced to learn a deep classifier in a standard supervised setting while incorporating hierarchical information about the classes. It encodes information about the relationships between classes through a mapping function $y^{soft}(C)$ resulting in a categorical distribution over the classes, which is defined as follows:

$$y_A^{soft}(C) = \frac{\exp\left(-\beta\ d(A,C)\right)}{\sum_{B\in\mathcal{C}} \exp\left(-\beta\ d(B,C)\right)} \tag{3.14}$$

where d is the class distance function defined over the class hierarchy as the height of $\mathrm{LCA}(C_i, C_j)$ divided by the height of the class hierarchy and β is a hyper-parameter; $\beta \to \infty$ result in standard one-hot setting while $\beta = 0$ gives the uniform distribution.

The soft classifier is then learned by optimising the following loss function:

$$\mathcal{L}_{soft}(x, C) = -\sum_{A\in\mathcal{C}} y_A^{soft}(C) \log p(y = A|x) \tag{3.15}$$

where $\mathcal{C}$ is the set of classes (leaf nodes in the class hierarchy), C is the target class for example x and $p(y = A|x)$ is the probability of predicting x as class A.

We adapt this approach to the few-shot context by deriving *soft-labels* based on a sub-hierarchy covering only the current episode's classes.

Results and discussion

Table 3.3 and table 3.4 report the classification performance of the different approaches in both 5-way N-shot configurations, N = 5 and N = 1 respectively. We re-implement the *flat* method (ProtoNet) according to (Zhang et al., 2021a).

Our proposed *h-ProtoNet* achieves the highest accuracy and outperforms both *flat* prototypes (ProtoNet) and the *soft-labels* hierarchical loss in the 5-shot setting. We obtain the best performance with $\gamma = 2$, that is to say when we put more weights on the prototypes that correspond to the lower level of the hierarchy (corresponding to the leaf nodes, $\gamma > 1$). In the 1-shot setting, we outperform the *flat* ProtoNet achieving better results with $\gamma = 0.5$ which corresponds to the higher level of the hierarchy (corresponding to nodes close to the root, $\gamma < 1$). We observe that that incorporating information related to the class hierarchy leads to improved performance in our approach, as well as with *soft*

Table 3.3: 5-shot classification results computed on the test set of the NWPU-RESISC45 dataset at different levels of the class hierarchy: overall acc represents the classification accuracy at the leaves (level 4) and thus the NWPU-RESISC45 classes; L3-acc and L2-acc give the accuracy at level 3 and level 2, respectively; P_H is the hierarchical precision. All accuracy results are averaged over 1000 test episodes and are reported with a 95% confidence interval.

Method	**#HP**	**overall acc**	**L3-acc**	**L2-acc**	P_H
ProtoNet (Snell et al., 2017)	1	77.84 ± 0.40	80.89 ± 0.37	85.55 ± 0.41	81.43 ± 0.35
Soft-labels (Bertinetto et al., 2020)	4	76.77 ± 0.41	79.93 ± 0.37	85.39 ± 0.42	80.70 ± 0.35
h-ProtoNet (ours)	0.5	77.75 ± 0.39	80.98 ± 0.35	85.53 ± 0.41	81.42 ± 0.34
h-ProtoNet (ours)	1	78.41 ± 0.40	81.60 ± 0.36	$\mathbf{85.99 \pm 0.40}$	82.00 ± 0.34
h-ProtoNet (ours)	2	$\mathbf{78.65 \pm 0.40}$	$\mathbf{81.72 \pm 0.36}$	$\mathbf{85.99 \pm 0.40}$	$\mathbf{82.12 \pm 0.34}$

Table 3.4: 1-shot classification results computed on the test set of the NWPU-RESISC45 dataset at different levels of the class hierarchy: overall acc represents the classification accuracy at the leaves (level 4) and thus the NWPU-RESISC45 classes; L3-acc and L2-acc give the accuracy at level 3 and level 2, respectively; P_H is the hierarchical precision. All accuracy results are averaged over 1000 test episodes and are reported with a 95% confidence interval.

Method	**#HP**	**overall acc**	**L3-acc**	**L2-acc**	P_H
ProtoNet (Snell et al., 2017)	1	58.90 ± 0.61	62.56 ± 0.63	72.24 ± 0.71	64.57 ± 0.57
Soft-labels (Bertinetto et al., 2020)	4	$\mathbf{61.69 \pm 0.62}$	$\mathbf{65.66 \pm 0.62}$	$\mathbf{74.88 \pm 0.70}$	$\mathbf{67.41 \pm 0.57}$
h-ProtoNet (ours)	0.5	60.72 ± 0.62	64.77 ± 0.63	74.21 ± 0.70	66.57 ± 0.57
h-ProtoNet (ours)	1	60.20 ± 0.62	64.00 ± 0.62	72.83 ± 0.73	65.68 ± 0.59
h-ProtoNet (ours)	2	60.08 ± 0.62	64.09 ± 0.62	73.58 ± 0.71	65.92 ± 0.58

labels. The performance of our hierarchical prototypes is comparable to that of *soft labels*, indicating that we are effectively taking hierarchical information into account. Further investigation is needed to understand why our performance is either superior or inferior to that of *soft labels*, which we plan to explore in future research.

We observe that our h-ProtoNet is more sensitive to the class hierarchy when few labelled data are available (the 1-shot setting), thus further enhancing the performance of the *flat* ProtoNet. In this case, prototypes at higher levels of the hierarchy allow for significant information transfer between leaf prototypes.

Note that these values of $\gamma = 2$ for 5-shot and $\gamma = 0.5$ for 1-shot would have been selected if we perform a cross-validation on the validation set. We argue that the improvement observed in the case of the hierarchical prototypes is due to an efficient regularisation of the latent space, with a loss that encourages leaves within the same branch of the level hierarchy to be closer. As such, the performances at level 2 and 3 are improved, but also the overall accuracy.

Impact of γ hyper-parameter The aim of this investigation is to analyse the impact of the γ hyper-parameter of h-ProtoNet on the regularisation of its latent space, which in turn affects the transfer of hierarchical information among the prototypes at the leaf level. To achieve this, a series of experiments were carried out using a 5-shot 5-way setup with different values of the hyper-parameter γ. Additionally, we set the dimension of the latent space to 2 in order to obtain a visual comprehension of how the regularisation of the latent space is affected by the γ hyper-parameter.

Table 3.5 reports the accuracy values of various configurations of h-ProtoNet at both the finest and coarsest levels of the class hierarchy. Additionally, Figure 3.4 illustrates the latent space of a training episode for various configurations, providing a visual representation of the effect of the γ hyper-parameter on the learned latent space.

We experimented with four γ values: 0.5, 1, 2 and ∞. By definition, when $\gamma = 1$, equal importance is assigned to prototypes at every level of the class hierarchy. This leads to a latent space that strives to maintain the membership constraint at each level of the class hierarchy with equal significance (Figure 3.4(b)). Values of $\gamma < 1$ tend to prioritise prototypes at the upper levels of the class hierarchy (i.e., class nodes closer to the root), thereby assigning greater importance to predictions at coarser levels while being relatively less concerned with predictions at finer levels. This can be observed in Figure 3.4(a), where classes such as *transportation* and *public services* serve as examples. Values of $\gamma > 1$, on the other hand, tend to give greater weight to the lower levels of the hierarchy (i.e., leaf nodes), which can result in a decreased adherence to hierarchical constraints (Figure 3.4(c)). This is confirmed by Table 3.5 where we can observe that increasing the value of the hyper-parameter γ results in a decrease of the accuracy at the coarse classes (L2-acc), eventually leading to a *flat* ProtoNet when the $\gamma = \infty$ (Figure 3.4(d)).

Table 3.5: 5-shot classification accuracy of the two-dimensional h-ProtoNet on the test set of the NWPU-RESISC45 dataset at both the finest and coarsest levels of the class hierarchy. All accuracy results are averaged over 1000 test episode and are reported with a 95% confidence interval.

γ	**0.5**	1	2	∞
Overall acc	$\mathbf{54.18 \pm 0.49}$	52.28 ± 0.50	51.80 ± 0.49	52.10 ± 0.49
L2-acc	$\mathbf{74.99 \pm 0.64}$	74.69 ± 0.63	71.47 ± 0.70	69.27 ± 0.70

Does visual-semantic proximity matter? Although the hierarchy we have defined over the NWPU-RESISC45 classes is not a visual hierarchy *per se*, it nevertheless reflects

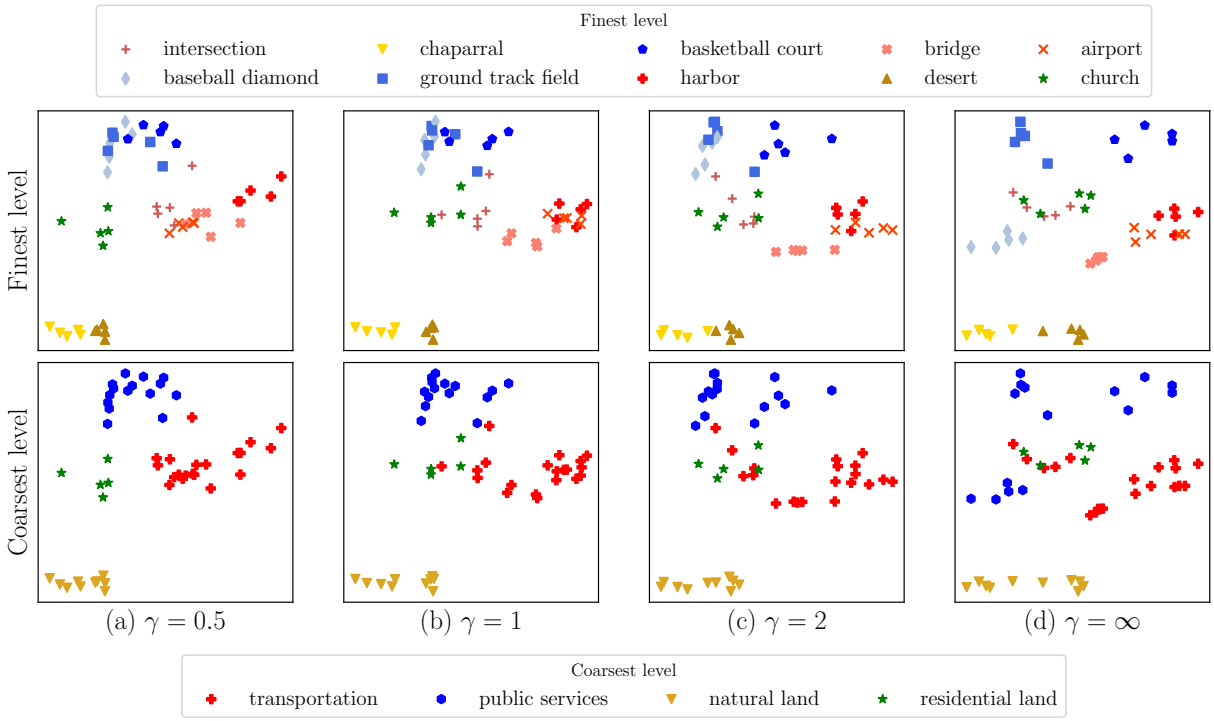


Figure 3.4: Two-dimensional embeddings of our h-ProtoNet at different γ values for the finest (top) and coarsest (bottom) levels of the class hierarchy.

some visual relationships between the dataset scene images. Figure 3.5 illustrates this feature by representing the latent space of both two-dimensional *flat* prototypical network (Figure 3.5(a) and (c)) and our hierarchical approach, $\gamma = 0.5$, (Figure 3.5(b) and (d)) of a training episode at the finest and coarsest levels of the class hierarchy. ProtoNet models, both *flat* and hierarchical, were trained using a 5-shot 5-way setting, with a two-dimensional latent space.

As we can observe, the *flat* ProtoNet has successfully managed at some point to organise its latent space by only considering the visual similarity between the scene images which also tend to reflect their semantic relationships, such as for the categories *medium residential* and *dense residential* which are clustered in the same area (see Figure 3.5(a)) resulting in the coarser class *residential land* (Figure 3.5(c)). However, it was difficult to distinguish other classes such as *intersection* and *church*, in which we can find visual similarities, although they are semantically distant according to the hierarchy we have defined. This issue is overcome by our h-ProtoNet (see Figure 3.5(b) and (d)).

As visual features are leveraged by deep networks, it is interesting to investigate the relevance of our approach with regard to the particularity of the class hierarchy. This leads

Figure 3.5: Two-dimensional embeddings of both the *flat* prototypical network (a and c) and our hierarchical approach, $\gamma = 0.5$, (b and d) at the finest (a and b) and coarsest (c and d) levels of the class hierarchy. The hierarchy below is the sub-hierarchy covering the finest training episode classes.

to answering the following question: *what happens if we impose an arbitrary hierarchy that can potentially alter the relationship between visual and semantic proximity?*

Therefore, based on the class hierarchy we defined, we swapped between: *lake* and

freeway, *airplane* and *forest*, *bridge* and *dense residential*, *mobile park home* and *circular farmland*, *basketball court* and *desert*, in order to build a new class hierarchy that violates the visual-semantic proximity relationship and repeat our experiment.

As illustrated in Figure 3.6, our h-ProtoNet (Figure 3.6(b) and (d)) has succeeded in reorganising its latent space following the new class hierarchy of classes for a given training episode. Nevertheless, it fails to generalise to new classes as shown in table 3.6.

Table 3.6: 5-shot classification accuracy of the two-dimensional h-ProtoNet on the test set of the NWPU-RESISC45 dataset at both the finest and coarsest levels of the new class hierarchy. All accuracy results are averaged over 1000 test episode and are reported with a 95% confidence interval.

γ	0.5	1	2	∞
Overall acc	45.01 ± 0.59	46.83 ± 0.59	44.83 ± 0.65	$\mathbf{52.10 \pm 0.49}$
L2-acc	52.51 ± 0.70	54.12 ± 0.68	50.25 ± 0.69	$\mathbf{59.96 \pm 0.65}$

Thus, the model performs better under the assumption that there are visual similarities between images of scenes within the same sub-hierarchy. However, if visual-independent semantic proximity relationships are desired, a more powerful semantic representation, such as text, may be required.

3.3.5 Conclusion

In this section, we have explicitly leveraged the hierarchical information about the classes to tackle a challenging issue, namely the few-shot RSISC.

We presented a novel prototypical network which defines hierarchical prototypes that match the nodes of the class hierarchy. We evaluated our method on the NWPU-RESISC45 RS scene dataset in an FSL context and showed that the hierarchical prototypes provide latent space regularisation, providing mostly better performance than the *flat* prototypes but also than a competitive hierarchical loss introduced in another context. Moreover, we assessed the impact of the γ hyper-parameter and visually illustrated the resulting behaviour in a two-dimensional space. We also examined the effect of a pre-defined class hierarchy, which revealed that the model's performance improves when images of scenes in the same sub-hierarchy are assumed to have visual similarities.

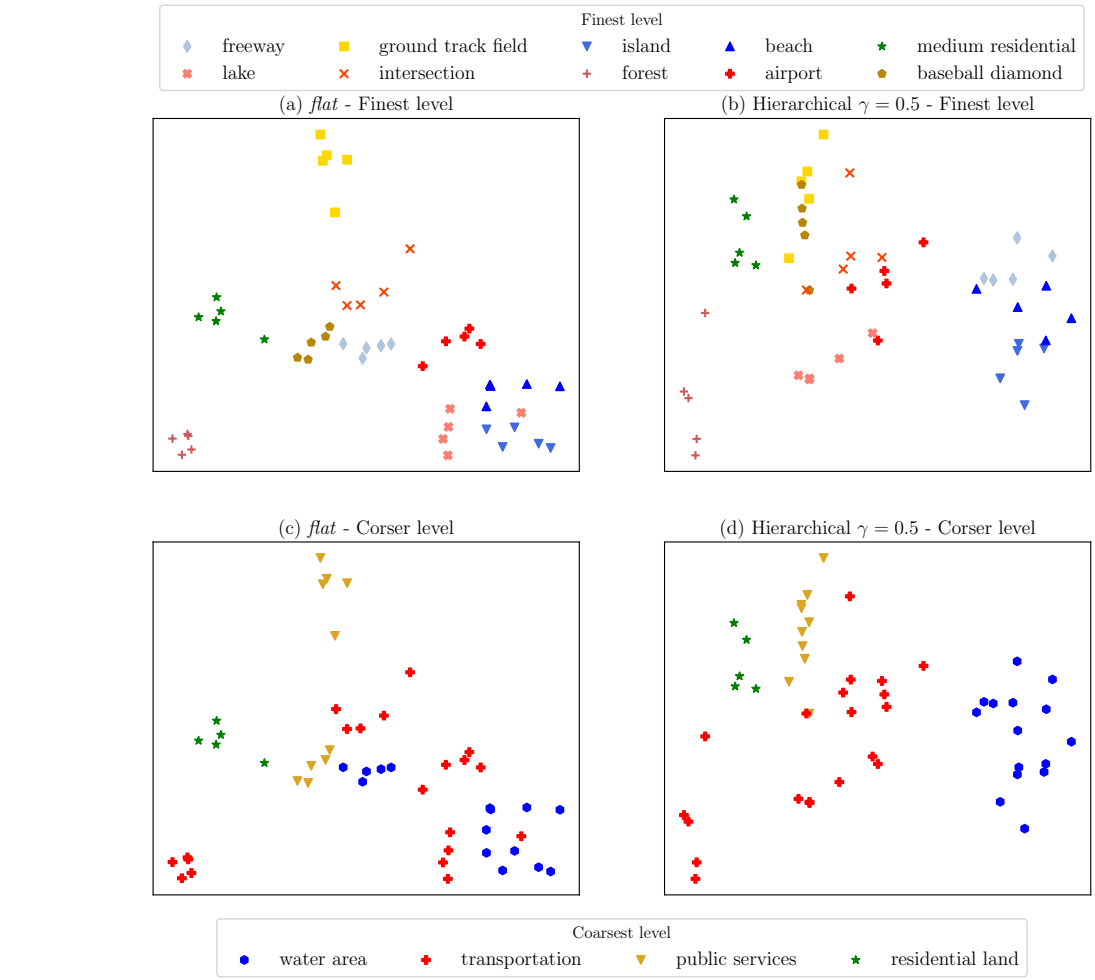


Figure 3.6: Two-dimensional embeddings of both the *flat* prototypical network (a and c) and our hierarchical approach, $\gamma = 0.5$, (b and d) at the finest (a and b) and coarsest (c and d) levels of the new class hierarchy. The hierarchy bellow is the sub-hierarchy covering the training episode classes.

3.4 Chapter summary

This chapter focused on investigating whether explicit incorporation of the semantic knowledge of RS scene classes, provided by the class hierarchy, can improve the performance of RSISC. In this context, we have considered two challenging settings.

The initial investigation aims to confirm the potential benefits of incorporating a class hierarchy within an RS context. In particular, we introduced the label-driven VAE which is an extension of classic VAE. Together with this architecture, we used the soft local ranking loss to drive the construction of the VAE latent space according to the class hierarchy. Our experiments, revealed that hierarchical information about classes can be an appealing source of information which can be used as a supplement to improve the performance of our framework. Nevertheless, it is important to note that this study was only a preliminary exploration of the advantages offered by the additional information provided by the class hierarchy.

In the second investigation, within the few-shot setting, we have proposed a hierarchical prototypical network for image scene classification. More precisely, we have augmented the traditional prototypical network establishing prototypes at every level of the class hierarchy, rather than solely at the leaf node level. The class hierarchy information is then brought in through these hierarchical prototypes which will be involved as part of a weighted sum of the cross-entropy loss over the different levels of the class hierarchy. The experimental results showed the advantages of utilising the class hierarchy for addressing the few-shot RSISC problem. Our hierarchical prototypical network provided regularisation of the latent space and achieved better performance compared to the classical counterpart.

In the forthcoming chapter, we examine alternative methods for incorporating hierarchical information, specifically implicit hierarchical information. To accomplish this, we utilise hyperbolic space, which has demonstrated its effectiveness in embedding hierarchical data or data with underlying hierarchies.

Chapter 4

Classification of remote sensing scene images in the hyperbolic space

This chapter is built upon the research presented in the two articles "*hyperbolic Prototypical Network for Few Shot Remote Sensing Scene Classification*", which has been submitted to PRL, and "*hyperbolic variational auto-encoder for remote sensing scene embeddings*", which has been accepted for presentation at IGARSS23.

4.1 Introduction

In this second part of the thesis, we focus on approaches that deal with the implicit hierarchical information by operating in hyperbolic space. Such a space has demonstrated its suitability for representing hierarchical data or data with an underlying hierarchy (Nickel & Kiela, 2017; Peng et al., 2022). The objective of this chapter is therefore to investigate the potential of hyperbolic representations in the context of RS data, with a specific focus on scene images. Additionally, we will assess whether the claims made in prior works utilising hyperbolic space hold true in this context.

Limited studies within the image community have considered hyperbolic space as an embedding space despite its popularity within the ML community. We mention two hyperbolic studies that we believe are worth investigating: hyperbolic VAE, which is among the first frameworks to deal with images in a hyperbolic space, and prototypical networks, which were introduced in a pioneering study addressing image embedding in hyperbolic space. It is noteworthy that, to our knowledge, the investigations conducted by (Li et al., 2022b; Sun et al., 2022) represent the only studies within the RS community that has examined hyperbolic space. The authors in (Sun et al., 2022) aimed to reduce the dimensionality of hyperspectral images (HSI) by employing an unsupervised approach to

select more consistent bands. On the other hand, the authors in (Li et al., 2022b) proposed a hybrid attention module (SHAM) to perform semantic segmentation.

Hyperbolic VAEs (H-VAEs) have been successfully used to embed data into a hyperbolic space (Mathieu et al., 2019; Nagano et al., 2019) so that meaningful features can be extracted. They were validated on the MNIST and Atari 2600 Breakout datasets by performing a classification step on the resulting embeddings, which showed that the H-VAE is able to better embed the data. Furthermore, despite the absence of a clear hierarchy within these datasets, in particular MNIST dataset, a hierarchical structure was induced. This suggests that even better results can be anticipated for images that possess a genuine hierarchical arrangement, namely RS scene images.

The first section of this chapter therefore investigates the suitability of H-VAE to embed remote sensing scene images in hyperbolic space and whether hierarchical information among scene classes can be recovered. The second part of this chapter is dedicated to hyperbolic prototypical networks. Here we use the prototypical network defined in the hyperbolic space, particularly the Poincaré Ball model, to better classify scene images within the few-shot setting.

4.2 Hyperbolic variational auto-encoder for remote sensing scene embeddings

Hyperbolic variational auto-encoder (H-VAE) was introduced by (Mathieu et al., 2019). In this work, the authors proposed a generalisation of the normal distribution to the hyperbolic space, in particular the Poincaré model. They evaluated the H-VAE on the MNIST dataset and demonstrated outstanding results. Furthermore, despite the non-obvious hierarchy present in MNIST, H-VAE was able to recover the underlying hierarchical structure of the data through its hyperbolic latent space. This highlights the potential of H-VAE in capturing underlying hierarchical representations even in datasets lacking obvious hierarchical information. Moreover, H-VAE produced high-quality representations at low-dimensional embedding space. In line with the work of (Mathieu et al., 2019), (Nagano et al., 2019) proposed an alternative to the normal distribution in the Lorentz model, which was used to define a H-VAE in the Lorentz model. They evaluated their H-VAE on both the MNIST and Atari 2600 Breakout datasets, and demonstrated superior performance compared to the Euclidean VAE (E-VAE).

Several researchers were inspired by the interesting outcomes of the aforementioned

studies and sought to explore alternative ways of modelling data. As such, H-VAEs have been employed in various tasks across different types of data, including text generation (Dai et al., 2021), unsupervised segmentation of 3D voxel-grid biomedical images (Hsu et al., 2021) and semi-supervised drug embedding (Yu et al., 2020). In each of these tasks, H-VAE has consistently demonstrated superiority over its Euclidean counterpart.

Driven by the effectiveness of H-VAE and its promising outcomes, this section aims to employ an H-VAE to embed remote sensing scene images in hyperbolic space. Our objective is to investigate whether the performance of H-VAE is superior to E-VAE when dealing with hierarchically-structured data. To achieve this, we assess the quality of both H-VAE and E-VAE embeddings and, consequently, their latent spaces by solving a classification task.

4.2.1 Overall framework

Inspired from previous studies (Mathieu et al., 2019; Nagano et al., 2019), we adopt a hybrid architecture of the hyperbolic VAE in which the encoder and decoder networks are Euclidean networks and only the latent space of the VAE is hyperbolic. The wrapped normal distribution (Nagano et al., 2019) is a generalisation of this distribution to hyperbolic space, namely the Lorentz model, which we discuss in more detail in subsection 4.2.3. Furthermore, we add the Euclidean feature clipping technique (Guo et al., 2022b) to overcome the numerical problems arising from hyperbolic projection operations that result in out-of-space embeddings due to unrepresentable values in floating point arithmetic. Feature clipping limits the effective radius of the Poincaré Ball model (Nickel & Kiela, 2017), which pushes the hyperbolic embeddings further from the boundary. However, in the Lorentz model, it constrains the tangent embeddings to remain proximate to the Lorentz origin in order to ensure numerical stability of the Lorentz projection. We provide further details on feature clipping in the section below. The overall framework of the approach is illustrated in Figure 4.1.

4.2.2 Feature clipping

Extending deep neural networks to hyperbolic space is a challenging task considering the generalisation complexity of the basic required operations. Therefore, the majority of studies on hyperbolic space use hybrid "Euclidean-hyperbolic" architectures. However, passing between the Euclidean and hyperbolic layers of these hybrid architectures often

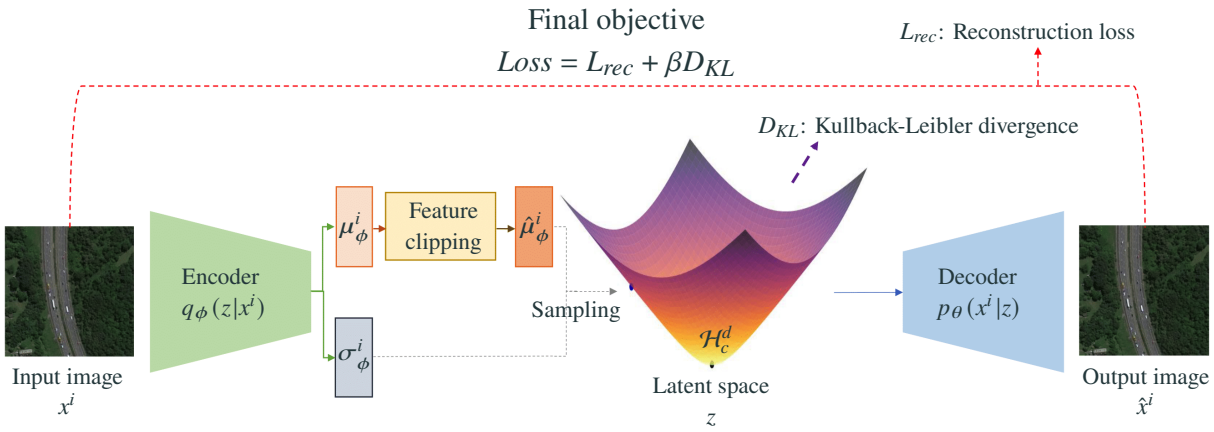


Figure 4.1: Overview of the hyperbolic VAE for remote sensing image embeddings.

leads to numerical problems resulting in the vanishing of the gradient. (Guo et al., 2022b) suggested that the Euclidean features should be clipped before moving to the hyperbolic layers which allows to push the hyperbolic embeddings further away from the Poincaré boundary. Figure 4.2 illustrates the relationship between the clipping value and the effective radius of the Poincaré Ball. The clipping technique allows hybrid architectures to cope with numerical problems, which seems to avoid the vanishing gradient problem. Furthermore, it enhances the performance of hyperbolic networks and makes their behaviour steadier; it is defined as:

$$x_r^E = \min\left\{1, \frac{r}{||x^E||}\right\}.x^E \tag{4.1}$$

where x_r^E is the clipped embedding of x^E which lies in the Euclidean space and r is the clipping value.

We consider the Lorentz model as the hyperbolic space of the H-VAE, as it allows a better generalisation of the normal distribution in the hyperbolic space (Nagano et al., 2019). Thus, the clipping here constrains the Euclidean embeddings which are in the origin tangent space to remain close to the origin in order to ensure the numerical stability of the Lorentz projection.

4.2.3 Hyperbolic Variational Auto-Encoder

Hyperbolic Variational Auto-Encoder (H-VAE) is a variant of VAE (we choose the E-VAE notation for Euclidean VAE) in which the latent variables are defined on a hyperbolic space. A wrapped normal distribution was proposed by (Nagano et al., 2019) for the

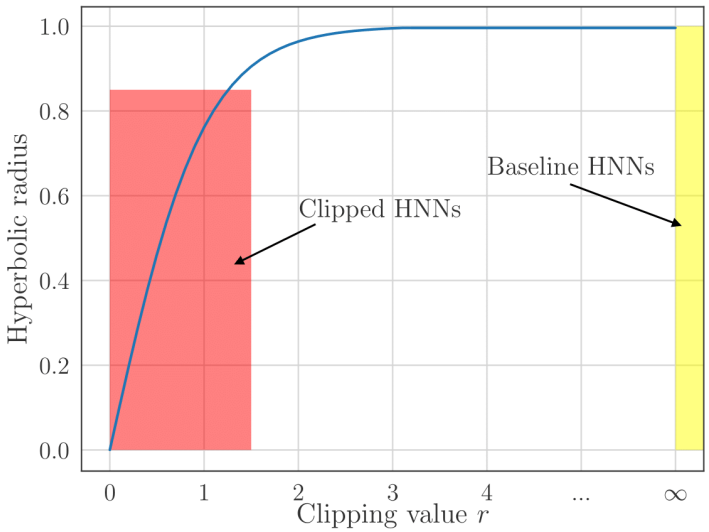


Figure 4.2: The relation between the clipping value r and the effective radius of the Poincaré Ball ($c = 1$).

Lorentz model, which we denote $\mathcal{G}(\mu, \Sigma)$, where $\mu \in \mathcal{H}_1^d$ and Σ are defined as positive. In the following, the curvature of the Lorentz model is set to -1, we thus drop the curvature c from the hyperbolic notations for simplicity. Sampling from the distribution $\mathcal{G}(\mu, \Sigma)$ can be summarised in 3 steps:

(1) sample a vector from the Gaussian distribution $\tilde{v} \sim \mathcal{N}(0, \Sigma)$ and interpret it as an element of the tangent space at the origin μ_0, $v = [0, \tilde{v}] \in \mathcal{T}_{\mu_0}\mathcal{H}^d$;

(2) parallel transport $v \in \mathcal{T}_{\mu_0}\mathcal{H}^d$ to the tangent space of the desired location μ, $u = \mathrm{PT}_{\mu_0 \to \mu}(v)$;

(3) use $\exp_\mu$ to map the transported vector u from the tangent space $\mathcal{T}_\mu\mathcal{H}^d$ to the manifold $\mathcal{H}^d$, $z = \exp_\mu(u)$.

This sampling strategy, which is summarised in Algorithm 1, is used in the H-VAE as a reparameterisation trick. Therefore our hyperbolic latent variables $z^{(i,l)} \sim q_\phi(z|x^i)$ are defined as:

$$z^{(i,l)} = g_\phi\left(v^{(i,l)}, \mu_\phi^i\right) = \exp_{\mu_\phi^i}\left(PT_{\mu_0 \to \mu_\phi^i}\left(v^{(i,l)}\right)\right) \tag{4.2}$$

where $v^{(i,l)} = \left[0, \tilde{v}^{(i,l)}\right]$ and $\tilde{v}^{(i,l)} \sim \mathcal{N}(0, \Sigma_\phi^i)$. Σ_ϕ^i and μ_ϕ^i are outputs of the encoder. μ_ϕ^i is assured to be in $\mathcal{H}^d$ by applying $\exp_{\mu_0}$ to the final layer of the encoder.

The Kullback-Leibler divergence (Eq. 3.3) must also be adapted to the hyperbolic

Algorithm 1 reparameterisation trick in the Lorentz model

Inputs: parameters $\mu \in \mathcal{H}^d$, Σ
Output: $z \in \mathcal{H}^d$
Require: $\mu_0 = \left[\frac{1}{\sqrt{c}}, 0, \ldots, 0\right]^T \in \mathcal{H}^d$
Sample $\tilde{v} \sim \mathcal{N}(0, \Sigma) \in \mathbb{R}^d$
$v = [0, \tilde{v}] \in \mathcal{T}_{\mu_0}\mathcal{H}^d$
Move $v \in \mathcal{T}_{\mu_0}\mathcal{H}^d$ to $u \in \mathcal{T}_{\mu}\mathcal{H}^d$: $u = \mathrm{PT}_{\mu_0 \to \mu}(v)$ ▷ Eq. 2.12
Map $u \in \mathcal{T}_{\mu}\mathcal{H}^d$ to $z \in \mathcal{H}^d$: $z = \exp_{\mu}(u)$ ▷ Eq. 2.10

space. We therefore need to extend the logarithmic probability density function $p(z)$ to Lorentz model as follows:

$$\log p(z) = \log p(v) - (d-1) \log \left(\frac{\sinh \left(\|v\|_{\mathcal{L}} \right)}{\|v\|_{\mathcal{L}}} \right) \tag{4.3}$$

where $z \in \mathcal{H}^d$, v is defined in the algorithmic description Algorithm 2 which summarises the calculation of the logarithmic probability density function (pdf) in the Lorentz model.

Algorithm 2 Log-pdf calculation in the Lorentz model

Inputs: sample $z \in \mathcal{H}^d$, parameters $\mu \in \mathcal{H}^d$, Σ
Output: $\log p(z)$
Require: $\mu_0 = [1, 0, \ldots, 0]^T \in \mathcal{H}^d$
Map $z \in \mathcal{H}^d$ to $u \in \mathcal{T}_{\mu}\mathcal{H}^d$: $u = \log_{\mu}(z) \in \mathcal{T}_{\mu}\mathcal{H}^d$ ▷ Eq. 2.11
Move $u \in \mathcal{T}_{\mu}\mathcal{H}^d$ to $v \in \mathcal{T}_{\mu_0}\mathcal{H}^d$: $v = \mathrm{PT}_{\mu \to \mu_0}(u) \in \mathcal{T}_{\mu_0}\mathcal{H}^d$ ▷ Eq. 2.12
Calculate $\log p(z)$ according to Eq. 4.3

4.2.4 Experimental study

This study focuses on the relevance of hyperbolic space in the context of remote sensing. The objective is therefore to investigate whether hyperbolic space fulfils its promise in the context of remote sensing and outperforms Euclidean space. In this perspective, for both the E-VAE and the H-VAE, we adopt a very simple VAE architecture with regard to those used recently in the remote sensing community (Cheng et al., 2017; Cheng et al., 2020; Dutta & Das, 2023). We evaluate the quality of the resulting embeddings and the ability to discriminate between classes using the simple $1-$NN classifier.

Experimental setup

Dataset The two models are learned on a subset of the NWPU-RESISC45 (Cheng et al., 2017) remote sensing scene dataset. All 45 classes are considered, for each, we randomly select 100 images for the training set, 50 images for the validation set and 80 images for the test set.

Implementation Details For both the E-VAE and the H-VAE, we choose the same following architecture. Both the encoder and the decoder are composed of 5 convolutional layers and a linear layer, each convolutional layer is followed by a batch normalisation layer and a Leaky ReLU activation, except for the decoder last convolutional layer which is followed by a tanh activation. The input size of the encoder network is set to 64×64. The latent space dimension d of the embedding z is set to 8, 16, 32, 64 and 128 respectively.

The Adam optimiser (Kingma & Ba, 2015) acts as our optimiser with a constant learning rate of $1e^{-3}$. The models are trained with mini-batches of size 64 for 1500 epochs with an early stopping of 50 epochs. The ELBO term is approximated by Monte Carlo (MC) estimation with $L = 1$. β scaling hyper-parameter weight of the KL divergence was chosen experimentally and set to $5e^{-5}$. The clipping hyper-parameter r is cross-validated.

Results and discussion

To compare our hyperbolic VAE (H-VAE) not only to its Euclidean counterpart, but also the reference model (H-VAE without clipping), we evaluate the quality of the resulting embeddings of different VAEs and the ability to discriminate between classes using the simple $1-$NN classifier. Experiments are conducted on a subset of the NWPU-RESISC45 dataset and reported in Table 4.1, results are averaged over 3 runs. The reported scores correspond to models trained with hyper-parameters providing most times the best performance across different dimensions (clipping value $r = 1$).

Prior studies (Mathieu et al., 2019; Nagano et al., 2019) have demonstrated the superiority of hyperbolic VAE w.r.t. its Euclidean counterpart in various context (images and graphs). However, this observation does not hold in our context. We further investigate this behaviour and we show that it is due to the numerical problems arising from hyperbolic projection operations that result in out-of-space embeddings. Prior studies (Mathieu et al., 2019; Nagano et al., 2019) have demonstrated the superiority of hyperbolic VAE w.r.t. its Euclidean counterpart in various context (images and graphs). However, this observation does not hold in our context. To avoid this problem, we suggest utilising feature clipping as

Table 4.1: 1-NN classification results computed on the test set of the NWPU-RESISC45 dataset at different levels of the class hierarchy: overall acc represents the classification accuracy at the leaves (level 4) and thus the NWPU-RESISC45 classes; L3-acc and L2-acc give the accuracy at level 3 and level 2, respectively (the higher the better); Results are averaged over 3 runs.

Space	Clip r	Metric	Latent Space Dimension d				
			8	16	32	64	128
E-VAE	/	Overall acc	12.00 ± 0.15	13.96 ± 0.56	13.21 ± 0.21	12.08 ± 0.10	12.39 ± 0.32
		L3-acc	18.38 ± 0.42	20.64 ± 0.41	19.33 ± 0.13	17.93 ± 0.57	17.88 ± 0.24
		L2-acc	28.34 ± 0.57	31.49 ± 0.38	30.97 ± 0.16	29.96 ± 0.47	29.95 ± 0.49
H-VAE	None	Overall acc	11.38 ± 0.58	11.53 ± 0.30	10.77 ± 1.02	10.21 ± 0.96	11.33 ± 0.53
		L3-acc	17.83 ± 0.58	17.45 ± 0.45	15.90 ± 0.66	14.82 ± 0.90	16.43 ± 1.02
		L2-acc	28.58 ± 0.68	28.46 ± 0.33	27.85 ± 0.83	29.96 ± 1.71	$29,77 \pm 0,90$
	1	Overall acc	$\mathbf{12.36 \pm 0.53}$	$\mathbf{14.18 \pm 0.42}$	$\mathbf{14.17 \pm 0.42}$	$\mathbf{14.18 \pm 0.44}$	$\mathbf{12.87 \pm 0.54}$
		L3-acc	$\mathbf{18.80 \pm 0.61}$	$\mathbf{20.89 \pm 0.59}$	$\mathbf{20.50 \pm 0.22}$	$\mathbf{20.00 \pm 0.71}$	$\mathbf{18.39 \pm 0.82}$
		L2-acc	$\mathbf{28.46 \pm 0.40}$	$\mathbf{31.54 \pm 0.67}$	$\mathbf{31.91 \pm 0.43}$	$\mathbf{31.66 \pm 0.78}$	$\mathbf{30.11 \pm 0.73}$

proposed in (Guo et al., 2022b). Consequently, we observe an improvement of $0.22 - 2.10\%$ and $0.98 - 3.97\%$ compared to the E-VAE and the baseline H-VAE, respectively, across the latent space dimension.

We note that the low accuracy values are due to the choice of the VAE architecture. Remote sensing data are complex and require very deep networks with a large amount of data to reach high performances. This was not used in this study since the focus was on comparing hyperbolic and Euclidean spaces rather than achieving the best results.

Impact of the clipping value Figure 4.3 shows the $1-$NN classification accuracy values on the test set in function of the clipping value r.

We observe that the H-VAE generally performs better with small values of the clipping hyper-parameter ($r < 1.2$). Larger clipping values often result in Euclidean tangent features far from the space origin. The projection operation (Eq. (2.10)) is employed to map these Euclidean tangent features to the Lorentz model. Nonetheless, in this scenario, performing such an operation necessitates a remarkably high floating point precision (i.e., a considerable number of bits) to adequately represent the resulting embeddings in the Lorentz model. This, however, is not feasible in PyTorch, as double precision is the highest floating-point number available, occupying 64 bits. The possibility of out-of-space embeddings therefore increases, leading to numerical instability of the network, which is reflected by the significant decrease of classification scores across dimensions.

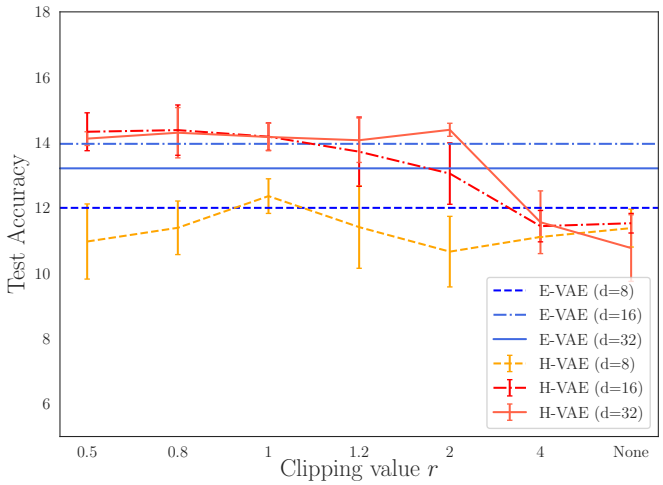


Figure 4.3: 1-NN classification accuracy of different VAE models on a subset of the NWPU-RESISC45 remote sensing scene dataset w.r.t. the clipping value r.

4.2.5 Conclusion

In this section, we investigated our hypothesis that hyperbolic space is better suited for handling RS scene images, which present an underlying hierarchical structure, compared to Euclidean space. To accomplish this, we utilised the VAE framework and compared the performance of the H-VAE against its Euclidean counterpart, the E-VAE. We confirmed the superiority of hyperbolic embeddings, in particular Lorentz embeddings, over Euclidean embeddings via the simple 1-NN classifier. We have also shown the importance of hyper-parameters, in particular the clipping which ensures a better numerical stability of the network during the learning, allowing hyperbolic embeddings to outperform their Euclidean counterparts.

Since our intuition has been confirmed in the HVAE context, we are now examining a more challenging context, namely few-shot learning.

4.3 Hyperbolic prototypical network for few-shot remote sensing scene classification

After the successful application of hyperbolic space to embed hierarchical data, the computer vision community began to take notice of this space, dedicating efforts to perform advanced studies on complex image datasets.

The work on hyperbolic image embedding (Khrulkov et al., 2020) was among the pioneering research studies which inspired further research in computer vision to consider hyperbolic space as a possible substitute for Euclidean space. (Liu et al., 2020a) presented a framework for embedding images and their corresponding semantic classes, provided by the class hierarchy and class descriptions, into a shared hyperbolic space, with the aim of addressing the zero-shot problem. Similarly, (Zhang et al., 2022) proposed a framework for few-shot classification, which comprises a fully hyperbolic network which takes precomputed features, along with a module that learns class prototypes from a hierarchical organisation of classes. (Ermolov et al., 2022) introduced transformers to the hyperbolic space, which were learned through pairwise cross-entropy loss, to enable image retrieval.

The aforementioned examples represent only a fraction of the research conducted on the utilisation of hyperbolic space in addressing computer vision problems. Although these works mainly focus on image classification and retrieval, it is worth noting that there are other studies that explore the application of hyperbolic space in object detection (Ge et al., 2022; Lang et al., 2022) and semantic segmentation (Chen et al., 2022a; Li et al., 2022b) tasks. For instance, (Ge et al., 2022) suggested a contrastive learning framework that combines Euclidean and hyperbolic spaces. Specifically, object embeddings were learned in the Euclidean space, while scene embeddings were encouraged to be located near the representations of their component objects in the hyperbolic space. (Chen et al., 2022a) designed a Hyperbolic Uncertainty Loss (HyperUL), which is an extension of the cross-entropy loss, to address the issue of uncertainty in semantic segmentation. The authors employed the hyperbolic distance metric to estimate the uncertainty of individual pixels in a more efficient manner. They then leveraged these calculated uncertainties as weights to adjust the training process for each pixel.

In this previous part of the chapter, we closely examined the dominant trend in image analysis within hyperbolic space, which aligns with our theme. In the upcoming sections, we will investigate the scene image representations obtained via a hyperbolic prototypical network and validate their effectiveness by performing few-shot classification. Unlike the previous section, we consider here the Poincaré Ball model as a hyperbolic space, which is found to be more efficient in representing image data than the Lorentz model (Guo et al., 2022b).

4.3.1 Hyperbolic prototypical network

To extend prototypical networks to the Poincaré Ball model and to perform different operations in this hyperbolic space, we first need to map the Euclidean features to the hyperbolic space (Eq. 2.3). The calculation of the class prototypes is then simply carried out in the Klein model via the *Einstein midpoint*, which is summarised in the Algorithm 3.

Algorithm 3 Poincaré prototypes

Inputs: $(x_i, y_i) \in S^k$ where $x_i \in \mathbb{B}_c^d$
Output: $p^k \in \mathbb{B}_c^d$
Project Poincaré embeddings x_i into Klein model : $\hat{x}_i = \Pi_{\mathbb{B}_c^d \to \mathbb{K}_c^d}(x_i) \in \mathbb{K}_c^d$ ▷ Eq. 2.17
Compute class prototype via *Einstein midpoint*: $\hat{p}^k = \text{HypAve}(\hat{x}_i) \in \mathbb{K}_c^d$ ▷ Eq. 2.16
Project class prototype $\hat{p}^k$ into Poincaré Ball model: $p^k = \Pi_{\mathbb{K}_c^d \to \mathbb{B}_c^d}\left(\hat{p}^k\right) \in \mathbb{B}_c^d$ ▷ Eq. 2.18

However, hyperbolic prototypical networks also adopt a hybrid network architecture. Only final operations are performed in hyperbolic space, namely the computation of prototypes and distances which are used to derive class membership probabilities for queries. The previous step, namely the feature extraction, is performed in the Euclidean space. We therefore require clipping of Euclidean features before moving to the hyperbolic space in order to ensure numerical stability and to improve the performance of hyperbolic networks. Figure 4.4 illustrates the overall framework of the hyperbolic prototypical network which we use to tackle the few-shot remote sensing scene classification problem.

4.3.2 Experimental study

Experimental setup

Dataset As in Section 3.3.4, the NWPU-RESISC45 dataset is splited into three disjoint subsets: meta-train, meta-validation and meta-test containing respectively 25, 12 and 8 categories. Furthermore, the meta-training set is also split into three subsets: training, validation and testing sets. Both meta-validation and validation sets are utilised in selecting the optimal model and hyper-parameters. We follow (Zhang et al., 2021a) and scale all images to 80×80 pixels to fit the feature extractor.

Implementation Details Following recent FSRSSC studies (Li et al., 2021c; Zhang et al., 2021a; Zhang et al., 2021b; Zhang et al., 2021c), we use ResNet-12 as a backbone for feature extraction. We also adopt the pre-training strategy as suggested in (Zhang et al.,

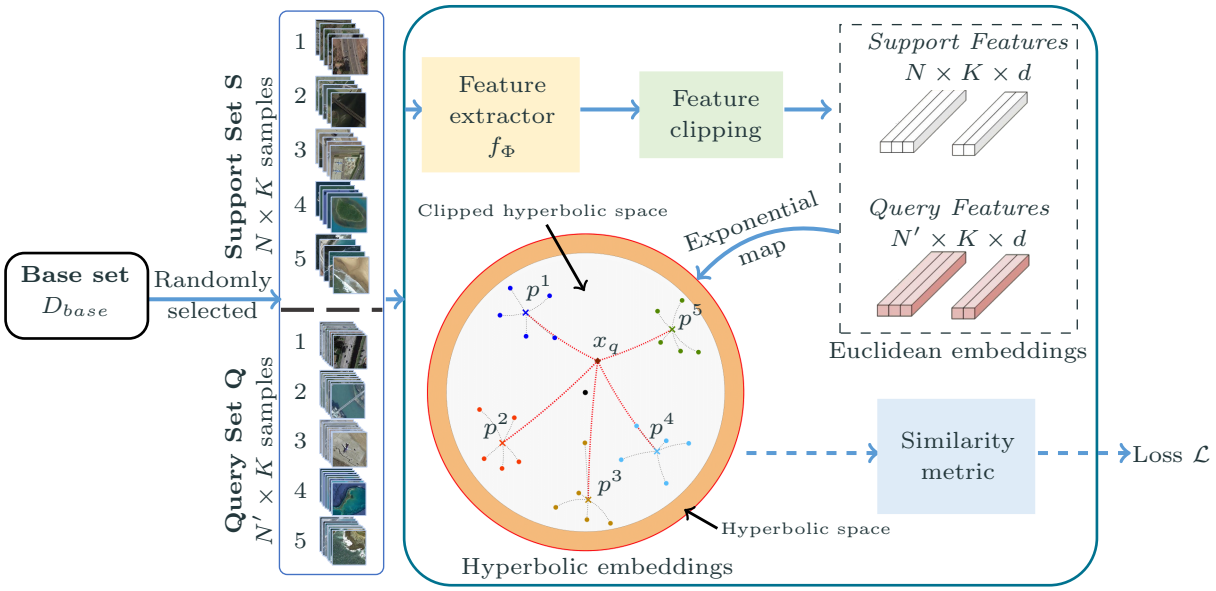


Figure 4.4: Overall framework of the hyperbolic prototypical network for few-shot image classification. In this example (5-shot), $K = 5$, $N = 5$, $N' \geq 1$ (usually set to 15) and the latent space dimension $d = 2$.

2021a) to better initialise the meta-learner feature extractor and enhance its performances. For both the Euclidean and hyperbolic approaches, we train the meta-learner for 400 epochs, with the best model parameters been selected based on the highest accuracy over the validation set.

During training, we randomly sample 500 episodes in each epoch to learn the model which parameters are updated every 2 episodes, i.e. the batch size is set to 2 and the average loss over the 2 episodes is used to learn the network parameters. The SGD optimiser is used to update the network parameters with momentum set to 0.9 and weight decay set to 0.0005. For the hyperbolic parameters, we use the Riemannian SGD optimiser (Bonnabel, 2013). The learning rate is set to 10^{-4}. We evaluate the learned model on a validation set after each training epoch by randomly sampling 500 episodes, the network weights with the highest accuracy over the validation set are retained as the best parameters.

For the meta-testing stage, we evaluate the best model on 1000 randomly sampled episodes from the test set D_{novel}.

For the hyperbolic hyper-parameters, we follow (Khrulkov et al., 2020) and set the hyperbolic curvature c to 0.01, in both pre-training and meta-training while the *clip* value and the temperature τ are cross-validated. A sensitivity study of these hyper-parameters is further discussed in Section 4.3.2.

Results and discussion

In both 5-way N-shot configurations (with $N = 1$ or 5), we compare our hyperbolic prototypical network (*H-ProtoNet*) not only to the reference model (Khrulkov et al., 2020) but also to the Euclidean prototypical network (Snell et al., 2017) counterpart (*E-ProtoNet*). We also compare with the cosine metric as a similarity function (*C-ProtoNet*) as advocated in a similar context (Zhang et al., 2021a). We re-implement these methods and cross-validate their hyper-parameters for a fair comparison.

Table 4.2: 1−shot classification results computed on the NWPU-RESISC45 test set. All accuracy results are averaged over 1000 test episodes and are reported with a 95% confidence interval. * refers to the baseline model (Khrulkov et al., 2020).

Approach	Metric	Latent Space Dimension d		
		32	128	512
E-ProtoNet	Acc	62.40 ± 0.62	64.15 ± 0.65	63.89 ± 0.59
	P_H	67.75 ± 0.59	69.38 ± 0.60	69.27 ± 0.55
C-ProtoNet	Acc	61.58 ± 0.65	63.85 ± 0.63	63.72 ± 0.62
	P_H	67.15 ± 0.60	69.03 ± 0.59	69.05 ± 0.57
H-ProtoNet*	Acc	56.89 ± 0.64	61.69 ± 0.63	63.48 ± 0.62
	P_H	62.87 ± 0.62	67.37 ± 0.60	69.07 ± 0.57
H-ProtoNet (ours)	Acc	$\mathbf{63.30 \pm 0.63}$	$\mathbf{66.05 \pm 0.64}$	$\mathbf{65.09 \pm 0.63}$
	P_H	$\mathbf{68.69 \pm 0.59}$	$\mathbf{71.26 \pm 0.58}$	$\mathbf{70.39 \pm 0.58}$

Table 4.3: 5−shot classification results computed on the NWPU-RESISC45 test set. All accuracy results are averaged over 1000 test episodes and are reported with a 95% confidence interval. * refers to the baseline model (Khrulkov et al., 2020).

Approach	Metric	Latent Space Dimension d		
		32	128	512
E-ProtoNet	Acc	78.96 ± 0.36	80.24 ± 0.37	77.79 ± 0.41
	P_H	82.19 ± 0.34	83.52 ± 0.33	81.26 ± 0.37
C-ProtoNet	Acc	76.76 ± 0.41	79.76 ± 0.37	78.75 ± 0.40
	P_H	80.43 ± 0.36	83.03 ± 0.34	82.09 ± 0.36
H-ProtoNet*	Acc	74.31 ± 0.41	78.45 ± 0.38	78.24 ± 0.38
	P_H	77.96 ± 0.39	81.95 ± 0.36	81.93 ± 0.33
H-ProtoNet (ours)	Acc	$\mathbf{79.37 \pm 0.39}$	$\mathbf{82.75 \pm 0.33}$	$\mathbf{80.74 \pm 0.37}$
	P_H	$\mathbf{82.79 \pm 0.34}$	$\mathbf{85.81 \pm 0.29}$	$\mathbf{84.04 \pm 0.32}$

Table 4.2 and Table 4.3 reports the classification accuracy on various latent space dimensions for 1-shot and 5-shot respectively. We stopped at dimension 32 as going further deteriorated the performance considerably. We report accuracy values and the hierarchical precision which is a global hierarchical metric to assess the model's ability to better reflect the semantic relationships between the novel scene categories.

We observe that the baseline H-ProtoNet (Khrulkov et al., 2020), which was reported to perform well in the literature, actually performs worse than both Euclidean ProtoNets

(E-ProtoNet and C-ProtoNet). However, our H-ProtoNet, which uses the clipping technique, outperforms not only the baseline H-ProtoNet but also both Euclidean ProtoNets.

Interestingly, our H-ProtoNet also improves the hierarchical precision over different dimensions of the latent space, especially in the 5-shot scenario. This supports to some extent our hypothesis that hyperbolic geometry, if carefully tackled, better handles data with an underlying hierarchical structure and allows for more meaningful embeddings.

Impact of hyper-parameters

Table 4.3 shows the scores of models trained with hyper-parameters providing most times the best performance across different dimensions. In this section, we investigate the impact of the hyper-parameter values on the performance of the three approaches. For a better understanding of the investigation, we only report the study done for 5-shot setting on dimension 128 which yields the best scores.

The inverse-temperature $1/\tau$ In Figure 4.5, we compare the test accuracy of the three models: E-ProtoNet, C-ProtoNet and our H-ProtoNet with respect to different $1/\tau$ values.

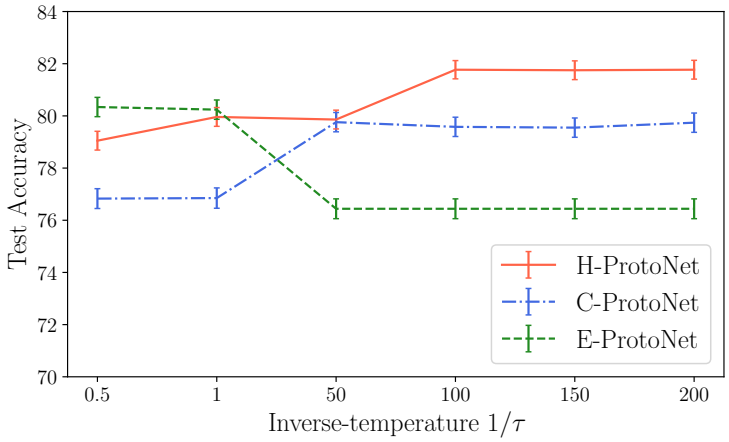


Figure 4.5: Test accuracy w.r.t. the inverse-temperature $1/\tau$.

We observe that small $1/\tau$ values are beneficial for E-ProtoNet unlike C-ProtoNet and our H-ProtoNet which favour larger values. Moreover, we notice that the latter are robust

when $1/\tau \geq 50$ and $1/\tau \geq 100$, respectively. Accordingly, we select best $1/\tau$ values, i.e. 1, 50 and 100 for the E-ProtoNet, C-ProtoNet and our H-ProtoNet, respectively

Feature clipping Figure 4.6 shows the classification accuracy on the test set in function of the clipping value r while fixing the $1/\tau$ to the value that yielded best performance.

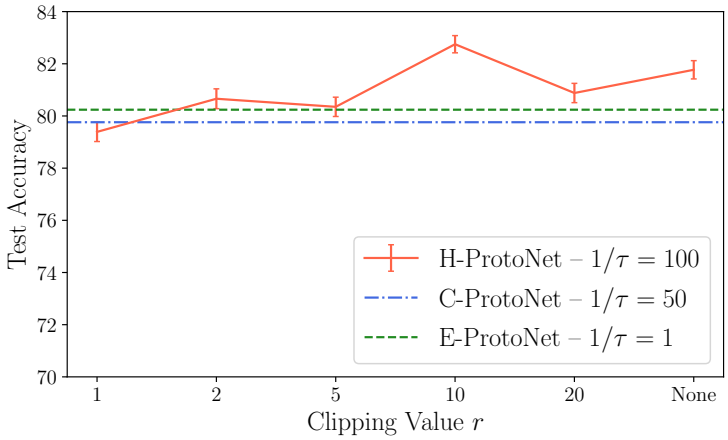


Figure 4.6: Test accuracy w.r.t. the clipping value r.

We observe that the model performs better with larger values of the clipping hyper-parameter r. Small clipping values constrain too much the latent space forcing hyperbolic embeddings to be positioned very close to the centre. This results in small hyperbolic distances between the query representations and the class prototypes, which reduces the benefits of working in hyperbolic space and makes model optimisation more difficult.

The higher the clipping values, the larger the effective hyperbolic latent space, tending towards a hyperbolic space without clipping.

Accordingly, we adopt a clipping of 10 which yields the best performance and corresponds approximately to 80% of the hyperbolic radius, resulting in a satisfactorily large hyperbolic space, while preventing the hyperbolic embeddings being close to the space boundary.

4.3.3 Conclusion

In this section, we investigate the ability of hyperbolic space, in particular the Poincaré Ball model, to handle a difficult task in the context of remote sensing, which is the few

shot scene classification, since the scene categories usually have an intrinsic hierarchical structure. We show that hyperbolic prototypes better encode the scene images and the intrinsic hierarchical relationships among them, providing a better latent space arrangement and thus higher performance than Euclidean prototypes. However, in practice, we have observed that great care is required when dealing with hyper-parameters. Specifically, we have identified two hyper-parameters that are especially important in our context: the clipping value and the temperature.

4.4 Chapter Summary

In contrast to the preceding chapter, where we introduced the class hierarchy explicitly, the current chapter was centred on the second type of hierarchical information which is implicitly inherent among the image data. In line with the current trend in the machine learning community, we utilised the hyperbolic space as an embedding space as it has been demonstrated to be highly suitable for embedding data with an inherent hierarchy. Much like the preceding chapter, we investigated two settings in this study: an unsupervised setting and a few-shot setting.

In the unsupervised setting, similar to chapter 3, we adopted the VAE framework to embed scene images. To be more specific, we utilised the extension of the normal distribution to hyperbolic space in order to construct our hyperbolic VAE. This entails retaining the entire VAE network in Euclidean space while generalising solely its latent space, which conforms to a normal distribution.

Similar to the approach in chapter 3, in the few-shot setting, we employed the prototypical network for scene image embeddings. The generalisation of the latter is achieved by adding a layer that enables the transition from the Euclidean features generated by the feature extractor to the hyperbolic space.

Since both are hybrid Euclidean-hyperbolic architectures, we utilised Euclidean feature clipping to ensure the numerical stability of both models.

In both unsupervised and few-shot settings, we have shown the superiority of the hyperbolic embeddings of RS scene images over their Euclidean counterparts. Nevertheless, operating within these spaces is not necessarily a straightforward process and does not necessarily guarantee superior results from the initial application. It is crucial to be mindful of hyper-parameters, such as the clipping value, which has a significant influence on the effectiveness of hyperbolic algorithms. Ensuring numerical stability in hyperbolic space

remains a major challenge. However, feature clipping is a simple yet effective solution to address this issue.

In summary, this chapter serves as an opening towards the application of hyperbolic space to remote sensing images. Although challenging, the properties of remote sensing images align well with the geometric properties of hyperbolic space, presenting a promising avenue for future research in this field.

Chapter 5

Conclusion and further works

In this concluding chapter, we aim to provide a summary of our main contributions in section 5.1. In section 5.2, we outline several future research directions, which have the potential to enhance the methods proposed in this thesis as well as to open up new perspectives for the remote sensing community related to the hierarchical image classification.

5.1 Conclusion

This thesis focused on the hierarchical information present among remote sensing scene images, which has received limited attention so far. Two primary methodologies were proposed to handle this information, which resulted in two main contributions.

As a first contribution of this thesis, we proposed a technique to leverage the explicit hierarchical information about the RS scene classes, which is provided by a multi-level class hierarchy, when learning the latent scene representation. Initially, we examined the effectiveness of the hierarchy in improving the scene representation in a generative context. Once we confirmed this hypothesis, we proposed a new hierarchical loss-based approach that incorporates the class hierarchy via hierarchical prototypes, which were assessed in the context of few-shot learning. Our experiments showed that the hierarchical approach outperformed its respective *flat* counterpart, highlighting the potential of utilising class hierarchy information to enhance remote sensing image scene classification performance.

The second contribution centred around the implicit form of the hierarchical information. As such, we proposed using the hyperbolic space as an embedding space, since it has been proven to handle data with an underlying hierarchy more effectively than the Euclidean space. During the initial phase of this thesis, which aligned with the early stages of learning in hyperbolic space, we investigated this research direction. We suggested to

embed RS scene images via a hyperbolic variational auto-encoder (H-VAE) and assess the quality of its embeddings, as well as its latent spaces, by solving a classification task. However, we encountered unexpected performance results (Hamzaoui et al., 2021) that were not consistent with what has been reported in the literature. We therefore focused on the introduction of the explicit hierarchy. Over time, new techniques and a deeper comprehension of hyperbolic space came to light (Guo et al., 2022b). Subsequently, we revisited the hyperbolic spaces. We explored two different settings, an unsupervised setting which is a revised H-VAE and a few-shot one. Our experimental results in both scenarios confirmed that, as we initially anticipated, hyperbolic space is well-suited for RS data, particularly for scene image classification. Our findings highlighted the significance of clipping technique (Guo et al., 2022b) as it is essential to ensure the superiority of hyperbolic space in our context, in contrast to research in other fields.

5.2 Perspectives

In light of the findings presented in this thesis, there are a number of perspectives that can guide future research in this field. First, we summarise various direct improvements of the proposed methods. Subsequently, we suggest some potential future research topics concerning the application of hyperbolic space in remote sensing, drawing inspiration from the work presented in this study.

5.2.1 Perspectives of our contributions

Extensions to hyperbolic space

Having gained a better understanding of how to achieve successful outcomes in the hyperbolic space, we can now reexamine the mid-thesis contributions (Chapter 3) in light of this novel space which seems to be a straightforward extension.

Label-driven VAE extension for hyperbolic space We pursued this direction in (Hamzaoui et al., 2021), where we attempted to adapt the label-driven variational auto-encoder to the hyperbolic space. Unfortunately, this approach did not yield the desired results, and the Euclidean variant outperformed it. Nonetheless, given the findings of Chapter 4 regarding practical techniques to be employed in order to benefit from the efficiency of hyperbolic space with respect to remote sensing images, it is worth revisiting

this direction. Since our previous attempt yielded poor performance, it is worth considering a more appropriate network architecture, better suited to our images, to further highlight the relevance of hyperbolic space in the remote sensing community.

Hierarchical ProtoNet extension for hyperbolic space Following the same reasoning, an extension of hierarchical prototypes to hyperbolic space seems to be an intriguing pursuit. However, an exploratory search for hyper-parameters that accentuate the hierarchical loss in the hyperbolic space and allow reaching the optimal performance should be carried out.

Hierarchical C-ProtoNet

The experiments conducted in Section 4.3 revealed that a prototypical network utilising cosine similarity as a distance measure produced results that were at least as good, if not better, than those obtained by a prototypical network employing Euclidean distance as a similarity function under standard conditions (where the latent dimension is 512). Hence, it is reasonable to contemplate the usage of a spherical embedding space for computing hierarchical prototypes and subsequently employing the cosine metric as a similarity function.

5.2.2 A step further

Leveraging class hierarchy via loss function in hyperbolic space

Our immediate plan is to extend our hierarchical approaches from Chapter 3 straight to hyperbolic space, followed by an exploration of the set of hyper-parameters to identify the ones that favour the hyperbolic space. However, it is important to note that the losses we intend to adjust were originally designed for Euclidean space, and their generalisation to hyperbolic space is achieved solely through modifications to the similarity function. Hence, it remains to be seen whether these adjusted losses will be as effective in hyperbolic space as they were in Euclidean space. As such, there is a compelling need to formulate a specialised loss function that takes into consideration not only the hierarchical information between classes but also the geometric properties specific to hyperbolic space. This would be highly advantageous in optimising performance for tasks involving hyperbolic space.

Hyperbolic space for hierarchical information within images

This thesis has investigated the relevance of hierarchical information between scene images in a classification context. Nonetheless, there is another type of hierarchical information that exists within images which describes the relationships between different pixels or objects in the image. The classes of these objects are often defined at multiple spatial-scales, and they have certain hierarchical relations between them. For example, at a finer level of detail, a building and its surrounding vegetation may be grouped together to form a residential area at an intermediary level based on their arrangement and the presence of certain types of vegetation or land cover. At a coarser level, larger urban areas can be defined based on overall characteristics such as population density, building density, and land use.

Although this thesis has concentrated on hierarchical information among images, it is important to consider the hierarchical relationships among objects within images. This can offer significant insights for numerous applications, including urban planning and environmental monitoring. As a result, comprehending and utilising hierarchical information at various scales can enhance the precision and detail of image analysis. This methodology can also be applied to segment images into significant regions and to identify objects of interest at different levels of detail.

Although hierarchical knowledge can be explicitly defined through multi-scale semantic labels, typically only one level of detail is provided. In either situation, hyperbolic space appears to be a compelling alternative for embedding images to better reflect the hierarchy or for discovering the underlying hierarchy.

Bibliography

Atigh, M. G., Keller-Ressel, M., & Mettes, P., Hyperbolic Busemann Learning with Ideal Prototypes, *in*: *Advances in Neural Information Processing Systems (NeurIPS)*, 2021, 103–115.

Barz, B., & Denzler, J., Hierarchy-Based Image Embeddings for Semantic Image Retrieval, *in*: *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2019, 638–647.

Bazi, Y., Rahhal, M. M. A., Al-Hichri, H., & Alajlan, N., (2019), Simple Yet Effective Fine-Tuning of Deep CNNs Using an Auxiliary Classification Loss for Remote Sensing Scene Classification, *Remote Sensing*, *11*, 2908.

Berg, P., Pham, M., & Courty, N., (2022), Self-Supervised Learning for Scene Classification in Remote Sensing: Current State of the Art and Perspectives, *Remote Sensing*, *14*, 3995.

Bertinetto, L., Müller, R., Tertikas, K., Samangooei, S., & Lord, N. A., Making Better Mistakes: Leveraging Class Hierarchies With Deep Networks, *in*: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, 13–19.

Bonnabel, S., (2013), Stochastic Gradient Descent on Riemannian Manifolds, *IEEE Transactions on Automatic Control*, *58*, 2217–2229.

Bossard, M, Feranec, J., Otahel, J, et al., (2000), *CORINE land cover technical guide: Addendum 2000* (Vol. 40).

Calbo, J., & Sabburg, J., (2008), Feature extraction from whole-sky ground-based images for cloud-type recognition, *Journal of Atmospheric and Oceanic Technology*, *25*, 3–14.

Castelluccio, M., Poggi, G., Sansone, C., & Verdoliva, L., (2015), Land Use Classification in Remote Sensing Images by Convolutional Neural Networks, *arXiv*, *arXiv:1508.00092*.

Castillo-Navarro, J., Saux, B. L., Boulch, A., Audebert, N., & Lefèvre, S., (2022), Semi-Supervised Semantic Segmentation in Earth Observation: The MiniFrance suite, dataset analysis and multi-task network study, *Machine Learning*, *111*, 3125–3160.

Chaib, S., Gu, Y., & Yao, H., (2016), An Informative Feature Selection Method Based on Sparse PCA for VHR Scene Classification, *IEEE Geoscience and Remote Sensing Letters*, *13*, 147–151.

Chami, I., Ying, Z., Ré, C., & Leskovec, J., Hyperbolic Graph Convolutional Neural Networks, *in*: *Advances in Neural Information Processing Systems (NeurIPS)*, 2019, 4869–4880.

Chen, B., Peng, W., Cao, X., & Röning, J., (2022a), Hyperbolic Uncertainty Aware Semantic Segmentation, *arXiv preprint arXiv:2203.08881*.

Chen, G., Zhang, X., Tan, X., Cheng, Y., Dai, F., Zhu, K., Gong, Y., & Wang, Q., (2018), Training Small Networks for Scene Classification of Remote Sensing Images via Knowledge Distillation, *Remote Sensing*, *10*, 719.

Chen, J., & Qian, Y., (2022), Hierarchical Multilabel Ship Classification in Remote Sensing Images Using Label Relation Graphs, *IEEE Transactions on Geoscience and Remote Sensing*, *60*, 1–13.

Chen, W., Han, X., Lin, Y., Zhao, H., Liu, Z., Li, P., Sun, M., & Zhou, J., Fully Hyperbolic Neural Networks, *in*: *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2022, 5672–5686.

Chen, Z., Fu, Y., Zhang, Y., Jiang, Y., Xue, X., & Sigal, L., (2019), Multi-Level Semantic Feature Augmentation for One-Shot Learning, *IEEE Transactions on Image Processing*, *28*, 4594–4605.

Cheng, G., Cai, L., Lang, C., Yao, X., Chen, J., Guo, L., & Han, J., (2022), SPNet: Siamese-Prototype Network for Few-Shot Remote Sensing Image Scene Classification, *IEEE Transactions on Geoscience and Remote Sensing*, *60*, 1–11.

Cheng, G., Han, J., & Lu, X., (2017), Remote Sensing Image Scene Classification: Benchmark and State of the Art, *Proceedings of the IEEE*, *105*, 1865–1883.

Cheng, G., Xie, X., Han, J., Guo, L., & Xia, G., (2020), Remote Sensing Image Scene Classification Meets Deep Learning: Challenges, Methods, Benchmarks, and Opportunities, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *13*, 3735–3756.

Cheriyadat, A. M., (2014), Unsupervised Feature Learning for Aerial Scene Classification, *IEEE Transactions on Geoscience and Remote Sensing*, *52*, 439–451.

Cho, H., Demeo, B., Peng, J., & Berger, B., Large-Margin Classification in Hyperbolic Space, *in*: *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2019, 1832–1840.

Cui, Y., Chapel, L., & Lefèvre, S., A Subpath Kernel for Learning Hierarchical Image Representations, *in*: *Graph-Based Representations in Pattern Recognition (GbRPR) International Workshop*, 2015, 34–43.

Dai, S., Gan, Z., Cheng, Y., Tao, C., Carin, L., & Liu, J., APo-VAE: Text generation in hyperbolic space, *in*: *North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, 2021, 416–431.

Dalal, N., & Triggs, B., Histograms of Oriented Gradients for Human Detection, *in*: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005, 886–893.

Davies, C. E., Moss, D., & Hill, M. O., (2004), EUNIS habitat classification revised 2004, *European environment agency-European topic centre on nature protection and biodiversity*, 127–143.

Deng, J., Dong, W., Socher, R., Li, L., Li, K., & Fei-Fei, L., ImageNet: A large-scale hierarchical image database, *in*: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009, 248–255.

Deng, P., Xu, K., & Huang, H., (2022), When CNNs Meet Vision Transformer: A Joint Framework for Remote Sensing Scene Classification, *IEEE Geoscience and Remote Sensing Letters*, *19*, 1–5.

Dhall, A., Makarova, A., Ganea, O., Pavllo, D., Greeff, M., & Krause, A., Hierarchical Image Classification using Entailment Cone Embeddings, *in*: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) workshops*, 2020, 3649–3658.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., & Houlsby, N., An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, *in*: *International Conference on Learning Representations (ICLR)*, 2021.

Dutta, S., & Das, M., (2023), Remote sensing scene classification under scarcity of labelled samples - A survey of the state-of-the-arts, *Computers and Geosciences*, *171*, 105295.

Ermolov, A., Mirvakhabova, L., Khrulkov, V., Sebe, N., & Oseledets, I. V., Hyperbolic Vision Transformers: Combining Improvements in Metric Learning, *in*: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, 7399–7409.

Fréchet, M., Les éléments aléatoires de nature quelconque dans un espace distancié, *in*: *Annales de l'institut Henri Poincaré*, *10*, 1948, 215–310.

Frome, A., Corrado, G. S., Shlens, J., Bengio, S., Dean, J., Ranzato, M., & Mikolov, T., DeViSE: A Deep Visual-Semantic Embedding Model, *in*: *Advances in Neural Information Processing Systems (NIPS)*, 2013, 2121–2129.

Ganea, O., Bécigneul, G., & Hofmann, T., Hyperbolic Neural Networks, *in*: *Advances in Neural Information Processing Systems (NeurIPS)*, 2018, 5350–5360.

Garg, A., Bagga, S., Singh, Y., & Anand, S., HierMatch: Leveraging Label Hierarchies for Improving Semi-Supervised Learning, *in*: *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2022, 2061–2070.

Ge, S., Mishra, S., Kornblith, S., Li, C., & Jacobs, D., (2022), Hyperbolic Contrastive Learning for Visual Representations beyond Objects, *arXiv preprint arXiv:2212.00653*.

Ge, W., Huang, W., Dong, D., & Scott, M. R., Deep Metric Learning with Hierarchical Triplet Loss, *in*: *European Conference on Computer Vision (ECCV)*, 2018, 272–288.

Gerhards, M., Schlerf, M., Mallick, K., & Udelhoven, T., (2019), Challenges and Future Perspectives of Multi-/Hyperspectral Thermal Infrared Remote Sensing for Crop Water-Stress Detection: A Review, *Remote Sensing*, *11*, 1240.

Goel, A., Banerjee, B., & Pizurica, A., (2019), Hierarchical Metric Learning for Optical Remote Sensing Scene Categorization, *IEEE Geoscience and Remote Sensing Letters*, *16*, 952–956.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. C., & Bengio, Y., (2020), Generative adversarial networks, *Communications of the ACM*, *63*, 139–144.

Goyal, P., Choudhary, D., & Ghosh, S., Hierarchical Class-Based Curriculum Loss, *in*: *International Joint Conference on Artificial Intelligence (IJCAI)*, 2021, 2448–2454.

Gromov, M., (1987), *Hyperbolic groups*.

Guo, W., Li, S., Yang, J., Zhou, Z., Liu, Y., Lu, J., Kou, L., & Zhao, M., (2022a), Remote Sensing Image Scene Classification by Multiple Granularity Semantic Learning, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *15*, 2546–2562.

Guo, Y., Wang, X., Chen, Y., & Yu, S. X., Clipped Hyperbolic Classifiers Are Super-Hyperbolic Classifiers, *in*: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, 1–10.

Hamzaoui, M., Chapel, L., Pham, M., & Lefèvre, S., A Hierarchical Prototypical Network for Few-Shot Remote Sensing Scene Classification, *in*: *International Conference on Pattern Recognition and Artificial Intelligence (ICPRAI)*, *13364*, 2022, 208–220.

Hamzaoui, M., Chapel, L., Pham, M.-T., & Lefèvre, S., Hyperbolic Variational Auto-Encoder for Remote Sensing Scene Classification, *in*: *Journées Francophones des Jeunes Chercheurs en Vision par Ordinateur (ORASIS)*, 2021.

Han, W., Wang, L., Feng, R., Gao, L., Chen, X., Deng, Z., Chen, J., & Liu, P., (2020), Sample generation based on a supervised Wasserstein Generative Adversarial Network for high-resolution remote-sensing scene classification, *Information Sciences*, *539*, 177–194.

He, K., Zhang, X., Ren, S., & Sun, J., Deep Residual Learning for Image Recognition, *in*: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, 770–778.

Hsu, J., Gu, J., Wu, G. H., Chiu, W., & Yeung, S., Capturing implicit hierarchical structure in 3D biomedical images with self-supervised hyperbolic representations, *in*: *Advances in Neural Information Processing Systems (NeurIPS)*, 2021, 5112–5123.

Huang, G., Liu, Z., van der Maaten, L., & Weinberger, K. Q., Densely Connected Convolutional Networks, *in*: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, 2261–2269.

Iandola, F. N., Moskewicz, M. W., Ashraf, K., Han, S., Dally, W. J., & Keutzer, K., (2016), SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <1MB model size, *arXiv preprint arXiv:1602.07360*.

Khrulkov, V., Mirvakhabova, L., Ustinova, E., Oseledets, I. V., & Lempitsky, V. S., Hyperbolic Image Embeddings, *in*: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, 6417–6427.

Kingma, D. P., & Ba, J., Adam: A Method for Stochastic Optimization, *in*: *International Conference on Learning Representations (ICLR)*, 2015.

Kingma, D. P., & Welling, M., Auto-Encoding Variational Bayes, *in*: *International Conference on Learning Representations (ICLR)*, 2014.

Kochurov, M., Karimov, R., & Kozlukov, S., (2020), Geoopt: Riemannian Optimization in PyTorch, *arXiv preprint arXiv:2005.02819*.

Kolisnik, B., Hogan, I., & Zulkernine, F. H., (2021), Condition-CNN: A hierarchical multi-label fashion image classification model, *Expert Systems with Applications*, *182*, 115195.

Krizhevsky, A., & Hinton, G., (2009), *Learning multiple layers of features from tiny images* (tech. rep.), University of Toronto.

Krizhevsky, A., Sutskever, I., & Hinton, G. E., ImageNet Classification with Deep Convolutional Neural Networks, *in*: *Advances in Neural Information Processing Systems (NIPS)*, 2012, 1106–1114.

Lang, C., Braun, A., Schillingmann, L., & Valada, A., On Hyperbolic Embeddings in Object Detection, *in*: *GCPR*, 2022, 462–476.

Lei, Z., Li, H., Zhao, J., Jing, L., Tang, Y., & Wang, H., (2022), Individual Tree Species Classification Based on a Hierarchical Convolutional Neural Network and Multitemporal Google Earth Images, *Remote Sensing*, *14*, 5124.

Li, A., Luo, T., Lu, Z., Xiang, T., & Wang, L., Large-Scale Few-Shot Learning: Knowledge Transfer With Class Hierarchy, *in*: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, 7212–7220.

Li, L., Han, J., Yao, X., Cheng, G., & Guo, L., (2021a), DLA-MatchNet for Few-Shot Remote Sensing Image Scene Classification, *IEEE Transactions on Geoscience and Remote Sensing*, *59*, 7844–7853.

Li, L., Zhou, T., Wang, W., Li, J., & Yang, Y., Deep Hierarchical Semantic Segmentation, *in*: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, 1236–1247.

Li, X., Xu, F., Liu, F., Xia, R., Tong, Y., Li, L., Xu, Z., & Lyu, X., Hybridizing Euclidean and Hyperbolic Similarities for Attentively Refining Representations in Semantic Segmentation of Remote Sensing Images, *in*: *19*, 2022, 1–5.

Li, X., Du, Z., Huang, Y., & Tan, Z., (2021b), A deep translation (GAN) based change detection network for optical and sar remote sensing images, *ISPRS J. Photogramm. Remote Sens.*, *179*, 14–34.

Li, X., Li, H., Yu, R., & Wang, F., Few-shot scene classification with attention mechanism in remote sensing, *in*: *Journal of physics: conference series*, *1961*, 2021, 012015.

Liu, L., Zhou, T., Long, G., Jiang, J., & Zhang, C., (2022), Many-Class Few-Shot Learning on Multi-Granularity Class Hierarchy, *IEEE Trans. Knowl. Data Eng.*, *34*, 2293–2305.

Liu, S., Chen, J., Pan, L., Ngo, C., Chua, T., & Jiang, Y., Hyperbolic Visual Embedding Learning for Zero-Shot Recognition, *in*: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, 9270–9278.

Liu, X., Zhou, Y., & Zhao, H., (2021), Robust hierarchical feature selection driven by data and knowledge, *Information Sciences*, *551*, 341–357.

Liu, Y., Liu, Y., Chen, C., & Ding, L., (2020b), Remote-sensing image retrieval with tree-triplet-classification networks, *Neurocomputing*, *405*, 48–61.

Liu, Y., Suen, C. Y., Liu, Y., & Ding, L., (2019), Scene Classification Using Hierarchical Wasserstein CNN, *IEEE Transactions on Geoscience and Remote Sensing*, *57*, 2494–2509.

Long, Y., Xia, G., Li, S., Yang, W., Yang, M. Y., Zhu, X. X., Zhang, L., & Li, D., (2021), On Creating Benchmark Dataset for Aerial Image Interpretation: Reviews, Guidances, and Million-AID, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *14*, 4205–4230.

Lowe, D. G., (2004), Distinctive Image Features from Scale-Invariant Keypoints, *International Journal of Computer Vision*, *60*, 91–110.

Lv, P., Wu, W., Zhong, Y., Du, F., & Zhang, L., (2022), SCViT: A Spatial-Channel Feature Preserving Vision Transformer for Remote Sensing Image Scene Classification, *IEEE Transactions on Geoscience and Remote Sensing*, *60*, 1–12.

Marmanis, D., Datcu, M., Esch, T., & Stilla, U., (2016), Deep Learning Earth Observation Classification Using ImageNet Pretrained Networks, *IEEE Geoscience and Remote Sensing Letters*, *13*, 105–109.

Marszalek, M., & Schmid, C., Semantic Hierarchies for Visual Object Recognition, *in*: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007, 1–7.

Mathieu, E., Lan, C. L., Maddison, C. J., Tomioka, R., & Teh, Y. W., Continuous Hierarchical Representations with Poincaré Variational Auto-Encoders, *in*: *Advances in Neural Information Processing Systems (NeurIPS)*, 2019, 12544–12555.

Mayouf, M., & de Saint-Cyr, F. D., GH-CNN: A new CNN for coherent hierarchical classification, *in*: *International Conference on Artificial Neural Networks (ICANN)*, *13532*, 2022, 669–681.

Miao, W., Geng, J., & Jiang, W., (2022), Semi-Supervised Remote-Sensing Image Scene Classification Using Representation Consistency Siamese Network, *IEEE Transactions on Geoscience and Remote Sensing*, *60*, 1–14.

Miller, G. A., (1998), *WordNet: An electronic lexical database*.

Nagano, Y., Yamaguchi, S., Fujita, Y., & Koyama, M., A Wrapped Normal Distribution on Hyperbolic Space for Gradient-Based Learning, *in*: *International Conference on Machine Learning (ICML)*, 2019, 4693–4702.

Nickel, M., & Kiela, D., Learning Continuous Hierarchies in the Lorentz Model of Hyperbolic Geometry, *in*: *International Conference on Machine Learning (ICML)*, 2018, 3776–3785.

Nickel, M., & Kiela, D., Poincaré Embeddings for Learning Hierarchical Representations, *in*: *Advances in Neural Information Processing Systems (NIPS)*, 2017, 6338–6347.

Odaibo, S. G., (2019), Tutorial: Deriving the Standard Variational Autoencoder (VAE) loss function, *arXiv preprint arXiv:1907.08956*.

Ojala, T., Pietikäinen, M., & Mäenpää, T., (2002), Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *24*, 971–987.

Peng, W., Varanka, T., Mostafa, A., Shi, H., & Zhao, G., (2022), Hyperbolic Deep Neural Networks: A Survey, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *44*, 10023–10044.

Ramzi, E., Audebert, N., Thome, N., Rambour, C., & Bitot, X., Hierarchical Average Precision Training for Pertinent Image Retrieval, *in*: *European Conference on Computer Vision (ECCV)*, 2022, 250–266.

Ravi, S., & Larochelle, H., Optimization as a Model for Few-Shot Learning, *in*: *International Conference on Learning Representations (ICLR)*, 2017.

Sakai, H., & Iiduka, H., (2022), Riemannian Adaptive Optimization Algorithm and its Application to Natural Language Processing, *IEEE Transactions on Cybernetics*, *52*, 7328–7339.

Schumann, G. J., Brakenridge, G. R., Kettner, A. J., Kashif, R., & Niebuhr, E., (2018), Assisting Flood Disaster Response with Earth Observation Data and Products: A Critical Assessment, *Remote Sensing*, *10*, 1230.

Sen, O., & Keles, H. Y., (2022a), *PFG-Journal of Photogrammetry, Remote Sensing and Geoinformation Science*, *90*, 161–175.

Sen, O., & Keles, H. Y., (2022b), A Hierarchical Approach to Remote Sensing Scene Classification, *PFG–Journal of Photogrammetry, Remote Sensing and Geoinformation Science*, *90*, 161–175.

Seo, Y., & Shin, K., (2019), Hierarchical convolutional neural networks for fashion image classification, *Expert Systems with Applications*, *116*, 328–339.

Sha, Z., & Li, J., (2022), MITformer: A Multiinstance Vision Transformer for Remote Sensing Scene Classification, *IEEE Geoscience and Remote Sensing Letters*, *19*, 1–5.

Shahbazi, M., Théau, J., & Ménard, P., (2014), Recent applications of unmanned aerial imagery in natural resource management, *GIScience & Remote Sensing*, *51*, 339–365.

Shi, X., Salewski, L., Schiegg, M., & Welling, M., Relational Generalized Few-Shot Learning, *in*: *British machine vision conference (BMVC)*, 2020.

Shimizu, R., Mukuta, Y., & Harada, T., Hyperbolic Neural Networks++, *in*: *International Conference on Learning Representations (ICLR)*, 2021.

Shin, S., Kim, S., Kim, Y., & Kim, S., (2020), Hierarchical Multi-Label Object Detection Framework for Remote Sensing Images, *Remote Sensing*, *12*, 2734.

Simonyan, K., & Zisserman, A., Very Deep Convolutional Networks for Large-Scale Image Recognition, *in*: *International Conference on Learning Representations (ICLR)*, 2015.

Snell, J., Swersky, K., & Zemel, R. S., Prototypical Networks for Few-shot Learning, *in*: *Advances in Neural Information Processing Systems (NIPS)*, 2017, 4077–4087.

Sun, H., Zhang, L., Ren, J., & Huang, H., (2022), Novel hyperbolic clustering-based band hierarchy (HCBH) for effective unsupervised band selection of hyperspectral images, *Pattern Recognition*, *130*, 108788.

Sun, X., Wang, B., Wang, Z., Li, H., Li, H., & Fu, K., (2021), Research Progress on Few-Shot Learning for Remote Sensing Image Interpretation, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *14*, 2387–2402.

Sung, F., Yang, Y., Zhang, L., Xiang, T., Torr, P. H. S., & Hospedales, T. M., Learning to Compare: Relation Network for Few-Shot Learning, *in*: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, 1199–1208.

Swain, M. J., & Ballard, D. H., (1991), Color indexing, *International Journal of Computer Vision*, *7*, 11–32.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S. E., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A., Going deeper with convolutions, *in*: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, 1–9.

Tang, X., Ma, Q., Zhang, X., Liu, F., Ma, J., & Jiao, L., (2021), Attention Consistent Network for Remote Sensing Scene Classification, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *14*, 2030–2045.

Taoufiq, S., Nagy, B., & Benedek, C., (2020), HierarchyNet: Hierarchical CNN-Based Urban Building Classification, *Remote Sensing*, *12*, 3794.

Tifrea, A., Bécigneul, G., & Ganea, O., Poincare Glove: Hyperbolic Word Embeddings, *in*: *International Conference on Learning Representations (ICLR)*, 2019.

Van Westen, C. J., Castellanos, E., & Kuriakose, S. L., (2008), Spatial data for landslide susceptibility, hazard, and vulnerability assessment: An overview, *Engineering geology*, *102*, 112–131.

Vinyals, O., Blundell, C., Lillicrap, T., Kavukcuoglu, K., & Wierstra, D., Matching Networks for One Shot Learning, *in*: *Advances in Neural Information Processing Systems (NIPS)*, 2016, 3630–3638.

Wah, C., Branson, S., Welinder, P., Perona, P., & Belongie, S., (2011), The Caltech-UCSD Birds-200-2011 Dataset.

Wang, D., Zhang, Q., Xu, Y., Zhang, J., Du, B., Tao, D., & Zhang, L., (2022a), Advancing Plain Vision Transformer Towards Remote Sensing Foundation Model, *IEEE Transactions on Geoscience and Remote Sensing*.

Wang, M., Zhang, X., Niu, X., Wang, F., & Zhang, X., (2019), Scene classification of high-resolution remotely sensed image based on resnet, *Journal of Geovisualization and Spatial Analysis*, *3*, 1–9.

Wang, S., Chen, X., Wang, Y., Long, M., & Wang, J., Progressive Adversarial Networks for Fine-Grained Domain Adaptation, *in*: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, 9210–9219.

Wang, W., Chen, Y., & Ghamisi, P., (2022b), Transferring CNN With Adaptive Learning for Remote Sensing Scene Classification, *IEEE Transactions on Geoscience and Remote Sensing*, *60*, 1–18.

Wu, B., Xu, C., Dai, X., Wan, A., Zhang, P., Tomizuka, M., Keutzer, K., & Vajda, P., (2020), Visual Transformers: Token-based Image Representation and Processing for Computer Vision, *arXiv*, *arXiv:2006.03677*.

Xia, G., Hu, J., Hu, F., Shi, B., Bai, X., Zhong, Y., Zhang, L., & Lu, X., (2017), AID: A Benchmark Data Set for Performance Evaluation of Aerial Scene Classification, *IEEE Transactions on Geoscience and Remote Sensing*, *55*, 3965–3981.

Xia, G.-S., Yang, W., Delon, J., Gousseau, Y., Sun, H., & Maître, H., (2010), Structural High-resolution Satellite Image Indexing, *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, *38*.

Xu, J., & Du, Q., (2020), Learning neural networks for text classification by exploiting label relations, *Multimedia Tools and Applications*, *79*, 22551–22567.

Yang, F., Wang, R., & Chen, X., (2022a), SEGA: Semantic Guided Attention on Visual Prototype for Few-Shot Learning, 1586–1596.

Yang, M., Zhou, M., Li, Z., Liu, J., Pan, L., Xiong, H., & King, I., (2022b), Hyperbolic Graph Neural Networks: A Review of Methods and Applications, *arXiv*, *arXiv:2202.13852*.

Yang, Y., & Newsam, S. D., Bag-of-visual-words and spatial extensions for land-use classification, *in*: *ACM SIGSPATIAL International Symposium on Advances in Geographic Information Systems (ACM-GIS)*, 2010, 270–279.

Yang, Z., Bastan, M., Zhu, X., Gray, D., & Samaras, D., Hierarchical Proxy-based Loss for Deep Metric Learning, *in*: *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2022, 449–458.

Yu, K., Visweswaran, S., & Batmanghelich, K., (2020), Semi-supervised Hierarchical Drug Embedding in Hyperbolic Space, *Journal of chemical information and modeling*, *60*, 5647–5657.

Yuan, Y., Fang, J., Lu, X., & Feng, Y., (2019), Remote Sensing Image Scene Classification Using Rearranged Local Features, *IEEE Transactions on Geoscience and Remote Sensing*, *57*, 1779–1792.

Zeng, P., Lin, S., Sun, H., & Zhou, D., (2022), Exploiting Hierarchical Label Information in an Attention-Embedding, Multi-Task, Multi-Grained, Network for Scene Classification of Remote Sensing Imagery, *Applied Sciences*, *12*, 8705.

Zhang, B., Jiang, H., Feng, S., Li, X., Ye, Y., & Ye, R., Hyperbolic Knowledge Transfer with Class Hierarchy for Few-Shot Learning, *in*: *International Joint Conference on Artificial Intelligence (IJCAI)*, 2022, 3723–3729.

Zhang, F., Du, B., & Zhang, L., (2015), Saliency-Guided Unsupervised Feature Learning for Scene Classification, *IEEE Transactions on Geoscience and Remote Sensing*, *53*, 2175–2184.

Zhang, P., Bai, Y., Wang, D., Bai, B., & Li, Y., (2021a), Few-Shot Classification of Aerial Scene Images via Meta-Learning, *Remote Sensing*, *13*, 108.

Zhang, P., Fan, G., Wu, C., Wang, D., & Li, Y., (2021b), Task-Adaptive Embedding Learning with Dynamic Kernel Fusion for Few-Shot Remote Sensing Scene Classification, *Remote Sensing*, *13*, 4200.

Zhang, P., Li, Y., Wang, D., & Wang, J., (2021c), RS-SSKD: Self-Supervision Equipped with Knowledge Distillation for Few-Shot Remote Sensing Scene Classification, *Sensors*, *21*, 1566.

Zhao, L., Tang, P., & Huo, L., (2014), A 2-D wavelet decomposition-based bag-of-visual-words model for land-use scene classification, *International Journal of Remote Sensing*, *35*, 2296–2310.

Zhao, Z., Luo, Z., Li, J., Chen, C., & Piao, Y., (2020), When Self-Supervised Learning Meets Scene Classification: Remote Sensing Scene Classification Based on a Multitask Learning Framework, *Remote Sensing*, *12*, 3276.

Zhou, W., Newsam, S. D., Li, C., & Shao, Z., (2018), PatternNet: A Benchmark Dataset for Performance Evaluation of Remote Sensing Image Retrieval, *ISPRS journal of photogrammetry and remote sensing*, *145*, 197–209.

Zhu, X., & Bain, M., (2017), B-CNN: Branch Convolutional Neural Network for Hierarchical Classification, *arXiv preprint arXiv:1709.09890*.

Zhu, Y., Zhou, D., Xiao, J., Jiang, X., Chen, X., & Liu, Q., HyperText: Endowing FastText with Hyperbolic Geometry, *in*: *Findings of the Association for Computational Linguistics (EMNLP)*, 2020, 1166–1171.

COLLEGE MATHSTIC
DOCTORAL BRETAGNE
BRETAGNE OCEANE

Université Bretagne Sud
ubs

Titre : De l'espace euclidien à l'espace hyperbolique : Repenser la classification hiérarchique des images de scènes de télédétection

Mot clés : classification de scènes, hiérarchie de classes, espace hyperbolique

Résumé : Les images de télédétection sont complexes et présentent généralement une structure hiérarchique qui est souvent négligée, en particulier par les méthodes de classification de scènes. Ces dernières ont tendance à traiter toutes les non cibles classes de manière égale, ce qui peut conduire à des erreurs importantes lorsqu'il y a confusion entre des classes non liées sémantiquement. En introduisant l'information hiérarchique dans leur apprentissage, ces approches peuvent être rendues plus cohérentes. Cette information est souvent disponible de manière explicite via la hiérarchie de classes ou implicitement dans les données. Cette thèse se concentre donc sur la classification de scènes à l'aide de l'information de la hiérarchie.

D'abord, nous introduisons la hiérarchie de classes dans l'apprentissage d'un classifieur via une fonction de perte hiérarchique. Nous évaluons son impact dans un cadre avec peu d'exemples (few-shot) avec des prototypes hiérarchiques définis à chaque niveau de la hiérarchie de classes. Les résultats des expérimentations montrent que la hiérarchie de classes est une source d'information prometteuse pour améliorer les performances du classifieur. Ensuite, nous utilisons l'espace hyperbolique comme espace d'analyse car il est mieux adapté au traitement des données présentant une hiérarchie sous-jacente. Nous évaluons cette approche dans deux cadres : non supervisé et few-shot. Les résultats des expérimentations mettent en évidence le potentiel de l'espace hyperbolique pour la classification de scènes, ce qui en fait une approche prometteuse pour la communauté de la télédétection.

Title: From Euclidean to Hyperbolic Space: Rethinking Hierarchical Classification of Remote Sensing Scene Images

Keywords: scene classification, class hierarchy, hyperbolic space

Abstract: Remote sensing images are complex and typically exhibit a hierarchical structure which is often overlooked, particularly in scene classification methods. These methods tend to treat all non-target classes with equal importance, which can lead to severe mistakes when confusion between semantically unrelated classes. By introducing hierarchical information into the learning process, these approaches can provide more coherent predictions. This hierarchical information is often available explicitly via the class hierarchy or implicitly within the data. This thesis therefore focuses on scene classification with hierarchical information.

Firstly, we introduce the class hierarchy when training a classifier via a hierarchical loss function. We evaluate its impact in a few-shot setting with hierarchical prototypes defined at each level of the class hierarchy. Experimental results reveal that the class hierarchy is a promising source of information to im-

prove the scene classifier performance. Subsequently, we consider the hyperbolic space as an embedding space as it is better suited to handle data with an underlying hierarchy. We evaluate the approach within two settings: unsupervised and few-shot. The experimental results highlight the potential of the hyperbolic space for scene classification, making it a promising approach for the remote sensing community.