This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Hyperbolic Defect Feature Synthesis for Few-Shot Defect Classification

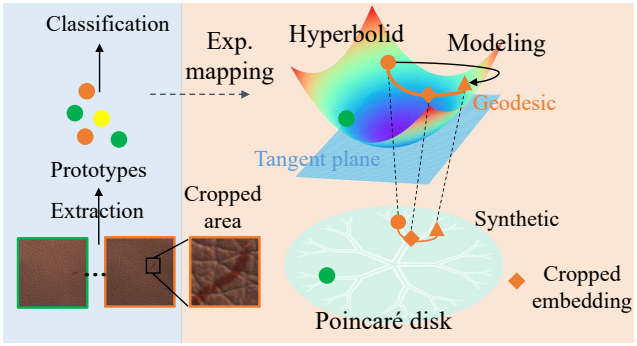
Huimin Li[1] Boxuan Hu[1] Yulin Zhang[1,2] Xiuzhuang Zhou[3,4] Junlin Hu[1,2*]
[1]School of Software, Beihang University [2]Qingdao Research Institute, Beihang University
[3]Beijing University of Posts and Telecommunications [4]Beijing Ketai Industrial Intelligence (BKII)
{lihuimin, huboxuan, zhangyulin, hujunlin}@buaa.edu.cn xiuzhuang.zhou@bupt.edu.cn

Abstract

Defect synthesis, as a core technology for addressing the problem of few-shot defect classification, has been widely adopted in industrial scenarios. It helps alleviate the problem of insufficient model generalization capability owing to data scarcity by establishing a data augmentation pipeline. Recently, remarkable progress has been achieved in both explicit defect image generation and implicit defect feature synthesis approaches. However, existing methods are always conducted in Euclidean space. Constrained by the flatness of Euclidean space, it is difficult to synthesize defect data containing complex structures. In this paper, we attempt to explore defect generation in hyperbolic space and propose a hyperbolic defect feature synthesis (HypDFS) method. By modeling the potential defect distribution via a small number of hyperbolic defect prototypes and further optimizing the synthetic defect features with the hierarchical defect contrastive loss in hyperbolic space, our HypDFS method can obtain a better generalized defect representation that is more conducive to downstream few-shot defect classification task. Extensive experiments on the MVTec-FS benchmark and standard MTD dataset under the few-shot settings demonstrate that the proposed HypDFS surpasses the Euclidean baseline by a large margin, showing promising prospects for defect synthesis in hyperbolic space.

1. Introduction

Industrial production is undergoing profound changes due to the increasingly diverse and refined demands, gradually accelerating from large-scale, single variety production to small-scale, multi variety production [5, 48]. This poses a huge challenge to defect classification. Traditional methods [2, 16, 19, 23, 37, 55, 58] generally require a large amount of samples for training. Their performance significantly decreases when dealing with small samples, making it difficult to accurately identify defects in products. Few-shot defect classification has therefore emerged as a key area of research in the current intelligent industrial field, attracting widespread attention from both academic and industrial communities.



(a) Existing methods (b) Our HypDFS method

Figure 1. An illustration of the differences between (a) the existing Euclidean space-based methods and (b) our HypDFS method. Our HypDFS learns the tree-like semantic hierarchy of the defective images with additional synthetic features in hyperbolic space.

Numerous studies [8, 25, 29, 39, 42, 45, 49] have been conducted on the few-shot defect classification problem, among which the synthesis-based strategy is one of the most promising routes. Synthesis-based methods [14, 17, 22, 25, 54] can be roughly divided into three categories: handcrafted rule-based augmentation, AIGC-based generation, and synthetic feature-based augmentation. The handcrafted rule-based strategy [22, 29, 51] may involve generating new samples and expanding the training data using operations such as rotation, flipping, scaling, and noise addition. However, this type of augmentation requires in-depth domain knowledge, and incorrect application may corrupt the semantic information of the original data. Although current AIGC-based generation methods [8, 17, 53] have shown promising results for general image generation tasks, significant challenges remain in industrial applications, particularly for defect generation tasks. Moreover, some of such

*This work was supported by the National Natural Science Foundation of China under Grant 62476020, Grant U25A20446, and Grant 62576049. *(Corresponding author: Junlin Hu.)*

methods cannot operate in scenarios with limited computational resources. Therefore, synthetic feature-based methods strike a balance between computational resources and model performance in the current context. However, existing synthetic feature-based methods [3, 28] in Euclidean space struggle to capture complex relationships between features.

In this paper, we explore the problem of few-shot defect classification by synthesizing defect features through a small number of prototypes. We present a hyperbolic defect feature synthesis (HypDFS) method to capture the semantic hierarchy among defective images, which extends the defect feature synthesis to hyperbolic space. Unlike existing Euclidean space-based methods limited by the flatness of Euclidean space, as shown in Fig. 1, we model the potential distribution patterns of defects under scarce data conditions in hyperbolic space. The main contributions of this paper can be summarized as follows.

- We propose a hyperbolic defect feature synthesis approach that can generate a more desirable defect representation via a small number of hyperbolic defect prototypes, extending the family of defect feature synthesis techniques to the hyperbolic space.
- We train a residual adapter to dynamically fuse substantial pre-trained general features with industrial defect features. Furthermore, we propose a hyperbolic distance-based hierarchical defect contrastive loss to improve the representation of real and synthetic defects, thereby enhancing performance of few-shot defect classification.
- Extensive experiments on the MVTec-FS benchmark and standard MTD dataset under few-shot settings show that our HypDFS method outperforms the Euclidean baseline by a large margin, demonstrating the promising prospects for defect synthesis in hyperbolic space.

2. Related Work

2.1. Defect Synthesis

Defect synthesis has garnered considerable attention as a core technique to address few-shot classification in industrial scenarios [6, 44, 54]. It alleviates the issue of inadequate model generalization caused by scarce data by constructing a data augmentation framework, offering a practical solution to the few-shot classification in industrial scenarios. Handcrafted rule-based approaches [22, 29, 51] constitute the most prevalent paradigm in defect synthesis. For example, Li *et al*. [22] propose CutPaste, which synthesizes defect samples by cropping, rotating, and transforming normal images, and then classifies defects by calculating the distance of normal samples and generated defect samples. Lyu *et al*. [29] present MVREC that uses multi-scale and offsetting operations to generate enhancement defects with multiple view instances. However, the types of defect synthesized by these methods are very limited. A prominent alternative paradigm leverages deep generative architectures [17, 26, 52, 54, 57], such as generative adversarial networks and diffusion models, to learn the intrinsic patterns of defect. For example, Jin *et al*. [17] present DualAnoDiff, using a dual stream diffusion model and a dynamic defect injection mechanism to achieve highly controllable defect generation. In addition, feature level defect synthesis is also imported in defect data augmentation. For instance, Liu *et al*. [28] propose a SimpleNet, synthesizing the defect features by using pre-trained features and noise. However, in the field of industrial defect generation, existing approaches remain confined to Euclidean space, which may hinder the precise capture of defect characteristics and the intricate hierarchical relationships between different defect types. In this paper, we aim to fill this gap by exploring the defect synthesis in hyperbolic space and propose a novel hyperbolic defect feature synthesis method.

2.2. CLIP Adaptation

Although vision-language models like CLIP have attracted widespread attention due to the powerful general semantic understanding and generalization capability, directly applying pre-trained CLIP to specific downstream tasks often leads to performance bottlenecks caused by distribution discrepancies between source and target domains. Recently, large-scale model approaches incorporating fine-tuning strategies have garnered significant attention in various visual tasks [13, 21, 35, 47]. The advantage of such methods lies in their robust cross-scenario adaptability. Even under few-shot conditions with scarce training data, they show outstanding generalization performance. For example, Gao *et al*. [10] propose CLIP-Adapter to fine-tune CLIP features for better generalization performance. Lyu *et al*. [29] achieve satisfactory results on few-shot defect classification task by fine-tuning CLIP. Differing slightly from existing approaches, we fine-tune the AlphaCLIP using synthetic defect features, enabling it to perform more effectively on downstream few-shot defect classification task.

2.3. Hyperbolic Feature Embeddings

With the rapid advancement of deep learning, the demands for data representation and model construction have grown increasingly complex and diverse in recent years [31]. Euclidean space has gradually revealed its limitations when handling certain intricate tasks [1, 31, 33, 40]. The emergence of hyperbolic space as an alternative to Euclidean space expands the existing paradigm of feature representation [31, 36]. Its unique geometric properties and impressive data representation capabilities enable it to gain prominence in various fields such as computer vision [24, 27, 50] and multimodal representation [4, 30, 34]. Hyperbolic space has demonstrated enormous potential in the field of

computer vision [9, 11, 24, 27, 30]. It can handle data with complex tree-like structures more effectively [18]. Employing the geometric properties of hyperbolic space enables the construction of more robust and efficient visual models, significantly enhancing the accuracy of downstream visual tasks. For example, Li *et al*. [24] attempt to detect image anomalies in hyperbolic space. Liu *et al*. [27] present a hyperbolic framework for generalized category discovery problem. Furthermore, hyperbolic space also performs well in multimodal representation. For instance, Mandica *et al*. [30] introduce the hyperbolic space into BLIP-2 architecture and explore the scaling of hyperbolic multimodal representation. Pal *et al*. [34] present a compositional entailment learning framework to learn and align image-text representation in a shared hyperbolic space. However, for the problem of defect synthesis, there is no relevant work conducted in hyperbolic space. To this aim, we propose a hyperbolic defect feature synthesis method to model the latent distribution of defects and refine the generated features via the corresponding defective areas in hyperbolic space, making the model better comprehend the hierarchical relationships among defect data.

3. Proposed Method

This section first introduces some basic concepts of hyperbolic geometry and our motivation, and then presents our HypDFS, a defect feature synthesis method for industrial few-show classification in hyperbolic space. Fig. 2 shows the overall pipeline of the proposed HypDFS method.

3.1. Preliminary and Motivation

Hyperbolic Preliminary. Hyperbolic space, as a representative of non-Euclidean geometry, is a powerful alternative to Euclidean space due to its unique geometric properties and powerful feature representation capabilities. Here, we introduce several concepts and basic operations of hyperbolic space used in our work. The n-dimensional hyperbolic space $\mathbb{H}^n$ is a smooth Riemannian manifold with constant negative curvature, characterized by an exponential expansion of spatial volume as the radius increases. This makes it particularly well-suited for embedding data with tree-like hierarchical structures. Several isometric models [9, 18] have been constructed to establish a mapping relationship between hyperbolic and Euclidean space. Here, we choose the well-known Poincaré ball owing to its excellent performance in various vision tasks. Poincaré ball can be defined as $\mathbb{D}_c^n = \{\mathbf{x} \in \mathbb{R}^n : c||\mathbf{x}||^2 < 1\}$, where the curvature parameter $c > 0$ and the actual curvature is $-c^2$ following the settings in [9]. In addition, the tangent space at a given point $\mathbf{u} \in \mathbb{D}_c^n$ refers to the linear space comprising all tangent vectors at that point.

Unlike addition in Euclidean space, the corresponding operation in hyperbolic space should conform to manifold constraints. Given two points $\mathbf{x}, \mathbf{y} \in \mathbb{D}_c^n$, their addition operation $\mathbf{x} \oplus_c \mathbf{y}$ [9] is defined as

$$\mathbf{x} \oplus_c \mathbf{y} = \frac{(1 + 2c\langle \mathbf{x}, \mathbf{y} \rangle + c\|\mathbf{y}\|^2)\mathbf{x} + (1 - c\|\mathbf{x}\|^2)\mathbf{y}}{(1 + 2c\langle \mathbf{x}, \mathbf{y} \rangle + c^2\|\mathbf{x}\|^2\|\mathbf{y}\|^2)}. \quad (1)$$

Accordingly, the distance between two points in the hyperbolic space is defined as

$$d_{\mathbb{D}_c}(\mathbf{x}, \mathbf{y}) = \frac{2}{\sqrt{c}} \mathsf{arctanh}(\sqrt{c}\| - \mathbf{x} \oplus_c \mathbf{y}\|). \quad (2)$$

It can be shown that when c tends to zero, the hyperbolic distance $d_{\mathbb{D}_c}(\mathbf{x}, \mathbf{y})$ in Eq. (2) degenerates into the Euclidean distance $\lim_{c \to 0} d_{\mathbb{D}_c}(\mathbf{x}, \mathbf{y}) = 2\|\mathbf{x} - \mathbf{y}\|$.

Motivation. Our motivation for exploiting hyperbolic space is straightforward considering its advantages in modeling tree-like hierarchical relationships. Fig. 3 illustrates such advantages using a defect example from MVTec-FS benchmark [29]. We can see that the leather defect includes various types such as glue, fold, and color. More granularly, the color defect also includes different colors, shapes, and sizes. More quantitatively, the leather defect contains 5 subclasses of defects, with the number of instances in each category ranging from 9 to 12. When the depth of the tree is 3, the average branching factor is 9.3 for the leather defect, indicating its exponential growth characteristic, especially apparent under the influence of synthetic data.

3.2. HypDFS

As illustrated in Fig. 2, our HypDFS method mainly contains three components: hyperbolic defect prototypes, defect manifold modeling in hyperbolic space, and defect residual adapter and hierarchical defect contrastive learning. The details of each component are presented below.

3.2.1. Hyperbolic Defect Prototypes

Prototype features play a crucial role in the challenging realm of few-shot classification. Effective prototype features can accurately capture the distinctive attributes of each defect type and clearly distinguish the boundaries between different categories. To obtain generalized defect prototype features, inspired by [29], we first use handcrafted methods such as rotation and translation to augment existing defect data, while avoiding excessive deviation in the direction of defect feature generation during subsequent training to some extent. Then, we employ a pre-trained CLIP model to extract prototype features, as it demonstrates strong adaptability and scalability in defect image understanding tasks. Specifically, given the input defect image $\mathbf{I}$, we first obtain handcrafted augment data $\mathbf{I}_{\text{hd}}$ using augmentation operation $\text{AUG}(\cdot)$ such as rotation translation and scaling, and then extract the defect features $\mathbf{f}_{\text{hd}}$ by the pre-trained CLIP extractor $\text{CLIP}(\cdot)$ as

$$\mathbf{I}_{\text{hd}} = \text{AUG}(\mathbf{I}), \quad \mathbf{f}_{\text{hd}} = \text{CLIP}(\mathbf{I}_{\text{hd}}), \quad (3)$$

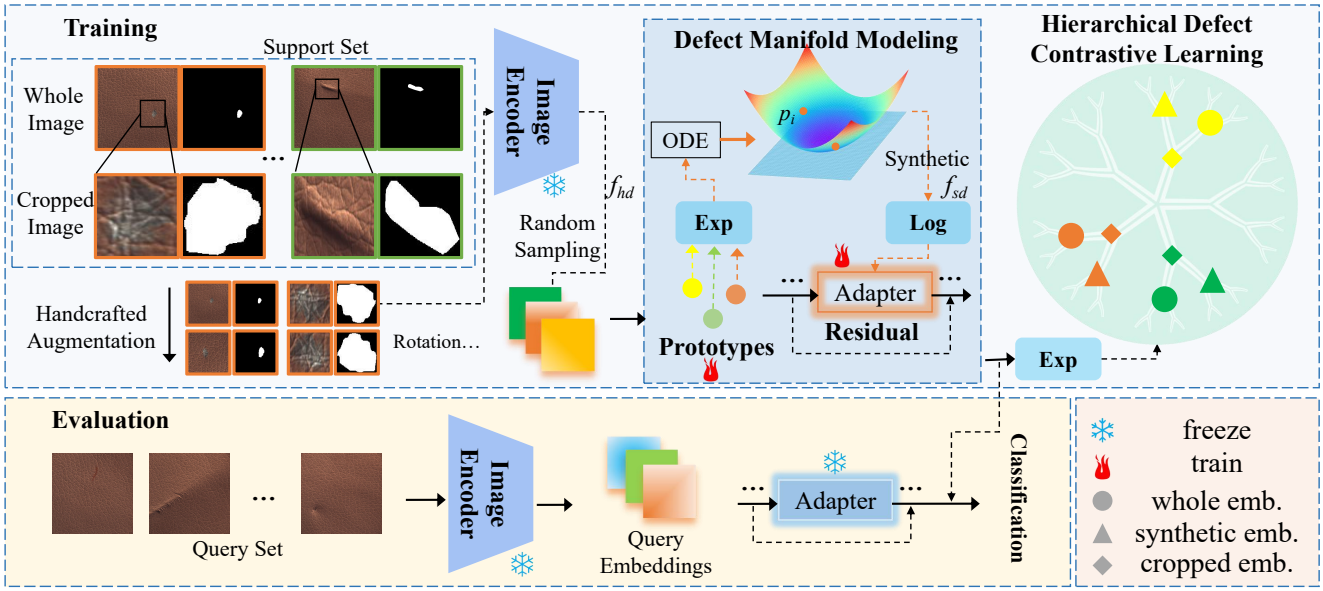


Figure 2. The pipeline of our HypDFS. During training, a pre-trained image encoder is first adopted to extract whole and corresponding cropped features of augmented defective images in support set. These prototype features are then employed to model distribution of each defect manifold in hyperbolic space. Subsequently, synthetic defect features are obtained via a sampling process to assist in training. Furthermore, a residual adapter is trained to dynamically integrate pre-trained general features with defective features, and a hierarchical defect contrastive learning is used to optimize real and synthetic feature embeddings, thereby improving the effectiveness of adapter. During evaluation, input data will be classified into the corresponding category in support set with the guidance of the trained adapter.

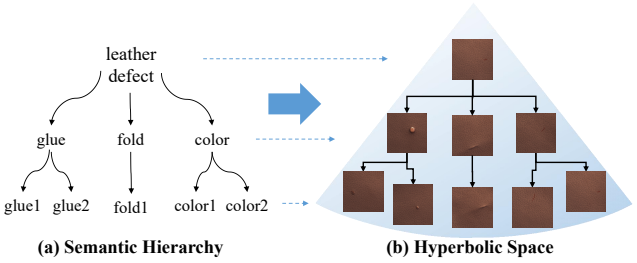


Figure 3. An illustration of (a) a tree-like semantic hierarchy in the context of leather defect from the MVTec-FS benchmark, and (b) ideal relationships among defective images in hyperbolic space.

where $\mathbf{f}_{\text{hd}} \in \mathbb{R}^{N_{hd} \times C}$, N_{hd} and C denote the number of augmented features and channels, respectively. The defect prototype feature $\mathbf{f}$ can be generated by sampling from $\mathbf{f}_{\text{hd}}$, to improve generalization ability, and then mapped from Euclidean space to hyperbolic space for dynamic feature synthesis using the exponential map as

$$\mathbf{h} = \exp_{\mathbf{u}}^{c}(\mathbf{f}) = \mathbf{u} \oplus_c \left(\tanh\left(\sqrt{c} \frac{\lambda_{\mathbf{u}}^{c} \|\mathbf{f}\|}{2} \right) \frac{\mathbf{f}}{\sqrt{c}\|\mathbf{f}\|} \right), \quad (4)$$

where c is the curvature parameter of the hyperbolic space, $\lambda_{\mathbf{u}}^{c} = 2/(1 + c\|\mathbf{u}\|^2)$ is the conformal factor, $\mathbf{u}$ can be simply set to $\mathbf{0}$ as suggested in [9, 18] (*i.e.* $\lambda_{\mathbf{0}}^{c} = 2$), $\mathbf{f} \in \mathbb{R}^C$ is the defect prototype feature in Euclidean space, and $\mathbf{h} \in \mathbb{D}_c^C$ is the mapped feature of $\mathbf{f}$ in hyperbolic space.

3.2.2. Defect Manifold Modeling in Hyperbolic Space

Several well-established ways [14, 17, 22] have been developed for various few-shot distribution modeling in industrial scenarios so far. However, these existing methods all work in Euclidean space. Inspired by [12, 32], we attempt to model defect feature distribution in hyperbolic space.

Let $\boldsymbol{Z}$ denote the random variables of the hyperbolic defect prototype feature $\mathbf{h}$. We then utilize a parameterized wrapped normal distribution with probability density function $\mathcal{P}(\boldsymbol{z}|c, \mathbf{h}, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ to model the Riemann distribution of defect as a projection of a gaussian distribution $\mathcal{N}$ on the tangent space of the given manifold:

$$\mathcal{P}(\boldsymbol{z}|c, \mathbf{h}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \mathcal{N}(\lambda_{\mathbf{u}}^{c} \log_{\mathbf{u}}^{c}(\boldsymbol{z})|\boldsymbol{\mu}, \boldsymbol{\Sigma}) \left(\frac{\sqrt{c}\, d_{\mathbb{D}_c}(\mathbf{h}, \boldsymbol{z})}{\sinh(\sqrt{c}\, d_{\mathbb{D}_c}(\mathbf{h}, \boldsymbol{z}))} \right), \quad (5)$$

where c is the curvature parameter, $\lambda_{\mathbf{u}}^{c}$ is the conformal factor, $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma} \in \mathbb{R}^{C \times C}$ denote the mean and covariance matrix. The logarithmic map $\log_{\mathbf{u}}^{c}(\cdot)$ projects defect prototype feature from hyperbolic space to Euclidean space as

$$\begin{aligned} \mathbf{f} &= \log_{\mathbf{u}}^{c}(\mathbf{h}) \\ &= \frac{2}{\sqrt{c}\lambda_{\mathbf{u}}^{c}} \operatorname{arctanh}(\sqrt{c}\| -\mathbf{u} \oplus_c \mathbf{h}\|) \frac{-\mathbf{u} \oplus_c \mathbf{h}}{\| -\mathbf{u} \oplus_c \mathbf{h}\|}, \end{aligned} \quad (6)$$

where $\mathbf{u}$ can be simply set to $\mathbf{0}$, $\mathbf{h} \in \mathbb{D}_c^C$ and $\mathbf{f} \in \mathbb{R}^C$.

For the parameters $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ in Eq. (5), they can be estimated from a set of defect prototypes $\{\mathbf{f}\}$, but this may result in a bias due to the small number of defect prototypes. More acceptable, we adopt a neural ordinary differential equation (ODE) to calculate the values of $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ for each defect category. Specially, the Runge-Kutta [12] (RK) approach is adopted in our experiments. Given the initial state $\boldsymbol{\mu}_0$ and $\boldsymbol{\Sigma}_0$ of $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$, the final values at the i-th iteration can be obtained using

$$\boldsymbol{\mu}_i,\ \boldsymbol{\Sigma}_i = \mathrm{RK}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0, \mathrm{ODE}_{\boldsymbol{\mu}}, \mathrm{ODE}_{\boldsymbol{\Sigma}}, i), \tag{7}$$

where $\mathrm{ODE}_{\boldsymbol{\mu}}$ and $\mathrm{ODE}_{\boldsymbol{\Sigma}}$ denote the neural networks to learn the parameters $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$, respectively. $\boldsymbol{\mu}_0$ and $\boldsymbol{\Sigma}_0$ are obtained from the given defect prototypes $\{\mathbf{f}\}$.

After model training, we can obtain the synthetic defect feature $\mathbf{h}_{\mathrm{sd}}$ in hyperbolic space through sampling using Eq. (5). Concretely, a sample is first sampled from the gaussian distribution in the Euclidean space, and then mapped to hyperbolic space via an exponential map Eq. (4). It is noted that here we adopt a reparameterization technique to address the issue of non-differentiability in the sampling process following the practices in [12, 20, 41].

3.2.3. Defect Residual Adapter and Hierarchical Defect Contrastive Learning

Defect Residual Adapter. To fully utilize the rich knowledge contained in pre-trained CLIP model for few-shot defect classification task, while achieving rapid and efficient adaptation of the model to specific domain, we train a lightweight residual adapter using real and synthetic defect data, which can guide the model to better learn defect related features according to the needs of specific industrial scenario. Given the real defect feature $\mathbf{f}$ and the synthesis defect feature $\mathbf{f}_{\mathrm{sd}}$ in Euclidean space, in which $\mathbf{f}_{\mathrm{sd}}$ is obtained from the hyperbolic synthesis defect feature $\mathbf{h}_{\mathrm{sd}}$ using the logarithmic map Eq. (6), we then obtain the fine-tuned adapted defect feature $\mathbf{f}_{\mathrm{out}}$ using:

$$\mathbf{f}_{\mathrm{in}} = \mathrm{CON}(\mathbf{f}, \mathbf{f}_{\mathrm{sd}}), \quad \mathbf{f}_{\mathrm{out}} = \mathrm{SiLU}(\mathbf{W}\mathbf{f}_{\mathrm{in}}) + \mathbf{f}_{\mathrm{in}}, \tag{8}$$

where CON represents the feature concatenation operation, $\mathrm{SiLU}(\cdot)$ is an activation function, $\mathbf{W} \in \mathbb{R}^{C\times C}$ denotes the learnable weight parameter, and $\mathbf{f}_{\mathrm{out}}$ is used in the calculation of vanilla cross-entropy loss L_{ce}.

Hierarchical Defect Contrastive Learning. In our HypDFS method, the learning process mainly involves three objectives: defect feature adaptation, defect feature synthesis, and defect hierarchical representation. Among them, feature adaptation is a common goal in the fine-tuning of large-scale pre-trained model. Here, we optimize a vanilla cross-entropy loss L_{ce} to achieve this goal. Furthermore, the feature synthesis process is constrained by

$$L_s = \max\left(m_d - \sum_i \sum_{j\neq i} d_{\mathbb{D}_c}(\mathbf{h}^i, \mathbf{h}^j_{\mathrm{sd}}), 0\right) + \max\left(m_d - \sum_i \sum_{j\neq i} d_{\mathbb{D}_c}(\mathbf{h}^i_{\mathrm{sd}}, \mathbf{h}^j_{\mathrm{sd}}), 0\right), \tag{9}$$

where $\{\mathbf{h}^i\}$ and $\{\mathbf{h}^i_{\mathrm{sd}}\}$ denote two sets of real defect prototype and synthetic defect features in hyperbolic space, and the margin $m_d = \sum_i \sum_{j\neq i} d_{\mathbb{D}_c}(\mathbf{h}^i, \mathbf{h}^j)$. The first term prevents the synthetic features from clustering near the prototypes excessively, and the second term prevents the synthetic features from being too close to each other and clustering together. In addition, we exploit a defect contrastive loss function based on hyperbolic distance to better capture the hierarchical relationships between defective images. Specifically, let $\mathbf{h}_{\mathrm{wd}}$ and $\mathbf{h}_{\mathrm{cd}}$ denote the whole defective feature and the corresponding cropped defective region feature in hyperbolic space. Then our proposed hierarchical defect contrastive loss function can be formulated as

$$L_{hdcl} = L_{ce} + \alpha L_s + \beta L_{cl}, \tag{10}$$

in which α and β are hyperparameters, and the contrastive loss between $\mathbf{h}_{\mathrm{wd}}$ and $\mathbf{h}_{\mathrm{cd}}$ is given by

$$L_{cl}(\mathbf{h}_{\mathrm{wd}}, \mathbf{h}_{\mathrm{cd}}) = -\sum_i \log \frac{\exp\left(d_{\mathbb{D}_c}(\mathbf{h}^i_{\mathrm{wd}}, \mathbf{h}^i_{\mathrm{cd}})/\tau\right)}{\sum_{j\neq i} \exp(d_{\mathbb{D}_c}(\mathbf{h}^i_{\mathrm{wd}}, \mathbf{h}^j_{\mathrm{cd}})/\tau)}, \tag{11}$$

where $\mathbf{h}_{\mathrm{wd}} \in \{\mathbf{h}, \mathbf{h}_{\mathrm{sd}}\}$, $d_{\mathbb{D}_c}$ is the hyperbolic distance using Eq. (2), and τ is a temperature hyperparameter.

4. Experiments

4.1. Experimental Settings

In this section, we empirically evaluate the effectiveness of our HypDFS method. Specifically, we conduct experiments on the MVTec-FS benchmark and the standard MTD dataset under few-shot settings. Following the evaluation protocol in [29], we adopt the mean classification accuracy to measure the performance of the different methods.

4.1.1. Datasets

MVTec-FS [29] is an important few-shot benchmark for industrial defect classification. It is officially released in 2025 as a few-shot version of the prevalent MVTec-AD. The MVTec-FS dataset comprises 14 industrial product categories: 5 texture categories and 9 object categories. A total of 46 defect types have been annotated, including common defects such as scratches, dents, dirt, and deformation. The number of instances for each defect type ranges from 9 to 58, totaling 1,228 defect images. It simulates the challenge

Table 1. Mean classification accuracy (%) comparison with state-of-the-art methods on MVTec-FS and MTD under few-shot settings.

Dataset	K	CLIP-A [10]	CLIP-ProtoNet [43]	CLIP-KNN [29]	CLIP-LinearProb[29]	Tip-A-F [56]	Zip-A-F [29]	HypDFS (Ours)
MVTec-FS	1	61.5	60.6	60.4	62.2	62.0	73.7	**79.3 (+5.6)**
	3	75.7	76.0	76.0	78.0	77.5	86.1	**89.3 (+3.2)**
	5	81.0	79.9	76.7	82.5	81.9	89.4	**91.7 (+2.3)**
MTD	1	51.7	50.6	50.6	54.5	52.9	55.4	**59.4 (+4.0)**
	3	67.8	64.9	55.4	69.9	67.9	67.6	**78.6 (+8.7)**
	5	76.8	74.2	65.2	78.0	76.6	78.6	**86.4 (+7.8)**

Table 2. Classification accuracy (%) comparison with state-of-the-art methods on 9 object categories of MVTec-FS benchmark.

K	Method	Bottle	Cable	Capsule	Hazelnut	MetalNut	Pill	Screw	Transistor	Zipper
1	CLIP-A [10]	51.9	61.8	44.5	66.3	59.6	54.3	72.7	63.8	77.3
	CLIP-ProtoNet [43]	50.6	62.8	46.8	62.9	62.0	59.3	74.7	55.2	74.6
	CLIP-KNN [29]	50.6	62.8	47.2	62.9	61.6	58.5	75.0	55.2	74.4
	CLIP-LinearProb [29]	55.0	62.8	49.1	69.1	60.0	63.7	73.0	56.2	73.9
	Tip-A-F [56]	51.3	63.2	46.8	65.1	61.2	62.7	74.3	61.0	76.6
	Zip-A-F [29]	58.1	77.5	60.4	77.1	71.6	72.4	67.3	**89.5**	79.3
	HypDFS (Ours)	**66.8**	**88.1**	**64.9**	**83.4**	**80.8**	**77.5**	**78.3**	85.7	**82.6**
3	CLIP-A [10]	68.1	77.9	61.5	76.0	72.0	68.6	85.7	88.6	85.1
	CLIP-ProtoNet [43]	70.0	80.0	63.0	77.7	71.2	69.1	87.0	85.7	83.4
	CLIP-KNN [29]	70.0	80.0	63.0	77.7	71.2	69.1	87.0	85.7	83.4
	CLIP-LinearProb [29]	71.3	82.1	63.4	78.3	73.6	75.6	86.7	87.6	83.4
	Tip-A-F [56]	71.9	81.8	61.5	78.3	73.2	73.3	87.0	87.6	85.4
	Zip-A-F [29]	76.9	**93.0**	74.0	82.3	83.6	83.5	88.3	**100.0**	88.8
	HypDFS (Ours)	**77.5**	**93.0**	**81.1**	**88.0**	**94.0**	**85.4**	**92.3**	99.1	**91.5**
5	CLIP-A [10]	67.5	83.9	69.8	78.9	78.8	77.3	90.7	93.3	87.6
	CLIP-ProtoNet [43]	67.5	84.9	72.1	74.9	78.4	74.6	89.3	99.1	87.1
	CLIP-KNN [29]	65.6	78.3	62.3	73.1	79.2	67.6	89.7	97.1	79.8
	CLIP-LinearProb [29]	70.0	86.7	69.8	78.9	80.0	81.2	92.7	92.4	86.3
	Tip-A-F [56]	68.1	85.6	70.2	77.1	80.8	80.7	91.0	96.2	87.6
	Zip-A-F [29]	77.5	**93.0**	81.1	88.6	91.2	84.7	92.0	**100.0**	90.0
	HypDFS (Ours)	**81.8**	92.3	**84.9**	**92.0**	**98.8**	**87.9**	**94.0**	**100.0**	**93.6**

of scarce defect samples in actual production. For example, the carpet defect contains 5 subclasses of defects, with around 10 images of each type in the training and test sets, respectively.

MTD [15] is a standard dataset for promoting intelligent manufacturing and quality control. It is designed for defect detection in the production process of magnetic tiles, covering five typical types of defects including blowhole, crack, break, fray, and uneven, totaling 392 defective images. Accurate classification in few-shot settings is more challenging due to low visibility and significant differences in defect shapes. Note that each defective image in MVTec-FS and MTD datasets contains pixel level annotations, outlining the boundaries and morphology of the defect area accurately.

4.1.2. Implementation Details

We adopt the AdamW optimizer for training, and the initial learning rate is $1e$-4 with a cosine annealing decay schedule. The batch size is 32 by default, which is slightly larger than the maximum value in the training set for each defect category to facilitate unified training. Note that the actual batch size and the batch samples may vary depending on the shot values. We use AlphaCLIP [46] equipped with ViTL/14 [7] as the default feature extractor. For the neural networks $\mathrm{ODE}_{\boldsymbol{\mu}}$ and $\mathrm{ODE}_{\boldsymbol{\Sigma}}$, we adopt fully-connected layer and self-attention layer following the settings in [12]. The curvature parameter c of the hyperbolic space is set to 0.05 by default, obtained by a heuristic way. The parameters α, β, and τ are set to 0.1, 0.1, and 0.3, respectively.

We adopt N-way K-shot settings for few-shot defect classification in our experiments, in which N is the number of defect categories and $K \in \{1, 3, 5\}$ represents the number of training images for each category. Following the common settings in [29], the model is first trained in the support set that is sampled from the training set during training and then evaluated in the query set during evaluation. Specially, for the MTD dataset, the number of images in each class varies significantly, ranging from 37 to 115. If training and test data are divided into roughly equal proportions, the image diversity in the training set of some categories will be significantly higher than that of other categories. Therefore, we do not divide them in equal proportions. In our experiments, we simply utilize the first 25 images in each category for training and the remaining images for testing, re-testing the results using open-source code from [29]. Such settings render a more reasonable comparison under few-shot classification task. All experiments are conducted on a single NVIDIA GeForce RTX 3090 GPU.

4.2. Results and Analysis

4.2.1. Comparison with State-of-the-art Methods

Tab. 1 highlights the mean classification accuracy of current widely adopted methods on the MVTec-FS benchmark and MTD dataset under few-shot settings. More detailed results on 9 object categories and 5 texture categories of MVTec-FS can be found in Tab. 2 and Tab. 3, respectively. It can be seen from Tab. 1 that our HypDFS method obtains the highest scores and surpasses the existing Euclidean space based methods by increasing the gains of 5.6%, 3.2%, and 2.3% on the MVTec-FS benchmark and 4.0%, 8.7%, and 7.8% on the MTD dataset, respectively. The empirical experiments demonstrate the superiority of our proposed HypDFS method by synthesizing the additional hyperbolic defect features. More specifically, this hyperbolic modeling scheme captures the hierarchical relationships between defective images, yielding more favorable representations for downstream defect classification tasks.

4.2.2. Effect of Different Pre-trained Architectures

The current pre-trained large-scale multimodal models play a crucial role in cross domain few-shot classification tasks. Hence, to explore the impact of different pre-trained architectures, we evaluate two popular CLIP models: VanillaClip [38] and AlphaCLIP [46], a derived version that focuses more on specific regions. Tab. 4 presents the mean classification accuracy under the different pre-trained models on the popular MTD dataset when K=5. Note that in Tab. 4, we do not fine tune the model and only adopt a vanilla cross-entropy loss for classification. It can be seen from these results that 1) different pre-trained CLIP models perform significantly differently, and 2) the pre-trained ViT architecture is better than ResNet-based architectures. To further validate the ability of AlphaCLIP to focus on specific defect areas, we conduct an additional experiment with the same weight assigned to the mask (*i.e.* AlphaCLIP-S in Tab. 4). From the results, it can be seen that with the help of mask, AlphaCLIP has extracted more effective feature representations that are more focused on defect regions, obtaining a performance of 17.6% (69.5% vs. 51.9%) improvement on ViT-L/14. Hence, we select the AlphaCLIP with ViT-L/14 backbone as default setting in subsequent experiments.

4.2.3. Effect of Curvature Parameter

To show the effect of different values of curvature parameter c, we evaluate three strategies on curvature parame-

Table 3. Classification accuracy (%) comparison with state-of-the-art methods on 5 texture categories of MVTec-FS benchmark.

K	Method	Car.	Gri.	Lea.	Til.	Wo.
1	CLIP-A	66.4	42.9	65.5	70.5	64.0
	CLIP-ProtoNet	60.5	42.0	62.7	71.0	63.4
	CLIP-KNN	59.6	42.0	62.7	69.5	62.9
	CLIP-ProbLinear	60.5	41.6	64.6	73.3	68.0
	Tip-A-F	60.9	44.9	63.2	72.9	64.6
	Zip-A-F	78.6	50.6	82.3	**96.2**	71.4
	HypDFS (Ours)	**85.0**	**52.6**	**91.3**	**96.2**	**76.6**
3	CLIP-A	70.9	57.6	77.3	89.5	81.4
	CLIP-ProtoNet	71.4	57.6	78.2	86.7	82.3
	CLIP-KNN	71.4	57.6	78.2	86.7	82.3
	CLIP-ProbLinear	74.1	60.0	82.3	91.0	83.1
	Tip-A-F	72.7	60.8	80.5	89.1	82.3
	Zip-A-F	**85.0**	71.8	90.9	**97.6**	90.0
	HypDFS (Ours)	84.1	**76.7**	**94.1**	**97.6**	**95.7**
5	CLIP-A	75.5	64.9	87.7	91.0	86.9
	CLIP-ProtoNet	73.6	59.6	83.6	89.1	84.9
	CLIP-KNN	74.1	55.5	81.8	89.5	79.7
	CLIP-ProbLinear	79.6	66.9	88.6	93.3	88.3
	Tip-A-F	74.5	65.3	88.2	91.0	89.7
	Zip-A-F	85.9	80.8	**92.7**	**97.6**	96.6
	HypDFS (Ours)	**86.8**	**82.4**	**92.7**	**97.6**	**98.5**

Table 4. Mean classification accuracy (%) of different pre-trained architectures on MTD dataset when $K = 5$.

Type	Backbone	Accuracy
VanillaClip	ResNet50	24.3
	ResNet101	33.2
	ViT-B/16	52.7
	ViT-L/14	52.5
	ViT-L/14@336	51.1
AlphaCLIP	ViT-B/16	65.0
	ViT-L/14	**69.5**
AlphaCLIP-S	ViT-B/16	50.0
	ViT-L/14	51.9

Table 5. Mean classification accuracy (%) of different curvature values on MTD dataset when $K = 5$.

Type	Curvature c	Accuracy
Zero	0	78.1
Fixed	0.01	79.0
	0.05	**79.2**
	0.1	79.1
	0.3	78.9
	0.5	79.1
	0.7	79.0
	1.0	79.0
Learnable	0.05	78.9

ter, *i.e*. zero, fixed and learnable. We use a fixed learning rate uniformly without a scheduled strategy for more stable curvature parameter learning. Tab. 5 presents the corresponding results on MTD dataset when K=5. It can be seen that a proper curvature parameter value can improve the classification accuracy. When c is set to 0.1, the score can be increased by 1.0% (78.1% vs. 79.1%) compared to the Euclidean baseline (*i.e*. c=0). In addition, simply setting curvature value as a learnable parameter does not result in better performance (79.2% vs. 78.9%) in our experiments. Therefore, it remains a challenge to automatically find the optimal curvature value of hyperbolic metric space that is more suitable for few-shot classification tasks. In our future work, we will explore other adaptive ways to find a more ideally and interpretable curvature value.

4.2.4. Effect of Defect Residual Adapter

Currently, the adaptation of pre-trained general features to defective features plays an important role in the application of large-scale models. To evaluate such effects, we train a defect residual adapter and present the results in Tab. 6, where notation 'NO' denotes that the adapter is not used for fine-tuning, and notation 'Real' and 'Real & Syn' mean adopting real and synthetic defect features to fine-tune the adapter. From these results, we can see that just a simple adaptation can significantly increase gains by 8.6% (69.5% vs. 78.1%) when K=5. Furthermore, the mean classification score can be further improved by 8.3% (78.1% vs. 86.4%) after using additional synthetic defect features, showing the effectiveness of the proposed HypDFS method and the potential for defect synthesis in hyperbolic space.

4.2.5. Effect of Hierarchical Defect Contrastive Learning

Tab. 7 clearly shows the effectiveness of the proposed hierarchical defect contrastive loss L_{hdcl}, in which 'Euclidean' represents using cross-entropy loss only in Euclidean space, 'Hyperbolic' denotes optimizing defect embeddings using hyperbolic space and hyperbolic distance Eq. (2) but without using synthetic features, and 'HDCL' (*i.e*. HypDFS) means using our proposed hierarchical defect contrastive loss. Note that the additional cropped defective region feature extracted by the pre-trained feature extractor $\mathrm{CLIP}(\cdot)$ has been adopted in HDCL. We can see that the mean classification accuracy has improved by 7.2% (86.4% vs. 79.2%) with the help of the proposed L_{hdcl} under hyperbolic settings (HDCL vs. Hyperbolic), surpassing the Euclidean space baseline (*i.e*. Euclidean) by a large margin. This implies the superiority of modeling and synthesizing defect features in hyperbolic space, which is further amplified by the proposed hierarchical defect contrastive loss.

Table 6. Mean classification accuracy (%) of defect residual adapter under different training strategies on MTD dataset.

Dataset	K	NO	Real	Real & Syn
MTD	1	51.0	55.0	**59.4**
	3	59.3	67.4	**78.6**
	5	69.5	78.1	**86.4**

Table 7. Mean classification accuracy (%) comparison of the proposed hierarchical defect contrastive loss with the Euclidean and hyperbolic settings on MTD dataset.

Dataset	K	Euclidean	Hyperbolic	HDCL
MTD	1	55.0	55.9	**59.4**
	3	67.4	68.2	**78.6**
	5	78.1	79.2	**86.4**

5. Conclusion

In this paper, we have explored the defect synthesis in hyperbolic space and presented a hyperbolic defect feature synthesis (HypDFS) method. Specifically, we first train a residual adapter to dynamically fuse substantial pre-trained general features with industrial defect features. Then we model the potential distribution of the defects via a small number of hyperbolic defect prototypes and further optimize the real and synthetic defect features with the proposed hierarchical defect contrastive loss in hyperbolic space, yielding a better generalized defect representation that is more conducive to downstream few-shot classification task. Extensive experiments on benchmarking datasets show the superiority of our HypDFS and the promising prospects for defect synthesis in hyperbolic space.

Limitations. In this paper, we only consider defect feature synthesis in two industrial image benchmarks. The performance of our HypDFS method in more benchmarks needs to be further evaluated. Furthermore, exploring other adaptive ways to find a more ideally curvature may be an alternative direction for this work in the future.

References

[1] Qi Bi, Jingjun Yi, Haolan Zhan, Wei Ji, and Gui-Song Xia. Learning fine-grained domain generalization via hyperbolic state space hallucination. In *AAAI Conference on Artificial Intelligence*, pages 1853–1861, 2025. 2

[2] Meiyun Chen, Zijian Liang, Zhenxiao Fu, Deqing Kong, and Dawei Zhang. An efficient defect detection method for MLCC based on EHL-DETR deep learning model. *IEEE Transactions on Instrumentation and Measurement*, 74:1–9, 2025. 1

[3] Thomas Defard, Aleksandr Setkov, Angelique Loesch, and Romaric Audigier. Padim: a patch distribution modeling framework for anomaly detection and localization. In *International Conference on Pattern Recognition*, pages 475–489, 2021. 2

[4] Karan Desai, Maximilian Nickel, Tanmay Rajpurohit, Justin Johnson, and Shanmukha Ramakrishna Vedantam. Hyperbolic image-text representations. In *International Conference on Machine Learning*, pages 7694–7731, 2023. 2

[5] Jan Diers and Christian Pigorsch. A survey of methods for automated quality control based on images. *International Journal of Computer Vision*, 131(10):2553–2581, 2023. 1

[6] Yuran Dong, Cheng Xie, Luyao Xu, Hongming Cai, Weiming Shen, and Haoyuan Tang. Generative and contrastive combined support sample synthesis model for few-/zero-shot surface defect recognition. *IEEE Transactions on Instrumentation and Measurement*, 73:1–15, 2023. 2

[7] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2020. 6

[8] Yuxuan Duan, Yan Hong, Li Niu, and Liqing Zhang. Few-shot defect image generation via defect-aware feature manipulation. In *AAAI Conference on Artificial Intelligence*, pages 571–578, 2023. 1

[9] Aleksandr Ermolov, Leyla Mirvakhabova, Valentin Khrulkov, Nicu Sebe, and Ivan Oseledets. Hyperbolic vision transformers: Combining improvements in metric learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7409–7419, 2022. 3, 4

[10] Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. Clip-adapter: Better vision-language models with feature adapters. *International Journal of Computer Vision*, 132(2): 581–595, 2024. 2, 6

[11] Zhi Gao, Yuwei Wu, Yunde Jia, and Mehrtash Harandi. Curvature generation in curved spaces for few-shot learning. In *IEEE/CVF International Conference on Computer Vision*, pages 8691–8700, 2021. 3

[12] Zhi Gao, Yuwei Wu, Yunde Jia, and Mehrtash Harandi. Hyperbolic feature augmentation via distribution estimation and infinite sampling on manifolds. In *Advances in Neural Information Processing Systems*, pages 34421–34435, 2022. 4, 5, 6

[13] Zeyu Han, Chao Gao, Jinyang Liu, Jeff Zhang, and Sai Qian Zhang. Parameter-efficient fine-tuning for large models: A comprehensive survey. *Transactions on Machine Learning Research*, pages 1–44, 2024. 2

[14] Teng Hu, Jiangning Zhang, Ran Yi, Yuzhen Du, Xu Chen, Liang Liu, Yabiao Wang, and Chengjie Wang. Anomalydiffusion: Few-shot anomaly image generation with diffusion model. In *AAAI conference on artificial intelligence*, pages 8526–8534, 2024. 1, 4

[15] Yibin Huang, Congying Qiu, and Kui Yuan. Surface defect saliency of magnetic tile. *The Visual Computer*, 36(1):85–96, 2020. 6

[16] Xi Jiang, Jian Li, Hanqiu Deng, Yong Liu, Bin-Bin Gao, Yifeng Zhou, Jialin Li, Chengjie Wang, and Feng Zheng. MMAD: A comprehensive benchmark for multimodal large language models in industrial anomaly detection. In *International Conference on Learning Representations*, 2025. 1

[17] Ying Jin, Jinlong Peng, Qingdong He, Teng Hu, Jiafu Wu, Hao Chen, Haoxuan Wang, Wenbing Zhu, Mingmin Chi, Jun Liu, and Yabiao Wang. Dual-interrelated diffusion model for few-shot anomaly image generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 30420–30429, 2025. 1, 2, 4

[18] Valentin Khrulkov, Leyla Mirvakhabova, Evgeniya Ustinova, Ivan Oseledets, and Victor Lempitsky. Hyperbolic image embeddings. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6418–6428, 2020. 3, 4

[19] Tongwha Kim and Kamran Behdinan. Advances in machine learning and deep learning applications towards wafer map defect recognition and classification: a review. *Journal of Intelligent Manufacturing*, 34(8):3215–3247, 2023. 1

[20] Wonyeol Lee, Hangyeol Yu, and Hongseok Yang. Reparameterization gradient for non-differentiable models. In *Advances in Neural Information Processing Systems*, pages 5558–5568, 2018. 5

[21] Tao Lei, Junwen Bai, Siddhartha Brahma, Joshua Ainslie, Kenton Lee, Yanqi Zhou, Nan Du, Vincent Y. Zhao, Yuexin Wu, Bo Li, Yu Zhang, and Ming-Wei Chang. Conditional adapters: Parameter-efficient transfer learning with fast inference. In *Advances in Neural Information Processing Systems*, pages 8152–8172, 2023. 2

[22] Chun-Liang Li, Kihyuk Sohn, Jinsung Yoon, and Tomas Pfister. Cutpaste: Self-supervised learning for anomaly detection and localization. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 9664–9674, 2021. 1, 2, 4

[23] Huimin Li and Junlin Hu. Feature consistency learning for anomaly detection. *IEEE Transactions on Instrumentation and Measurement*, 74:1–9, 2025. 1

[24] Huimin Li, Zhentao Chen, Yunhao Xu, and Junlin Hu. Hyperbolic anomaly detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17511–17520, 2024. 2, 3

[25] Liyuan Lin, Shuxian Zhao, Yiran Zhang, Aolin Wen, Shun Zhang, Jingpeng Yan, Ying Wang, and Yuan Zhou. Purposive data augmentation strategy and lightweight classification model for small sample industrial defect dataset. *IEEE*

Transactions on Industrial Informatics, 20(9):11475–11484, 2024. 1

[26] Juhua Liu, Chaoyue Wang, Hai Su, Bo Du, and Dacheng Tao. Multistage gan for fabric defect detection. *IEEE Transactions on Image Processing*, 29:3388–3400, 2019. 2

[27] Yuanpei Liu, Zhenqi He, and Kai Han. Hyperbolic category discovery. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9891–9900, 2025. 2, 3

[28] Zhikang Liu, Yiming Zhou, Yuansheng Xu, and Zilei Wang. Simplenet: A simple network for image anomaly detection and localization. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20402–20411, 2023. 2

[29] Shuai Lyu, Rongchen Zhang, Zeqi Ma, Fangjian Liao, Dongmei Mo, and Waikeung Wong. Mvrec: A general few-shot defect classification model using multi-view region-context. In *AAAI Conference on Artificial Intelligence*, pages 5937–5945, 2025. 1, 2, 3, 5, 6, 7

[30] Paolo Mandica, Luca Franco, Konstantinos Kallidromitis, Suzanne Petryk, and Fabio Galasso. Hyperbolic learning with multimodal large language models. In *European Conference on Computer Vision*, pages 382–398, 2024. 2, 3

[31] Pascal Mettes, Mina Ghadimi Atigh, Martin Keller-Ressel, Jeffrey Gu, and Serena Yeung. Hyperbolic deep learning in computer vision: A survey. *International Journal of Computer Vision*, 132(9):3484–3508, 2024. 2

[32] Yoshihiro Nagano, Shoichiro Yamaguchi, Yasuhiro Fujita, and Masanori Koyama. A wrapped normal distribution on hyperbolic space for gradient-based learning. In *International Conference on Machine Learning*, pages 4693–4702, 2019. 4

[33] Maximillian Nickel and Douwe Kiela. Learning continuous hierarchies in the lorentz model of hyperbolic geometry. In *International Conference on Machine Learning*, pages 3779–3788, 2018. 2

[34] Avik Pal, Max van Spengler, Guido Maria D'Amely di Melendugno, Alessandro Flaborea, Fabio Galasso, and Pascal Mettes. Compositional entailment learning for hyperbolic vision-language models. In *International Conference on Learning Representations*, 2025. 2, 3

[35] Junting Pan, Ziyi Lin, Xiatian Zhu, Jing Shao, and Hongsheng Li. St-adapter: Parameter-efficient image-to-video transfer learning. In *Advances in Neural Information Processing Systems*, pages 26462–26477, 2022. 2

[36] Wei Peng, Tuomas Varanka, Abdelrahman Mostafa, Henglin Shi, and Guoying Zhao. Hyperbolic deep neural networks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(12):10023–10044, 2021. 2

[37] Arun Poudel, Mohammad Salman Yasin, Jiafeng Ye, Jia Liu, Aleksandr Vinel, Shuai Shao, and Nima Shamsaei. Feature-based volumetric defect classification in metal additive manufacturing. *Nature Communications*, 13(1):6369, 2022. 1

[38] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763, 2021. 7

[39] Weilun Ren, Kangkang Song, Chin-Yin Chen, Yu Chen, Jingshan Hong, Mingrui Fan, Xiong Ouyang, Ying Zhu, and Jiangjian Xiao. Dd-aug: A knowledge-to-image synthetic data augmentation pipeline for industrial defect detection. *IEEE Transactions on Industrial Informatics*, 21(3):2284–2293, 2025. 1

[40] Frederic Sala, Chris De Sa, Albert Gu, and Christopher Ré. Representation tradeoffs for hyperbolic embeddings. In *International Conference on Machine Learning*, pages 4460–4469, 2018. 2

[41] Tim Salimans and Durk P Kingma. Weight normalization: A simple reparameterization to accelerate training of deep neural networks. In *Advances in Neural Information Processing Systems*, page 901, 2016. 5

[42] Qingfeng Shi, Jing Wei, Fei Shen, and Zhengtao Zhang. Few-shot defect image generation based on consistency modeling. In *European Conference on Computer Vision*, pages 360–376, 2024. 1

[43] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems*, pages 1–9, 2017. 6

[44] Yisheng Song, Ting Wang, Puyu Cai, Subrota K Mondal, and Jyoti Prakash Sahoo. A comprehensive survey of few-shot learning: Evolution, applications, challenges, and opportunities. *ACM Computing Surveys*, 55(13s):1–40, 2023. 2

[45] Jiaojiao Su, Qiwu Luo, Weihua Gui, and Chunhua Yang. Few-shot parameter efficient finetuning for SAM in salient steel surface defect detection. *IEEE Transactions on Industrial Informatics*, 21(10):7742–7753, 2025. 1

[46] Zeyi Sun, Ye Fang, Tong Wu, Pan Zhang, Yuhang Zang, Shu Kong, Yuanjun Xiong, Dahua Lin, and Jiaqi Wang. Alpha-clip: A clip model focusing on wherever you want. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13019–13029, 2024. 6, 7

[47] Yi-Lin Sung, Jaemin Cho, and Mohit Bansal. Vl-adapter: Parameter-efficient transfer learning for vision-and-language tasks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5227–5237, 2022. 2

[48] Xian Tao, Xinyi Gong, Xin Zhang, Shaohua Yan, and Chandranath Adak. Deep learning for unsupervised anomaly localization in industrial images: A survey. *IEEE Transactions on Instrumentation and Measurement*, 71:1–21, 2022. 1

[49] Huiyan Wang, Ruihao Peng, Yiheng Zhu, Jiachen Li, Mengchu Zhou, Ming Ying, and Shouguang Wang. Few-shot photovoltaic film defect detection with contextual ensemble language-image multimodal network. *IEEE Transactions on Industrial Informatics*, 21(8):6274–6283, 2025. 1

[50] Menglin Yang, Min Zhou, Rex Ying, Yankai Chen, and Irwin King. Hyperbolic representation learning: Revisiting and advancing. In *International Conference on Machine Learning*, pages 39639–39659, 2023. 2

[51] Vitjan Zavrtanik, Matej Kristan, and Danijel Skočaj. Draema discriminatively trained reconstruction embedding for surface anomaly detection. In *IEEE/CVF International Conference on Computer Vision*, pages 8330–8339, 2021. 1, 2

[52] Vitjan Zavrtanik, Matej Kristan, and Danijel Skočaj. Dsr–a dual subspace re-projection network for surface anomaly de-

tection. In *European Conference on Computer Vision*, pages 539–554, 2022. 2

[53] Gongjie Zhang, Kaiwen Cui, Tzu-Yi Hung, and Shijian Lu. Defect-gan: High-fidelity defect synthesis for automated defect inspection. In *IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2524–2534, 2021. 1

[54] Gongjie Zhang, Kaiwen Cui, Tzu-Yi Hung, and Shijian Lu. Defect-gan: High-fidelity defect synthesis for automated defect inspection. In *IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2524–2534, 2021. 1, 2

[55] Kang Zhang, Limin Liao, and Yonghua Wang. Discrete wavelet transform-based detection transformer for battery weld defect detection. *IEEE Transactions on Instrumentation and Measurement*, 74:1–11, 2025. 1

[56] Renrui Zhang, Wei Zhang, Rongyao Fang, Peng Gao, Kunchang Li, Jifeng Dai, Yu Qiao, and Hongsheng Li. Tip-adapter: Training-free adaption of clip for few-shot classification. In *European Conference on Computer Vision*, pages 493–510, 2022. 6

[57] Ximiao Zhang, Min Xu, and Xiuzhuang Zhou. Realnet: A feature selection network with realistic synthetic anomaly for anomaly detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16699–16708, 2024. 2

[58] Qinjun Zhao, Shanchang Fang, Yueyang Li, Hongwei Shang, Han Zhang, and Tao Shen. Track defect detection based on improved yolov5s. *IEEE Transactions on Industrial Informatics*, 21(4):3346–3355, 2025. 1