Say No to Freeloader: Protecting Intellectual Property of Your Deep Model

Lianyu Wang, Meng Wang, Huazhu Fu, *Senior Member, IEEE*,
and Daoqiang Zhang, *Senior Member, IEEE,*

Abstract—Model intellectual property (IP) protection has attracted growing attention as science and technology advancements stem from human intellectual labor and computational expenses. Ensuring IP safety for trainers and owners is of utmost importance, particularly in domains where ownership verification and applicability authorization are required. A notable approach to safeguarding model IP involves proactively preventing the use of well-trained models of authorized domains from unauthorized domains. In this paper, we introduce a novel Compact Un-transferable Pyramid Isolation Domain (CUPI-Domain) which serves as a barrier against illegal transfers from authorized to unauthorized domains. Drawing inspiration from human transitive inference and learning abilities, the CUPI-Domain is designed to obstruct cross-domain transfers by emphasizing the distinctive style features of the authorized domain. This emphasis leads to failure in recognizing irrelevant private style features on unauthorized domains. To this end, we propose novel CUPI-Domain generators, which select features from both authorized and CUPI-Domain as anchors. Then, we fuse the style features and semantic features of these anchors to generate labeled and style-rich CUPI-Domain. Additionally, we design external Domain-Information Memory Banks (DIMB) for storing and updating labeled pyramid features to obtain stable domain class features and domain class-wise style features. Based on the proposed whole method, the novel style and discriminative loss functions are designed to effectively enhance the distinction in style and discriminative features between authorized and unauthorized domains, respectively. Moreover, we provide two solutions for utilizing CUPI-Domain based on whether the unauthorized domain is known: target-specified CUPI-Domain and target-free CUPI-Domain. By conducting comprehensive experiments on various public datasets, we validate the effectiveness of our proposed CUPI-Domain approach with different backbone models. The results highlight that our method offers an efficient model intellectual property protection solution.

Index Terms—Deep model IP, domain transfer, deep learning.

1 Introduction

Deep Neural networks have made significant strides in diverse areas of machine learning. However, recent achievements heavily relied on abundant and high-quality data [1], [2], dedicated training resources [3], [4], and meticulous manual fine-tuning [5], [6]. Acquiring a proficiently trained deep model demands substantial time and effort in practice. While some DNN models, such as VGG [7], Inception-v3 [8], and SWIN [9], are publicly available for non-commercial use, many owners of models used in commercial applications prefer to keep their trained DNN models private. This is often due to business considerations and concerns related to privacy and security, especially in critical applications like autonomous driving, face recognition, and intrusion detection [10].

Unfortunately, once these models are sold, they can be easily copied and redistributed, infringing on the interests of the developers and potentially increasing security risks in safety-critical applications, causing incalculable damage. For instance, in commercial Machine Learning as a Service (MLaaS) platforms [11], well-trained deep models have high business value. They should be treated as the intellectual property (IP) of their creators/owners. When these models are uploaded to public platforms or deployed as remote services on the cloud, malicious users may steal them for financial gain. One of the most concerning threats is "Will releasing the model make it easy for the main competitor to copy this new feature and hurt owner differentiation in the market?". Therefore, it is crucial to regard and protect these models as valuable scientific and technological intellectual property (IP) [12], [13], [14].

Traditionally, model owners design, train, deliver, and deploy efficient deep models based on the authorized domain (represented by blue squares in Fig. 1). They grant specific users the right to use their well-trained models, enabling them to obtain correct predictions on the authorized domain, as depicted in Fig. 1 (a). However, since these models are trained with the overall features of the authorized domain, they may inadvertently cover unauthorized domains as well. Here, the unauthorized domain refers to a domain that shares the same task but exhibits different features compared to the authorized domain. This difference is primarily attributed to variations in the device, which in real cases is associated with the model owner rather than the user. This coverage has led to the emergence of freelancers who illegally steal well-trained deep models and deploy them on unauthorized domains (indicated by red squares in Fig. 1). They process the models using techniques such as domain adaptation [15], [16] or domain generation [17], [18] to obtain correct predictions and claim these models as their own products for profit. Such behavior infringes on the interests of model developers, and may even increase

- *L. Wang and M. Wang contributed equally to this work. Corresponding author: H. Fu (hzfu@ieee.org) and D. Zhang (dqzhang@nuaa.edu.cn).*
- *L. Wang and D. Zhang are with the Key Laboratory of Brain-Machine Intelligence Technology, Ministry of Education, Nanjing 211106, China.*
- *M. Wang and H. Fu are with the Institute of High Performance Computing (IHPC), Agency for Science, Technology and Research (A*STAR), 138632, Singapore.*

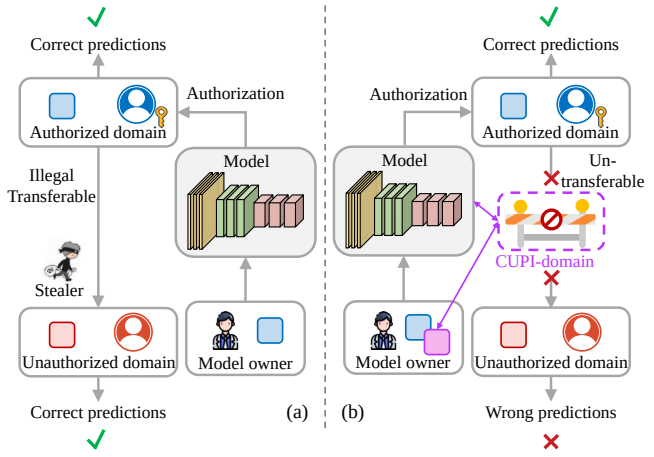

Fig. 1. Model IP protection with our proposed CUPI-Domain. (a): In traditional supervised learning (SL), owners train deep models on the authorized domain (blue square), and then authorized users can gain correct predictions on the authorized domain. However, freeloaders can illicitly steal these models and deploy them on unauthorized domains (red squares) to obtain usable results for profit. (b): Our method constructs a CUPI-Domain between the authorized and unauthorized domains, which could block the illegal transferring and lead to a wrong prediction for unauthorized domains.

the potential security risks on safety-critical applications, causing incalculable damage and hindering the long-term development of deep model [19]. Thus, protecting the rights of model owners and ensuring the safety of model deployment is essential.

Comprehensive model IP protection for deep models includes both ownership verification and applicability authorization [12], [13], [14], [20]. Ownership verification refers to "Who can use the model". Model owners can grant usage permission to specific users, and any other users will be infringing on the owner's IP rights when employing the model [12], [13]. However, it is important to acknowledge that there is no guarantee that the trained model will remain secure and not be leaked. There are two potential risks to consider: 1) Cyber hackers may attempt to steal the model from the cloud storage or other platforms where it is stored. 2) Specific users who have been granted permission may transfer the model to unauthorized or illegal users, which undermines the control and ownership of the model, potentially leading to misuse. These risks highlight the importance of implementing ownership verification mechanisms to protect against unauthorized access, sharing, and misuse of trained models. Common ownership verification mechanisms include embedding watermarks [21], [22], utilizing model fingerprints [23], or employing predefined triggers [24]. However, it is essential to recognize that these methods can be vulnerable to various techniques such as fine-tuning [25], classifier retraining [25], elastic weight consolidation algorithms [26], and watermark overwriting. These vulnerabilities can potentially weaken the effectiveness of the model's IP protection measures. Therefore, it is crucial to explore and develop more robust ownership verification techniques.

Applicability authorization refers to "Where can use the model". The model owner trains the deep model on the authorized domain and grants specific users the right to utilize the model within that domain, ensuring correct predictions [14], [20]. However, in practice, legal users and illegal users may employ techniques such as domain adaptive [15], [16], domain generalization [17], [18], etc., to process the model and utilize it on other domains with similar tasks [14], thereby achieving usable performance. This kind of infringement is not only easier to carry out but also more common and often concealed. This poses a significant challenge for maintaining control over the model's usage and protecting the IP of the model owner. Therefore, it is necessary to develop a more robust applicability authorization mechanism that restricts the model's performance to the tasks specified by the owner, ensuring that the model's performance on unauthorized domains is not superior to that of an untrained model. This will discourage unauthorized users from attempting to steal the model. To achieve this, a Non-Transferable Learning (NTL) method is proposed [14], which uses an estimator with a characteristic kernel from Reproducing Kernel Hilbert Spaces to approximate and increase the maximum mean difference between two distributions on finite samples. However, the authors only considered using limited samples to increase the mean distribution difference of features between domains and ignored outliers. The convergence region of NTL [14] is not tight enough. Moreover, the calculation of the maximum mean difference is class-independent, which reduces the model's feature recognition ability in the authorized domain to a certain extent.

To address these challenges, we propose a novel Compact Un-transferable Pyramid Isolation Domain (CUPI-Domain) which serves as a barrier against illegal transfers from authorized to unauthorized domains, as depicted in Fig 1 (b). Our proposed CUPI-Domain considers the overall features of each domain, comprising two components: shared features and private features. Shared features refer to semantic features, while private features include stylistic cues such as texture, saturation, perspective, brightness, background, etc. Initialized with the unauthorized domain, our proposed CUPI-Domain particularly emphasize the private features of the authorized domain, strategically fusing them with its semantic features, constructing richer style features for adaptively style augmentation. By intentionally diminishing the recognition capability of the CUPI-Domain, we can implicitly impede illegal transfers to unauthorized domains that possess irrelevant private style features, thereby leading to wrong predictions. Additionally, we design Domain-Information Memory Banks (DIMB) for storing and updating labeled pyramid features to obtain stable domain class features and domain class-wise style features. The novel style loss function is designed to effectively enhance the style difference between the authorized and unauthorized domains, whereas the discriminative loss functions are tailored to optimize the discriminative difference between the authorized/unauthorized domain and the CUPI-Domain. Moreover, we further develop two distinct solutions for leveraging the CUPI-Domain, depending on whether the unauthorized domain is known or not: target-specified CUPI-Domain and target-free CUPI-Domain.

In general, we highlight our five-fold contributions:

- **We propose a novel IP protection approach, named CUPI-Domain,** to block illegal transfers from autho-

rized to unauthorized domains. **Meanwhile, CUPI-Domain generators are designed** to generate labeled and style-rich CUPI-Domain by emphasizing the private features of the authorized domain, implicitly leading to failure in recognizing irrelevant private style features on unauthorized domains.

- **We introduce externel DIMB** to store specified features to obtain stable domain class features and domain class-wise style features for the subsequent computation.
- **Style and discriminative loss functions are designed** to further improve inter-domain differences while maintaining semantic consistency between the unauthorized domain and CUPI-Domain.
- **Two distinct solutions are developed for utilizing CUPI-Domain** based on whether the unauthorized domain is known or not: target-specified CUPI-Domain and target-free CUPI-Domain.
- **Finally, comprehensive experimental results on several public datasets demonstrate that CUPI-Domain effectively reduces the recognition ability on unauthorized domains while maintaining strong recognition on authorized domains.** As a plug-and-play module, our CUPI-Domain can be easily implemented within different backbones and provide efficient solutions.[1]

This paper is based on and extends our previous CVPR2023 version [27] in the following aspects. 1) We have implemented more standardized generators known as the **CUPI-Domain generators**. This generator effectively eliminates the style features of the input CUPI-Domain and subsequently integrates them with the style features of the authorized domain, resulting in enhanced similarity between the two. 2) We also designed external **DIMB**, which store and update labeled pyramid features to obtain stable domain class features and domain class-wise style features for the subsequent computation. 3) We designed novel **style loss function and discriminative loss functions** to improve the style difference and discriminative difference between authorized and unauthorized domains, while maintain the semantic consistency between the unauthorized domain and CUPI-Domain. 4) We conducted additional experiments on the **Office-home-65 (65 categories, 4 domains) [28], and DomainNet (345 categories, 6 domains) [29]** to demonstrate the continued effectiveness of our method as the data complexity increases. 5) We have invested substantial efforts in enhancing the presentation and organization of our paper, specifically focusing on the introduction, framework, key results, and discussion. We have provided more comprehensive explanations in the introduction section to offer a clearer understanding of the research context. Beside, we have introduced several new sections to elaborate on our novel framework, encompassing the method formulation, corresponding technical components, and loss functions. Additionally, we have diligently rewritten multiple sections to improve readability and provide more detailed explanations, particularly in the sections addressing quantitative comparisons, ablation experiments and discussion. These revisions aim to enhance the clarity and coherence of our paper, offering a more comprehensive and insightful reading experience.

1. https://github.com/LyWang12/CUPI-Domain.

2 Related Work

2.1 Model IP Protection

Currently, there are two primary categories of methods for protecting model IP: ownership verification and applicability authorization. In terms of ownership verification, watermark embedding [30] is a widely used classic method. Kuribayashi *et al.* [31] proposed a quantifiable watermark embedding method that aims to minimize the variations caused by embedding watermarks. Adi *et al.* [32] introduced a black-box tracking mechanism for ownership verification. Zhang *et al.* [33] proposed a model watermarking framework that utilizes spatial invisible watermarking to protect image processing models. However, it has been observed that watermark embedding approaches are vulnerable to certain watermark removal methods. Our experiments employ a simple watermark embedding in the model to verify ownership by triggering misclassification. Through comprehensive experimental results, we demonstrate that the proposed CUPI-Domain exhibits resistance against common methods of watermark removal.

Applicability authorization is an extension of usage authorization, where model owners typically employ a preset private key to encrypt the entire or partial network. This encryption ensures that only authorized users can obtain the private key and subsequently use the model. Various advanced methods have been developed for usage authorization. For instance, Alam *et al.* [34] introduced an explicit locking mechanism that utilizes S-Boxes with strong cryptographic properties to secure each training parameter of a lightweight deep neural network. Unauthorized access without knowledge of the legitimate private key can significantly degrade the model's accuracy. Song *et al.* [35] analyzed and calculated the critical weight of deep neural network models and reduced time costs by encrypting these critical weights to protect against unauthorized use. In terms of applicability authorization, Wang *et al.* [14] proposed a data-based approach called Non-Transferable Learning (NTL), which aims to preserve model performance on authorized data while intentionally degrading performance in other data domains . This method allows the model to act well within authorized domains while exhibiting reduced effectiveness when applied to unauthorized domains.

In contrast to these methods, we propose a novel approach that involves constructing a new class-dependent CUPI-Domain with infinite samples. This CUPI-Domain is designed to have features that are more similar to the authorized domain. By deliberately reducing the model's performance on both the CUPI-Domain and the target domain, we can achieve a tighter generalization bound for the well-trained model, thereby constraining the model's performance within the authorized domain.

2.2 Domain Transferring

In recent years, deep learning models have achieved revolutionary success in diverse areas of machine learning. To inherit the advantages of deep models, domain adaptive [15], [16] and domain generalization [17], [18] have

been proposed to rapidly improve the performance of well-trained deep models on target datasets.

Domain adaptive [15], [16] involves transferring a model from a labeled source domain to an unlabeled but relevant target domain, with the target domain's data being accessible during the training process [36]. Sun *et al.* [37] and Zhuo *et al.* [38] minimized inter-domain differences by aligning the second-order statistics of the source and target distributions. Saito *et al.* [39] adopted a new adversarial paradigm, where the adversarial way occurs between the feature extractor and the classifier, rather than between the feature extractor and the domain discriminator.

Domain generalization [17], [18] differs from domain adaptive in that the target domain is inaccessible during training phase [40], [41]. Tobin *et al.* [40] used domain randomization to generate more training data from simulated environments for generalization in real environments. Prakash *et al.* [41] further considered the structure of the scene when randomly placing objects for data generation, enabling the neural network to learn how to utilize context when detecting objects. Recently, several methods have been proposed and effectively applied to cross-domain applications. These methods aim to find an intermediate state that lies between the source domain and the target domain. By emphasizing the similarity between domains, they enhance the transferability of the model [42], [43], [44], [45], [46].

In contrast, the objective of our proposed CUPI-Domain is to explore an intermediate state that accentuates the distinctions between the authorized and unauthorized domains, thereby limiting the transferability of the model and ensuring the protection of the model owners' intellectual property with respect to their scientific and technological achievements.

3 Methodology

We first provide a comprehensive explanation of our proposed CUPI-Domain in Section 3.1, which serves as a barrier to restrict the model's performance within the authorized domain and diminish its feature recognition capabilities on unauthorized domains. Subsequently, we introduce external DIMB for storing and updating labeled pyramid features in Section 3.2. Additionally, we present two distinct solutions based on whether the unauthorized target domain is known or not: the target-specified CUPI-Domain and the target-free CUPI-Domain in section 3.3, these solutions are designed to safeguard the IP of the model.

3.1 Compact Un-transferable Pyramid Isolation Domain (CUPI-Domain)

In the deep neural network model, the overall features extracted by the feature extractor include two abstract components, *i.e.*, shared features, and private features [47], [48]. Shared features refer to semantic features which reflect the structural information of samples and play a leading role in sample recognition [49]. Private features refer to a collection of subtle, weak semantically related cues in the feature, such as lighting conditions, texture, hue, color saturation, and background [50]. Samples from different domains with the same task usually exhibit consistent semantic information while significant stylistic variations.

Most previous works on domain adaptive and domain generalization have primarily focused on enhancing feature transferability between domains. This has involved strengthening the model's emphasis on shared features while suppressing private style features that may appear as disturbances. However, to ensure the protection of the model's IP, this paper aims to restrict the model's feature recognition capability by accentuating the private style features of the authorized domain through style transfer techniques. This emphasis leads to failure in recognizing irrelevant private style features on unauthorized domains.

Style transfer studies have conjectured that styles exhibit homogeneity and consist of repeated structural motifs. In this context, two images are considered to have similar styles if the features extracted by a trained classifier exhibit shared statistics [50], [51], such as first- and second-order statistics, which are commonly used due to their computational efficiency. First- and second-order statistics refer to the mean and variance of the extracted features, which are called style features. Semantic features capture pure structural information without incorporating style cues. Following Dumoulin *et al.* [49], the semantic features f_{se} of the extracted feature f can be obtained by removing style features as:

$$f_{se} = \frac{f - \mu(f)}{\sigma(f)}, \tag{1}$$

where $\mu(f)$ and $\sigma(f)$ denote the mean and variance of f. Furthermore, style can be re-assign by $f \cdot \gamma + \beta$, where γ and β are learned parameters. Afterward, Huang *et al.* [50] further explored adapting f to an arbitrarily given style by using style features of another extracted feature instead of learned parameters.

Building upon these concepts, we present a novel CUPI-Domain that combines style features from the authorized domain with semantic features from the CUPI-Domain. We achieve this by initializing the CUPI-Domain with the unauthorized domain and then randomly extracting style features from the authorized domain, integrating them with the semantic features of the CUPI-Domain. This process results in the private style features of the CUPI-Domain becoming more similar to those of the authorized domain. By employing this strategy, we effectively diminish the model's feature recognition capabilities on both the CUPI-Domain and the unauthorized domain. Consequently, we implicitly obstruct the pathway between the authorized and unauthorized domains, thereby confining the model's performance exclusively to the authorized domain.

Our CUPI-Domain is generated by the CUPI-Domain generator, which is a lightweight, and plug-and-play module, as depicted in Fig. 2. f_s^l and f_i^l represent the deep features of the l-th feature extractor block in the authorized domain and the CUPI-Domain, respectively. To begin, f_s^l and f_i^l are concurrently fed into the CUPI-Domain generator. Subsequently, the mean $\mu(f_s^l)$ and variance $\sigma(f_s^l)$ of f_s^l are calculated along the channel dimension to represent private style features of the authorized domain, followed by a 1×1 convolution layer $Conv$. Next, removing style features of f_i^l. Finally, the $\mu(f_s^l)$ and variance $\sigma(f_s^l)$ of f_s^l are multiplied and added channel-wisely as:

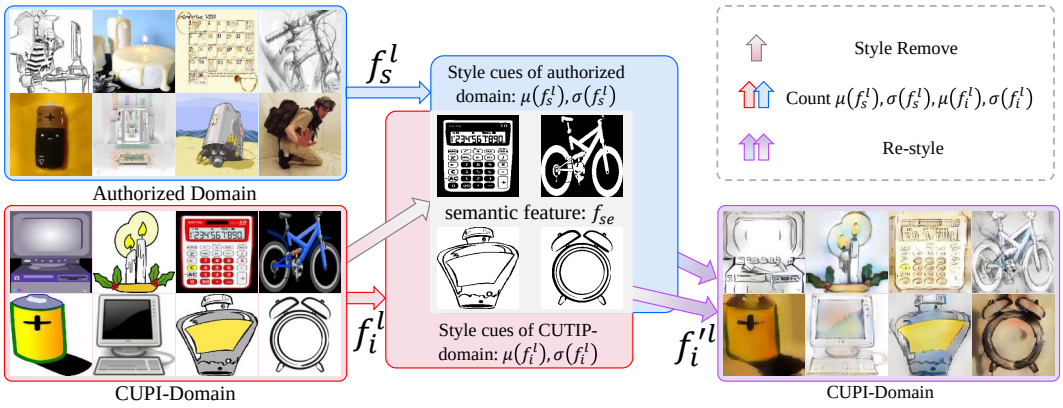

Fig. 2. Illustration of our proposed CUPI-Domain generator. The output feature of the authorized domain f_s^l and the CUPI-Domain f_i^l in the l-th feature extractor block are fed into the CUPI-Domain generator, and then the style features of f_s^l are combined with the semantic features of f_i^l to update the CUPI-Domain. This update process ensures that the private style features of the updated f'^l_i are more similar to the features of the authorized domain f_s^l while preserving their original semantic features.

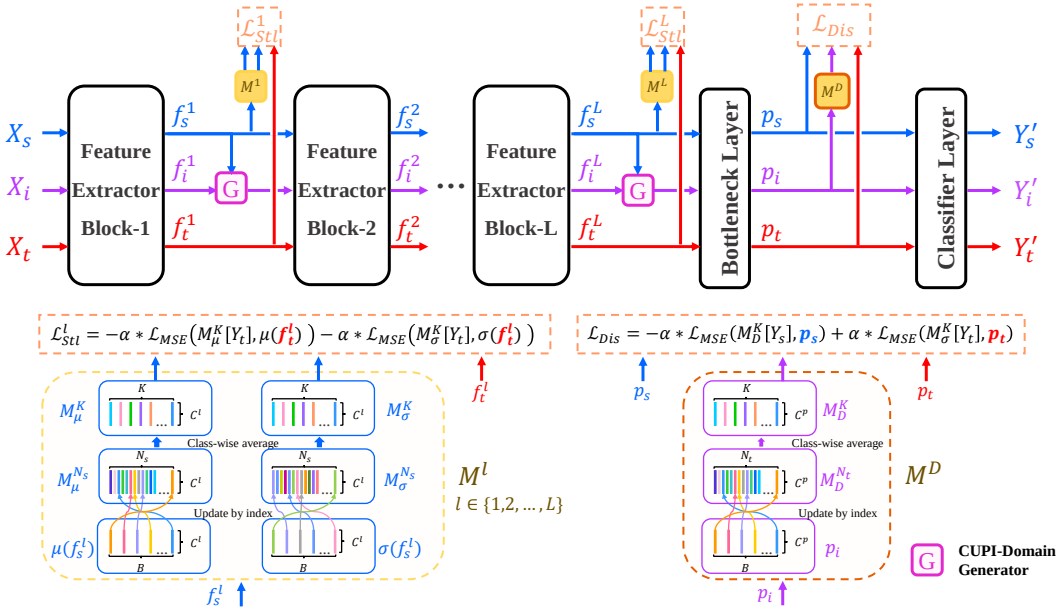

Fig. 3. The illustration of our proposed CUPI-Domain, including the feature extractor blocks, CUPI-Domain generators, the external DIMB, and style/distinctive loss functions. The samples from the authorized domain, CUPI-Domain, and unauthorized domain are fed into the feature extractor in parallel, denoted by blue, purple, and red, respectively. The CUPI-Domain generators are deployed after each feature extractor block, while the external DIMB are designed to store specified labeled features to obtain stable domain class features and domain class-wise style features for subsequent computation.

$$f'^l_i = \frac{f_i^l - \mu(f_i^l)}{\sigma(f_i^l)} \bigotimes Conv(\sigma(f_s^l)) \bigoplus Conv(\mu(f_s^l)). \quad (2)$$

Subsequently, f_s^l, f'^l_i, and f_t^l are simultaneously fed into the $l+1$-th feature extractor block in parallel. Through continuous feature fusion, as shown in Fig. 2, CUPI-Domain generator can build style-rich latent feature space for each class and then construct a labeled CUPI-Domain containing similar private styles to the authorized domain.

3.2 Domain-Information Memory Banks (DIMB)

In our method, we employ external DIMB comprising multiple memory cells to store and update domain-dependent pyramid features via sample index. This enables us to obtain stable domain class features and domain class-wise style features throughout the training process, as depicted at the bottom of Fig. 3. In this section, we provide a comprehensive description of the workflow associated with our proposed DIMB.

3.2.1 *Initialization Strategy of DIMB*

Before training, the pre-trained feature extractor with frozen model parameters is utilized to initialize the DIMB as follows:

- Since the output features of the l-th feature extractor $f_s^l \in \mathbb{R}^{B\times N^l\times C^l}$ in the authorized domain contain rich semantic information with style clues, and feature vector $p_i \in \mathbb{R}^{B\times C^p}$ before the classifier layer contains refined class-discriminative information in the CUPI-Domain, we store f_s^l and p_i in the memory cell M^l

and M^D respectively. Where B denotes the batch size of the input, N^l represents the output size of the l-th feature extractor, C^l and C^p denotes the feature dimension of the l-th feature extractor and bottleneck layer, respectively.

- Calculate the channel-wise mean $\mu\left(f_s^l\right) \in \mathbb{R}^{B\times C^l}$ and variance $\sigma\left(f_s^l\right) \in \mathbb{R}^{B\times C^l}$ of f_s^l, and update the corresponding B features in $M_\mu^{N_s} \in \mathbb{R}^{N_s\times C^l}$ and $M_\sigma^{N_s} \in \mathbb{R}^{N_s\times C^l}$ according to the sample index, meanwhile, do the same with p_i to update $M_D^{N_t} \in \mathbb{R}^{N_t\times C^p}$. Where N_s and N_t denote the number of samples in the authorized domain and the unauthorized domain respectively.
- Repeat the above steps until all samples in the authorized domain have been traversed. Calculate the class centers of all features in $\left\{M_\mu^{N_s}, M_\sigma^{N_s}, M_D^{N_t}\right\}$ according to their class labels and store the result in $\left\{M_\mu^K \in \mathbb{R}^{K\times C^l}, M_\sigma^K \in \mathbb{R}^{K\times C^l}, M_D^K \in \mathbb{R}^{K\times C^p}\right\}$, respective, where K denote the class number. Finally, unfreeze the model parameters.

3.2.2 Update Strategy of DIMB

During training, we directly discard those obsolete features f_s^l and p_i according to the index of the samples and replace them with the latest features, then calculate and update the corresponding features in memory cell $M^l = \left\{M_\mu^{N_s}, M_\sigma^{N_s}, M_\mu^K, M_\sigma^K\right\}(l \in 1, 2, \ldots, L)$ and $M^D = \left\{M_D^{N_t}, M_D^K\right\}$ as in the initialization strategy. Among them, M_D^K stores K class features, while M_μ^K and M_σ^K store style features of K classes, which are used for downstream computation. Furthermore, the proposed DIMB is only an external repository and does not participate in the back-propagation calculation of the network.

3.3 Model IP Protection with CUPI-Domain

3.3.1 Target-Specified CUPI-Domain

We first introduce our solutions for model IP protection with a given unauthorized target domain, termed target-specified CUPI-Domain.

Fig. 3 provides an overview of the entire framework trained using our proposed CUPI-Domain. The framework comprises L feature extractor blocks, L CUPI-Domain generators, $L+1$ external DIMB, a bottleneck layer, and a classifier layer. The data from the authorized domain, CUPI-Domain, and unauthorized domain are denoted as X_s, X_i (Initialized by X_t), and X_t, respectively. During the training process, X_s, X_i, and X_t are simultaneously fed into the feature extractor blocks, followed by a CUPI-Domain generator. The classifier at the end of the network is responsible for predicting the category of the sample, and the corresponding prediction results are represented as Y_s', Y_i', and Y_t', respectively.

Based on the designed framework and DIMB, we proposed a novel joint loss function that incorporates the classification loss $\mathcal{L}_{cls}$, style loss $\mathcal{L}_{Stl}$ and discriminative loss $\mathcal{L}_{Dis}$ to enhance the distinction between authorized and unauthorized domains, while maintaining the classification performance of authorized domain. The overall loss function $\mathcal{L}$ is as follows:

$$\mathcal{L} = \mathcal{L}_{cls} + \mathcal{L}_{Stl} + \mathcal{L}_{Dis}. \tag{3}$$

The classification loss function $\mathcal{L}_{cls}$ is devised to enhance the class recognition capability within the authorized domain while reducing the class recognition ability of the CUPI-Domain and unauthorized domain, which consists of the respective domain loss functions of the authorized domain $\mathcal{L}_s$, CUPI-Domain $\mathcal{L}_i$ and unauthorized domain $\mathcal{L}_t$, as:

$$\mathcal{L}_{cls} = \mathcal{L}_s - \mathcal{L}_i - \mathcal{L}_t. \tag{4}$$

For all three domains, the network can be trained using the Kullback-Leibler divergence loss function, which serves as a measure of dissimilarity between the predicted distribution and the target distribution:

$$\mathcal{L}_s = \alpha * KL(Y_s'||Y_s), \tag{5}$$
$$\mathcal{L}_i = \alpha * KL(Y_i'||Y_i), \tag{6}$$
$$\mathcal{L}_t = \alpha * KL(Y_t'||Y_t), \tag{7}$$

where $KL(\cdot)$ stands for Kullback–Leibler divergence, Y_s, Y_i and Y_t denote the label of the authorized domain, CUPI-Domain and the unauthorized domain, respectively. The rising factor $\alpha = \left(\frac{epoch\_number}{total\_epoch}\right)^{0.9}$ is used to speed up the convergence of the model on the authorized domain in the early stage of training. Additionally, to mitigate the impact of infinite amplification, an upper bound is set to limit the negative loss function, as suggested by Wang *et al.* [14].

The style loss function $\mathcal{L}_{Stl}$ is designed based on pyramid DIMB to enhance the distinction in style between authorized and other domains. It achieves this by incorporating stable multi-scale authorized domain style information, and real-time multi-scale style information of CUPI-Domain and unauthorized domain:

$$\begin{aligned}\mathcal{L}_{Stl} = &-\alpha * \sum_{l=1}^{L} \mathcal{L}_{MSE}(M_\mu^{(l)K}[Y_t], \mu(f_t^l)) \\ &- \alpha * \sum_{l=1}^{L} \mathcal{L}_{MSE}(M_\sigma^{(l)K}[Y_t], \sigma(f_t^l)).\end{aligned} \tag{8}$$

Where $\mathcal{L}_{MSE}$ denotes the mean squared error. The discriminative loss $\mathcal{L}_{Dis}$ is used to maintain the semantic consistency between the restyled CUPI-Domain and the unauthorized domain while enhancing the difference between CUPI-Domain and the authorized domain, which is defined as follows:

$$\mathcal{L}_{Dis} = -\alpha * \mathcal{L}_{MSE}(M_D^K[Y_s], p_s) + \alpha * \mathcal{L}_{MSE}(M_D^K[Y_t], p_t). \tag{9}$$

We summarize the strategy of our proposed target-specified CUPI-Domain in Supplementary Algorithm 1.

3.3.2 Target-Free CUPI-Domain

To tackle scenarios where the unauthorized domain is unknown, we introduce the Target-free CUPI-Domain. In such cases, it is not feasible to directly incorporate the unauthorized domain and CUPI-Domain into the model training process. To address this challenge, a synthesized unauthorized domain can be employed instead. For instance, Wang *et al.* [14] proposed a GAN-based approach that freezes parameters to generate synthesized samples in various directions as a substitute for the unauthorized domain training set. Although their method is capable of

producing high-quality synthesized samples, the generator's direction is predetermined, leading to certain limitations. Additionally, Huang *et al.* [50] developed an adaptive instance normalization (AdaIN) technique based on GAN, which can generate synthesized images with a specific style to complement a given content image.

To fully leverage the benefit, we introduce Gaussian noise to the AdaIN technique, enabling the generation of synthesized samples with random styles. Finally, we combine the synthesized samples obtained by the two aforementioned methods (*e.g.*, GAN and AdaIN) to replace the unauthorized domain training set and initialize the CUPI-Domain.

It is important to emphasize that our focus is on evaluating the effectiveness of the CUPI-Domain in preventing unauthorized feature transfer specifically within the context of synthesized images. Our primary focus remains on reducing the feature recognition ability of the model on unauthorized domains while preserving it on the authorized domain. Any data synthesis method could be utilized in our work for generating the unauthorized domain.

The framework of the Target-free CUPI-Domain is consistent with the target-specified CUPI-Domain and the training process is detailed in Supplementary Algorithm 2.

During the testing phase, the model is evaluated on the authorized domain test set and other unknown domains with the same task.

4 Experiment

4.1 Datasets

We evaluate our proposed CUPI-Domain on several popular domain adaption/generation benchmarks:

1) **Digit datasets: MNIST [52], USPS [53], SVHN [54] and MNIST-M [55]** are widely used digit datasets, each consisting of ten digits ranging from 0 to 9. These datasets include samples extracted from various scenes, providing a diverse range of digit images for evaluation and analysis.
2) **CIFAR10 [56] and STL10 [57]** are both ten-class classification datasets commonly used in computer vision research. To ensure consistency between the datasets, we adopt the processing procedure outlined by French *et al.* [58]. This ensures that the classes in both datasets are aligned and can be directly compared during experimentation and evaluation.
3) **VisDA-2017 [59]** is a Synthetic-to-Real dataset that consists of training (VisDA-T) and validation (VisDA-V) sets, each containing samples from 12 different categories.
4) **Office-Home-65 [28]** consists of images from four distinct domains: Artistic, Clipart, Product, and Real-World. Each domain contains 65 object categories, resulting in a total of 15,500 images in the dataset.
5) **DomainNet [29]** consists of images from six domains, including clipart, infograph, painting, quickdraw, real and sketch, with a total of 345 scene-level categories.

4.2 Implementation Details

The implementation of our comprehensive experiments is based on the platform Pytorch and an NVIDIA GeForce RTX 3090 GPU with 24GB of memory. The Adam optimizer with an initial learning rate of 0.0001 is adopted to optimize the model. The batch size for each domain and the epoch number is set to 32 and 30, respectively. To accommodate tasks of varying complexity, we employ different backbone architectures for various datasets followed by Wang *et al.* [14] for fair comparison. Specifically, we used VGG-11 [7] for digit datasets, VGG-19 [7] for CIFAR10 & STL10 and VisDA-2017 [59], and SWIN [9] for Office-Home-65 [28] and DomainNet [29]. Pre-trained models are used for a fair comparison. We set L to 4 for SWIN and 5 for VGG to align with the number of feature extractor blocks in each architecture. Consistent with standard evaluation protocols, we utilize accuracy (%) as the primary performance metric for each task.

4.3 Result of Target-Specified CUPI-Domain

We conducted experiments on digital datasets, as depicted in Table 1. The vertical axis represents the authorized domain, while the horizontal axis represents the unauthorized domain. We explored 16 transfer tasks by selecting various combinations of authorized and unauthorized domains from the four domains of the digital datasets. In each task, the data on the left of '$\Rightarrow$' presents the test accuracy of supervised learning (SL) on the unauthorized domain, while the data on right of '$\Rightarrow$' presents the test accuracy of CUPI-Domain on the unauthorized domain. 'Authorized/Unauthorized Drop' indicates the drop (relative drop) of CUPI-Domain relative to SL in authorized/unauthorized domains. Table 2 displays the 'Authorized/Unauthorized Drop' of CUPI-Domain, CUTI-Domain [27], and NTL [14]. From the results, we observe that the average drop of the CUPI-Domain on the unauthorized domain is 56.83 (86.89%), while the drop on the authorized domain is only 0.02 (0.02%). Comparatively, CUTI-Domain [27] and NTL [14] exhibit lower average performance degradations on the unauthorized domain but higher on the authorized domain. We further conduct statistical significance tests achieved by different methods, denoted with *(CUPI-Domain vs. CUTI-Domain [27]) and $^\diamond$(CUPI-Domain vs. NTL [14]). Our proposed CUPI-Domain shows a statistical difference ($p < 0.05$) over CUTI-Domain [27] and NTL [14]. These results indicate that CUPI-Domain effectively reduces the sample recognition ability of the model for the unauthorized domain while having minimal impact on the authorized domain.

Fig. 4 illustrates the performance on three different transfer tasks: CIFAR10 $\rightarrow$ STL10, STL10 $\rightarrow$ CIFAR10 and VisDA-T $\rightarrow$ VisDA-V. Each subgraph displays the accuracy of the corresponding method in the authorized domain, the accuracy in the unauthorized domain, and their drop (relative drop). SL achieves the highest classification accuracy on the unauthorized domain due to its wide generalization region. In contrast, the other three methods all reduced accuracy in the unauthorized domain, indicating the success of blocking the transfer pathway. Comparatively, CUPI-Domain shows higher degradation than others, demonstrating its superior ability to compress the model's generalization region.

To showcase the continued effectiveness of our method under increased numbers of categories and samples, we

TABLE 1
The accuracy (%) of target-specified CUPI-Domain on digit datasets. The vertical/horizontal axis denotes the authorized/unauthorized domain. In each task, the left of '⇒' shows the test accuracy of SL on the unauthorized domain, while the right side presents the accuracy of CUPI-Domain. 'Authorized/Unauthorized Drop' indicate the drop (relative drop) of CUPI-Domain relative to SL on authorized/unauthorized domains.

Authorized/Unauthorized	MNIST	USPS	SVHN	MNIST-M	Authorized Drop↓	Unauthorized Drop↑
MNIST [52]	99.2 ⇒ 99.2	98.0 ⇒ 6.5	38.2 ⇒ 6.0	67.8 ⇒ 8.5	0.00 (0.00%)	61.00 (88.37%)
USPS [53]	92.6 ⇒ 9.1	99.7 ⇒ 99.7	25.5 ⇒ 5.4	41.2 ⇒ 8.1	0.00 (0.00%)	45.57 (83.11%)
SVHN [54]	66.7 ⇒ 9.5	70.5 ⇒ 6.6	91.2 ⇒ 91.1	34.6 ⇒ 7.8	0.07 (0.08%)	49.30 (84.62%)
MNIST-M [55]	98.4 ⇒ 6.4	88.4 ⇒ 7.4	46.3 ⇒ 5.0	95.4 ⇒ 95.4	0.00 (0.00%)	71.43 (91.44%)

TABLE 2
'Authorized/Unauthorized Drop' of target-specified CUPI-Domain, CUTI-Domain [27] and NTL [14] on digit datasets. The best performance is indicated by the numbers in bold. Statistical significance (p-value < 0.05 [60], [61]) is denoted with: *(CUPI-Domain vs. CUTI-Domain [27]) and $^\diamond$(CUPI-Domain vs. NTL [14]).

Domain / Drop	Authorized Drop↓			Unauthorized Drop↑		
	CUPI-Domain	CUTI-Domain [27]	NTL [14]	CUPI-Domain	CUTI-Domain [27]	NTL [14]
MNIST [52]	**0.00 (0.00%)**	0.10 (0.10%)	1.00 (1.01%)	**61.00 (88.37%)**	**61.00 (88.56%)**	46.57 (75.60%)
USPS [53]	**0.00 (0.00%)**	0.10 (0.10%)	1.00 (1.00%)	**45.57 (83.11%)**	44.70 (80.72%)	38.67 (75.55%)
SVHN [54]	**0.07 (0.08%)**	0.30 (0.33%)	1.10 (1.23%)	**49.30 (84.62%)**	48.33 (81.73%)	40.60 (77.25%)
MNIST-M [55]	**0.00 (0.00%)**	0.00 (0.00%)	2.10 (2.30%)	**71.43 (91.44%)**	69.73 (88.75%)	60.10 (76.95%)
Mean	**0.02 (0.02%)**$^{*\diamond}$	0.13 (0.13%)	1.30 (1.39%)	**56.83 (86.89%)**$^{*\diamond}$	55.94 (84.94%)	46.48 (76.34%)

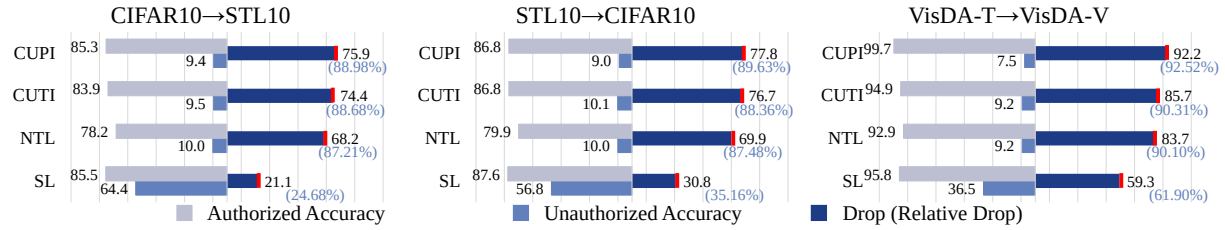

Fig. 4. The accuracy (%) of SL, target-specified NTL [14], CUTI-Domain [27], and CUPI-Domain on CIFAR10&STL10, and VisDA-2017. The title in each subgraph implies the authorized domain (left of '→') and the unauthorized domain (right of '→'). The bars with different colors represent the accuracy of each method in the authorized domain, the unauthorized domain, and the drop (relative drop) of the model performance, respectively.

TABLE 3
The average results of ownership verification by SL and CUPI-Domain, CUTI-Domain [27] and NTL [14]. 'Training Methods' shows SL and CUPI-Domain accuracy on the authorized domain without/with watermark. 'Watermark Removal Approaches on CUPI' represents CUPI-Domain accuracy after/before attacks by watermark removal methods: FTAL [25], RTAL [25], EWC [26], AU [26], and watermark overwriting followed by NTL [14]. 'Average Drop' represents the average decrease in test dataset accuracy with a watermark compared to without, after applying watermark removal methods. The best performance is indicated by the numbers in bold.

Source without Patch	Training Methods (%)		Watermark Removal Approaches on CUPI (%)					Average Drop↑		
	SL	CUPI	FTAL [25]	RTAL [25]	EWC [26]	AU [26]	Overwriting	CUPI	CUTI [27]	NTL [14]
	[Test w/o Watermark (%)]		[Test w/o Watermark (%)]							
Digit [52], [53], [54], [55]	96.7 / 96.9	8.9 / 96.0	8.6 / 98.0	11.3 / 97.1	10.2 / 100.0	8.7 / 100.0	9.8 / 97.5	**88.8**	87.5	82.6
CIFAR10 & STL10 [56], [57]	85.6 / 84.2	10.6 / 86.6	11.1 / 98.2	16.0 / 97.7	14.2 / 100.0	11.6 / 97.2	14.1 / 92.4	**83.7**	77.8	74.3
VisDA-2017 [59]	93.6 / 92.2	8.6 / 95.2	19.9 / 100.0	22.2 / 98.6	14.9 / 100.0	9.0 / 100.0	21.5 / 98.8	**82.0**	78.0	76.8
Office-Home-65 [28]	75.5 / 83.4	0.8 / 83.1	1.3 / 99.5	1.8 / 99.5	1.7 / 99.8	2.1 / 99.6	3.4 / 99.7	**97.6**	94.5	94.1
DomainNet [29]	39.0 / 45.4	0.4 / 44.7	3.2 / 63.6	5.7 / 59.3	0.8 / 63.6	0.6 / 62.9	3.2 / 63.9	**60.0**	52.2	47.3

conducted experiments on Office-Home-65 [28] and DomainNet [29], as shown in Supplementary Table 1, 2, 3 and 4. As the data complexity increases, we observed a decrease in the transfer performance of SL on the unauthorized domain (left of the ⇒) compared to the results presented in Table 1. This decline poses a challenge to further reduce the accuracy of the unauthorized domain. However, our proposed CUPI-Domain demonstrates consistent effectiveness, with relative degradation of 83.41% and 76.14% on Office-Home-65 [28] and DomainNet [29], respectively, exhibiting significantly better performance compared to CUTI-Domain [27]. Despite the fact that NTL [14] exhibits a relatively higher degradation on DomainNet [29], its performance experiences a significant drop of 16.06 (35.06%) in the authorized domain, which is deemed unacceptable. Besides, our CUPI-Domain achieve statistical difference ($p < 0.05$) over CUTI-

Domain [27] and NTL [14]. This phenomenon can be attributed to the fact that the calculation of its maximum mean difference is class-independent, thereby reducing the model's feature recognition capability in the authorized domain to some extent, especially when the complexity of the dataset increases.

4.4 Result of Ownership Verification

In this section, we focus on conducting ownership verification of the model by intentionally triggering classification errors. To this end, we apply a regular backdoor-based model watermark patch to the authorized domain dataset, followed by NTL [14]. This processed authorized domain is then treated as the new unauthorized domain. The accuracy of SL and CUPI-Domain on the authorized domain with/without the watermark patch is presented in Table 3 and Supplemeatary Table 5.

Comparing the second and third columns of the table, we can observe that SL shows minimal disparity in accuracy with and with the watermark patch, indicating a lack of sensitivity to the watermark patch. Conversely, CUPI-Domain exhibits a significant reduction in accuracy on the unauthorized domain with watermark embedding. This performance difference can be effectively utilized for model ownership verification.

Additionally, we evaluate the robustness of CUPI-Domain against five state-of-the-art model watermark removal methods, including FTAL [25], RTAL [25], EWC [26], AU [26], and watermark overwriting followed by NTL [14]. The last three columns depict the drop in accuracy between data with and without the watermark. Notably, CUPI-Domain, CUTI [27], and NTL [14] demonstrate effective resistance against watermark removal methods, with CUPI-Domain outperforming CUTI [27] and NTL [14] by approximately 4.7% and 8.2% in terms of performance respectively, and also demonstrating statistical difference ($p < 0.05$).

4.5 Result of Target-free CUPI-Domain

In Section 3.3.2, we proposed the target-free CUPI-Domain to address the scenario where the unauthorized domain is unknown by leveraging synthesized samples as a substitute for the unauthorized domain training set. To evaluate the performance of our proposed target-free CUPI-Domain on digit datasets, we conducted 4 transfer tasks. For each task, we selected one authorized domain from the digit datasets and used the remaining unknown domains as the unauthorized domain testing sets. We report the 'Authorized/Unauthorized Drop' metric and statistical difference ($p < 0.05$), as presented in Table 4 (with more details in Supplementary Table 6).

Analyzing the results, we observe that CUPI-Domain exhibits the highest average drop on the test sets, with a degradation of 55.10 (83.77%). Additionally, CUPI-Domain demonstrates a lower drop in the unauthorized domain in comparison to CUTI-Domain [27] and NTL [14], further highlighting its enhanced IP protection capability.

Fig. 5 visually presents the outcomes for CIFAR10 $\rightarrow$ STL10, STL10 $\rightarrow$ CIFAR10 and VisDA-T $\rightarrow$ VisDA-V. CUPI-Domain, CUTI [27], and NTL [14] exhibit higher degradation than SL consistent with the findings discussed earlier. Remarkably, CUPI-Domain demonstrates the highest degradation, indicating its superior ability to compress the model's generalization region. More experimental results on Office-Home-65 [28] and DomainNet [29] are detailed in Supplementary Table 7, 8, 9, and 10.

4.6 Result of Applicability Authorization

To assess the applicability authorization, we conduct experiments by constraining its generalization capabilities solely to the authorized domain. We adopted three evaluation schemes: 1) We introduce a specified watermark patch to the authorized domain (same as in Section 4.4), creating a new authorized domain training set. Subsequently, we combine the authorized domain, synthesized domain, and synthesized domain with the specified watermark patch to construct the unauthorized domain training set. 2) We use clean authorized data without a watermark patch as an authorized domain training set and then combine the authorized domain with a watermark patch, synthesized domain, and synthesized domains with a watermark patch to construct the unauthorized domain training set. 3) We consider clean authorized data without a watermark patch as an authorized domain training set and the synthesized domain as the unauthorized domain training set. During the testing phase, we evaluated the model's performance on several unknown domains, both with and without specified watermark patches.

The experimental results are reported in Table 5, with additional details available in Supplementary Table 11, 12, 13, 14, 15 and 16. The results of scheme 1 indicate that the proposed CUPI-Domain performs well on the authorized domain with the specific watermark patch but exhibits low accuracy on other unknown domains. In scheme 2, the proposed CUPI-Domain performs well only on the authorized domain without the watermark patch. When the watermark is considered insufficient to change the domain in scheme 3, the CUPI-Domain performs effectively on authorized domains, regardless of the presence of watermarks. This aligns with our expectation that the model's generalization ability is successfully constrained within the authorized domain, regardless of the presence or absence of the watermark patch.

Furthermore, in scheme 1/2/3, our proposed CUPI-Domain achieves an average drop of 88.57 (88.71%) / 85.50 (85.71%) / 86.36 (87.16%), surpassing CUTI's drop (relative drop) of 83.75 (84.27%) / 85.85 (83.27%) / 84.00 (85.05%) and NTL's drop (relative drop) of 81.03 (81.81%) / 80.75 (81.28%) / 81.74 (82.88%) with a statistical difference ($p < 0.05$). This improvement is attributed to the construction of the CUPI-Domain, where we leverage infinite samples that closely resemble the authorized domain to create a more compact generalization boundary. Consequently, CUPI-Domain exhibits a stronger ability to protect IP compared to NTL [14], which relies solely on limited features from the source and target domains for distance maximization.

4.7 Ablation Study of Components

To gain further insights into the effectiveness of each component in our proposed CUPI-Domain, we conduct an ablation study on five random tasks, as detailed in Table 6. The vertical axis represents experimental settings with different

TABLE 4
'Authorized/Unauthorized Drop' of target-free CUPI-Domain, CUTI-Domain [27] and NTL [14] on digit datasets.

Domain / Drop	Authorized Drop↓			Unauthorized Drop↑		
	CUPI-Domain	CUTI-Domain [27]	NTL [14]	CUPI-Domain	CUTI-Domain [27]	NTL [14]
MNIST [52]	**0.03 (0.03%)**	0.40 (0.40%)	0.70 (0.71%)	**60.43 (87.39%)**	59.17 (85.43%)	57.30 (84.06%)
USPS [53]	**0.33 (0.33%)**	0.60 (0.60%)	0.60 (0.60%)	**45.29 (81.80%)**	44.97 (80.96%)	42.90 (74.71%)
SVHN [54]	**1.10 (1.20%)**	2.50 (2.74%)	3.20 (3.51%)	**44.87 (76.53%)**	44.00 (74.19%)	41.53 (64.09%)
MNIST-M [55]	**0.19 (0.20%)**	0.30 (0.31%)	2.00 (2.10%)	**69.82 (89.36%)**	65.73 (83.96%)	63.03 (77.62%)
Mean	**0.41 (0.44%)**$^{*\diamond}$	0.95 (1.02%)	1.63 (1.73%)	**55.10 (83.77%)**$^{*\diamond}$	53.47 (81.13%)	51.19 (75.12%)

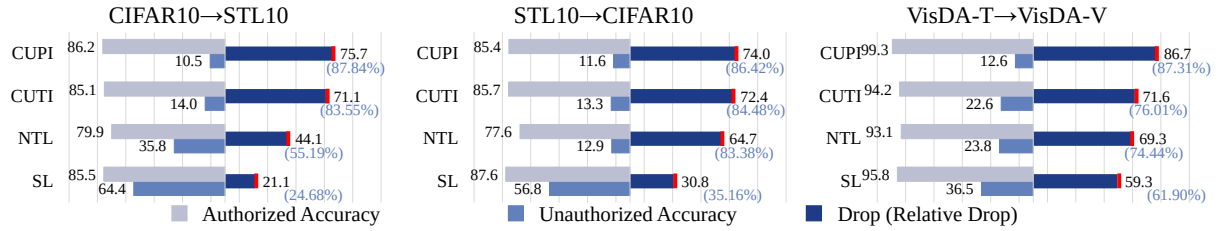

Fig. 5. The accuracy (%) of SL, target-free NTL [14], CUTI-Domain [27] and CUPI-Domain on CIFAR10&STL10, and VisDA-2017. The title in each subgraph implies the authorized domain (left of '→') and the unauthorized domain (right of '→'). The bars with different colors represent the accuracy of each method in the authorized domain, the unauthorized domain, and the drop (relative drop) of the model performance, respectively.

TABLE 5
The average 'Authorized Domain', 'Other Domains' and 'Drop' in applicability authorization scheme 1/2/3 of CUPI-Domain, CUTI-Domain [27], and NTL [14] on the digital datasets, CIFAR10&STL10 and VisDA-2017.

Scheme	Authorized Doamin ↑			Other Domains↓			Drop↑		
	CUPI	CUTI [27]	NTL [14]	CUPI	CUTI [27]	NTL [14]	CUPI	CUTI [27]	NTL [14]
Scheme-1	**99.84**$^{*\diamond}$	99.37	99.03	**11.27**$^{*\diamond}$	15.62	18.00	**88.57 (88.71%)**$^{*\diamond}$	83.75 (84.27%)	81.03 (81.81%)
Scheme-2	**99.75**$^{*\diamond}$	99.47	99.34	**14.25**$^{*\diamond}$	16.63	18.60	**85.50 (85.71%)**$^{*\diamond}$	85.85 (83.27%)	80.75 (81.28%)
Scheme-3	**99.08**$^{*\diamond}$	98.77	98.64	**12.72**$^{*\diamond}$	14.78	16.90	**86.36 (87.16%)**$^{*\diamond}$	84.00 (85.05%)	81.74 (82.88%)

TABLE 6
Accuracy (%) of CUPI-Domain on five random tasks with various components combinations. The vertical axis axis represents the combinations of different components, while the horizontal axis represents the loss function, with '✓' indicating the correspondence.

Architecture	$\mathcal{L}_s$	$\mathcal{L}_i$	$\mathcal{L}_t$	$\mathcal{L}_{Stl}$	$\mathcal{L}_{Dis}$	MNIST → SVHN	CIFAR10 → STL10	VisDA-T → VisDA-V	Artistic → Product	quickdraw → infograph
SL	✓					38.2	64.4	36.5	64.9	3.5
SL+CUPI-Domain generators (without $\mathcal{L}_t$)	✓	✓				11.4	10.1	10.7	10.9	2.2
SL+CUPI-Domain generators (without $\mathcal{L}_i$)	✓		✓			9.4	11.4	11.9	10.4	3.0
SL+CUPI-Domain generators	✓	✓	✓			9.4	11.2	10.1	9.9	2.8
SL+CUPI-Domain generators+DIMB (without $\mathcal{L}_{Stl}$)	✓	✓	✓	✓		6.3	11.0	8.2	9.4	1.8
SL+CUPI-Domain generators+DIMB (without $\mathcal{L}_{Dis}$)	✓	✓	✓		✓	6.3	9.9	9.6	9.4	1.6
SL+CUPI-Domain generators+DIMB (CUPI-Domain)	✓	✓	✓	✓	✓	**6.1**	**9.5**	**7.5**	**8.1**	**1.4**

component combinations (indicated by '+'), and the checkmark '✓' indicates the corresponding loss function on the horizontal axis. For each task, we randomly selected two domains from each dataset, designating one as the authorized domain (left of '→') and the other as the unauthorized domain (right of '→'), as shown on the horizontal axis.

The complexity of the dataset varies, leading to differences in the difficulty of feature extraction and the construction of the CUPI-Domain. Consequently, the accuracy of CUPI-Domain with different partial combinations on the unauthorized domain also varies. However, three key observations remain consistent across all tasks: firstly, the accuracy of different variants on the unauthorized domain is significantly lower than that of SL, indicating the positive impact of each component; secondly, the variants with DIMB perform better than those without it, indicating that DIMB can obtain stable domain class features and domain class-wise style features, further amplifying the difference between authorized domains and unauthorized domains, thereby reducing the identification of unauthorized domains. thirdly, the complete combinations consistently achieve the lowest accuracy on the unauthorized domain,

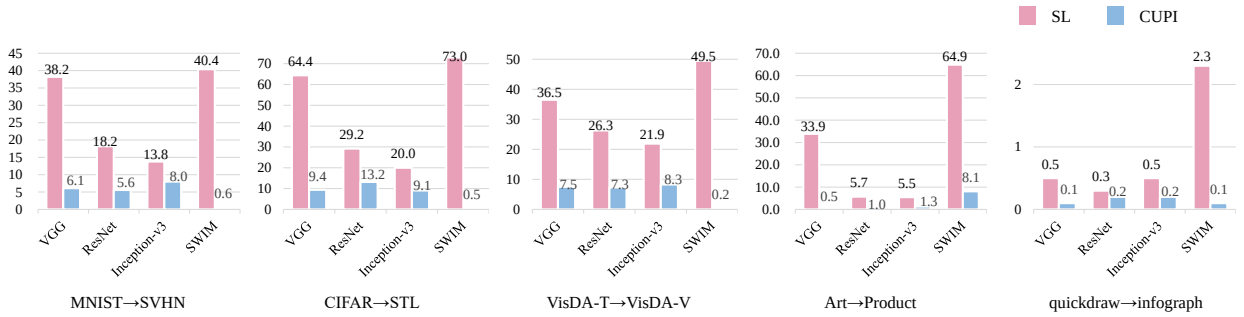

Fig. 6. The accuracy (%) of SL and target-specified CUPI-Domain combined with four different backbones (VGG [7], ResNet [62], Inception-v3 [8], and SWIM [9] on five random tasks.

highlighting the effectiveness of our proposed loss function in protecting model IP.

4.8 Ablation Study of Backbone

In this section, we investigate the IP protection capability of the proposed CUPI-Domain in combination with different backbone architectures, as illustrated in Fig. 6. We selected four distinct backbone networks, including VGG [7], ResNet [62], Inception-v3 [8], and SWIM [9]. We perform a total of five transfer tasks based on the number of available datasets. For each transfer task, we randomly selected two domains from each dataset, designating one as the authorized domain (left of '→') and the other as the unauthorized domain (right of '→'). In each subfigure, the pink bars indicate the accuracy of SL on the unauthorized domain, while the blue bars depict the accuracy of CUPI-Domain on the unauthorized domain.

Based on the experimental results, we can observe a consistent pattern: irrespective of the dataset, CUPI-Domain consistently outperforms SL in reducing the recognition ability on the unauthorized domain when integrated with various backbones. Furthermore, when combined with SWIM [9], CUPI-Domain exhibits higher accuracy on the authorized domain and lower accuracy on the unauthorized domain. This can be attributed to the stronger feature extraction ability of SWIM [9] on complex datasets compared to the other backbones. As a result, when integrated with SWIM [9], CUPI-Domain becomes more similar to the authorized domain, leading to a more compact representation of the model's performance within the authorized domain. This signifies a stronger IP protection ability with a stronger feature extractor.

5 Discussion

5.1 Protection & Generalization

In the target-specified scenario, CUPI-Domain effectively diminishes the identification ability of unauthorized domains by constraining the generalization boundary of the authorized domain while preserving its universality. However, in target-free scenarios, where the distribution of unauthorized domains varies, the IP protection mechanism of the model naturally results in a certain degree of loss of universality.

The trade-off between protection and generalization is a critical consideration. While our framework prioritizes model IP protection, it is equally imperative to ensure that the protected model remains effective across diverse deployment scenarios. We acknowledge the need for further investigation and research to address this trade-off, such as exploring methods to enhance the generalization ability of protected models through additional training or data augmentation strategies. This area is one that we plan to delve into in future work to ensure that our framework remains practical and effective in real-world applications.

5.2 Attacks from Input Data

The design of our proposed CUPI-Domain is grounded in the deep features of the domain, extracted by the feature extractor of the backbone. By emphasizing the private features of authorized domains, CUPI-Domain can establish more restrictive generalization boundaries for trained models, implicitly preventing unauthorized transfer to unauthorized domains with irrelevant private-style features. In this section, we further explore whether CUPI-Domain can resist input-based IP attacks.

Recently, several image translation methods based on CycleGAN [63] have been proposed to translate images between two different domains, aiming to reduce the domain gap [64]. To verify the resistance of CUPI-Domain to such methods, we applied CycleGAN [63] to generate transformed images for each task pair (i.e., any two different domains with the same task) and tested the corresponding trained CUPI-Domain, CUTI-Domain [27] and NTL [14] on these images to assess their IP protection capabilities. The experimental results of the trained target-specified model and target-free model are presented in Table 7 (with more details in Supplementary Table 17).

Table 7 showcases the average accuracy of the trained target-specified/free models on translated images. The mean accuracy of different methods ranges from 12% to 25%, demonstrating poor recognition ability. These results suggest a certain level of resistance to input-based attacks of these IP protection methods, with CUPI-Domain demonstrating the best performance with a statistical difference ($p < 0.05$). This underscores the effectiveness of CUPI-Domain in preserving the IP of the model against potential adversarial attacks.

5.3 Attacks from Model Stealing

We further verify the ability of the proposed CUPI-Domain to resist some of the latest model stealing attack meth-

TABLE 7
The average accuracy (%) of target-specified/free CUPI-Domain, CUTI-Domain [27] and NTL [14] on translated input data [63], the unauthorized domain after model stealing attack by DeepSteal [65] and Knockoff-Nets [66].

Domain	Target-Specified			Target-Free		
	CUPI	CUTI [27]	NTL [14]	CUPI	CUTI [27]	NTL [14]
CycleGAN [63]	**12.39***	15.45	15.97	**21.06***	22.41	24.05
DeepSteal [65]	**8.71**$^{\diamond}$	9.45	13.53	**10.19**$^{*\diamond}$	11.31	17.51
Knockoff [66]	**10.75**$^{\diamond}$	16.80	17.19	**9.57**$^{*\diamond}$	19.92	17.08

ods (i.e., DeepSteal [65] and Knockoff-Nets [66]). Table 7, Supplementary Table 18 and 19 show the accuracy of the substitute model according to CUPI-Domain, CUTI-Domain [27], and NTL [14] on the unauthorized domain in target-specified/free scenarios. The average accuracy of the unauthorized domain remains low, indicating that these methods can effectively thwart model-stealing attacks to some extent. Among them, CUPI-Domain has the lowest accuracy with a statistical difference ($p < 0.05$), demonstrating the strongest IP protection ability.

5.4 Exploration on More Data Types

To further verify the applicability and limitations of the proposed CUPI-Domain, we deployed it on text classification tasks (5 categories, random 4 domains from 35) [67] and audio classification tasks (10 categories, 3 domains) [68]. The experimental results are presented in Table 8, Supplementary Table 20 and Table 21. As shown in the Table, CUPI-Domain, CUTI-Domain [27], and NTL [14] have successfully reduced the performance of unauthorized access while retaining the performance of authorized domains, demonstrating certain IP protection capabilities, with CUPI-Domain shows the strongest protection capability with a statistical difference ($p < 0.05$).

However, it should be emphasized that the design and development of CUPI-Domain are based on image features, and its performance on text and audio tasks is weaker than on images. Further exploration and improvement are needed to enhance CUPI-Domain's adaptability to text and audio IP protection tasks. This is an area that we plan to explore in future work to ensure that our framework remains adaptable and effective in more real-world applications.

5.5 Exploration on Real-World Clinical Data

To further validate the applicability of the proposed CUPI-Domain in real-world scenarios, we deployed it in a tuberculosis classification task with two categories and two domains (ChinaSet [69] and Montgomery CXR [70]). The experimental results are presented in Table 9 (with more details in Supplementary Table 22, Supplementary Table 23), which demonstrates that CUPI-Domain exhibits robust IP protection capabilities in both target-specified and target-free scenarios, with a statistical difference ($p < 0.05$).

6 Conclusion

Protecting model intellectual property (IP) in the field of artificial intelligence is a significant and crucial challenge, and its significance cannot be overstated. To tackle this issue, we propose a novel approach called CUPI-Domain, which acts as a protective barrier to confine the model's performance within authorized domains. To achieve this, the pyramid CUPI-Domain generator is designed to construct an isolation domain that closely resembles the features of the authorized domain, thereby hindering the model's ability to generalize to unauthorized domains and leading to recognition failure when encountering new private-style features. Additionally, we design external DIMB for storing and updating labeled pyramid features with corresponding loss functions. Furthermore, we offer two solutions to deploy the CUPI-Domain: target-specified and target-free, depending on whether information about the unauthorized domain is available. Through extensive experiments conducted on several popular cross-domain datasets, we demonstrate the effectiveness of our lightweight and easily deployable CUPI-Domain. We believe that our work will contribute to the advancement of research on model IP protection and security, highlighting its crucial importance in real-world applications.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Nos. 62136004, 62276130), the Key Research and Development Plan of Jiangsu Province (No. BE2022842), and Huazhu Fu's A*STAR Central Research Fund and Career Development Fund (C222812010).

References

[1] C. Sun, A. Shrivastava, S. Singh, and A. Gupta, "Revisiting unreasonable effectiveness of data in deep learning era," in *ICCV*, 2017, pp. 843–852.
[2] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *CVPR*, 2009, pp. 248–255.
[3] T. Chilimbi, Y. Suzue, J. Apacible, and K. Kalyanaraman, "Project adam: Building an efficient and scalable deep learning training system," in *11th USENIX Symposium on Operating Systems Design and Implementation (OSDI 14)*, 2014, pp. 571–582.
[4] N. P. Jouppi, C. Young, N. Patil, D. Patterson, G. Agrawal, R. Bajwa *et al.*, "In-datacenter performance analysis of a tensor processing unit," in *Proceedings of the 44th Annual International Symposium on Computer Architecture*, 2017, pp. 1–12.
[5] F. Shamshad, S. Khan, S. W. Zamir, M. H. Khan, M. Hayat, F. S. Khan, and H. Fu, "Transformers in medical imaging: A survey," *arXiv*, jan 2022. [Online]. Available: http://arxiv.org/abs/2201.09873
[6] R. Bommasani, D. A. Hudson, E. Adeli, R. Altman, S. Arora, S. von Arx *et al.*, "On the opportunities and risks of foundation models," *arXiv*, aug 2021. [Online]. Available: http://arxiv.org/abs/2108.07258
[7] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.
[8] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *CVPR*, 2016, pp. 2818–2826.
[9] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *ICCV*, 2021, pp. 10 012–10 022.
[10] M. Chen and M. Wu, "Protect your deep neural networks from piracy," in *2018 IEEE International Workshop on Information Forensics and Security (WIFS)*. IEEE, 2018, pp. 1–7.
[11] M. Ribeiro, K. Grolinger, and M. A. Capretz, "Mlaas: Machine learning as a service," in *2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA)*. IEEE, 2015, pp. 896–902.

TABLE 8
Results of target-specified CUPI-Domain, CUTI-Domain [27] and NTL [14] on Amazon Reviews Dataset [67] and DCASE Dataset [68].

Domain / Drop	Authorized Drop↓			Unauthorized Drop↑		
	CUPI-Domain	CUTI-Domain [27]	NTL [14]	CUPI-Domain	CUTI-Domain [27]	NTL [14]
Amazon [67]	**0.17 (0.26%)***◇	0.56 (0.90%)	0.53 (0.82%)	**42.50 (70.13%)***◇	40.26 (66.35%)	39.94 (65.76%)
DCASE [68]	**0.00 (0.00%)**◇	0.17 (0.32%)	0.87 (1.62%)	**16.67 (58.17%)***◇	14.76 (48.63%)	14.06 (46.25%)

TABLE 9
Results of target-specified/free CUPI-Domain, CUTI-Domain [27] and NTL [14] on the tuberculosis classification task.

Domain / Drop	Authorized Drop↓			Unauthorized Drop↑		
	CUPI	CUTI [27]	NTL [14]	CUPI	CUTI [27]	NTL [14]
Target-specified	**0.00 (0.00%)***◇	0.50 (0.55%)	0.50 (0.55%)	51.05 (75.88%)*◇	46.85 (69.39%)	46.85 (70.50%)
Target-free	**0.00 (0.00%)***◇	1.05 (1.17%)	1.05 (1.17%)	45.30 (66.85%)*◇	42.15 (61.99%)	37.45 (54.86%)

[12] M. Xue, Y. Zhang, J. Wang, and W. Liu, "Intellectual property protection for deep learning models: Taxonomy, methods, attacks, and evaluations," *IEEE Transactions on Artificial Intelligence*, pp. 1–1, 2021.

[13] A. Chakraborty, A. Mondai, and A. Srivastava, "Hardware-assisted intellectual property protection of deep learning models," in *57th ACM/IEEE Design Automation Conference*, 2020, pp. 1–6.

[14] L. Wang, S. Xu, R. Xu, X. Wang, and Q. Zhu, "Non-transferable learning: A new approach for model ownership verification and applicability authorization," in *ICLR*, 2021.

[15] W. Mei and W. Deng, "Deep visual domain adaptation: A survey," *Neurocomputing*, vol. 312, pp. 135–153, 2018.

[16] K. You, M. Long, Z. Cao, J. Wang, and M. I. Jordan, "Universal domain adaptation," in *CVPR*, 2019, pp. 2720–2729.

[17] K. Zhou, Z. Liu, Y. Qiao, T. Xiang, and C. C. Loy, "Domain generalization: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–20, 2022.

[18] J. Wang, C. Lan, C. Liu, Y. Ouyang, T. Qin, W. Lu *et al.*, "Generalizing to unseen domains: A survey on domain generalization," *IEEE Transactions on Knowledge and Data Engineering*, pp. 1–1, 2022.

[19] B. Sampat and H. L. Williams, "How do patents affect follow-on innovation? evidence from the human genome," *American Economic Review*, vol. 109, no. 1, pp. 203–236, 2019.

[20] H. C. Tanuwidjaja, R. Choi, S. Baek, and K. Kim, "A comprehensive survey of privacy-preserving federated learning: A taxonomy, review, and future directions," *ACM Computing Surveys*, vol. 54, no. 6, pp. 1–36, 2021.

[21] P. W. Wong and N. Memon, "Secret and public key image watermarking schemes for image authentication and ownership verification," *IEEE Transactions on Image Processing*, vol. 10, no. 10, pp. 1593–1601, 2001.

[22] C. Song, T. Ristenpart, and V. Shmatikov, "Machine learning models that remember too much," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, 2017, pp. 587–601.

[23] E. L. Merrer, P. Perez, and G. Tredan, "Adversarial frontier stitching for remote neural network watermarking," *Neural Computing and Applications*, vol. 32, no. 13, pp. 9233–9244, 2020.

[24] L. Fan, K. W. Ng, and C. S. Chan, "Rethinking deep neural network ownership verification: Embedding passports to defeat ambiguity attacks," *Advances in Neural Information Processing Systems 32*, 2019.

[25] Y. Adi, C. Baum, M. Cisse, B. Pinkas, and J. Keshet, "Turning your weakness into a strength: Watermarking deep neural networks by backdooring," in *27th USENIX Security Symposium*, 2018, pp. 1615–1631.

[26] X. Chen, W. Wang, C. Bender, Y. Ding, R. Jia, B. Li, and D. Song, "Refit: a unified watermark removal framework for deep learning systems with limited data," in *Proceedings of the 2021 ACM Asia Conference on Computer and Communications Security*, 2021, pp. 321–335.

[27] L. Wang, M. Wang, D. Zhang, and H. Fu, "Model barrier: A compact un-transferable isolation domain for model intellectual property protection," in *CVPR*, 2023, pp. 20 475–20 484.

[28] H. Venkateswara, J. Eusebio, S. Chakraborty, and S. Panchanathan, "Deep hashing network for unsupervised domain adaptation," in *CVPR*, 2017, pp. 5018–5027.

[29] X. Peng, Q. Bai, X. Xia, Z. Huang, K. Saenko, and B. Wang, "Moment matching for multi-source domain adaptation," in *ICCV*, 2019, pp. 1406–1415.

[30] C. Song, T. Ristenpart, and V. Shmatikov, "Machine learning models that remember too much," in *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, 2017, pp. 587–601.

[31] M. Kuribayashi, T. Tanaka, and N. Funabiki, "Deepwatermark: Embedding watermark into dnn model," in *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, 2020, pp. 1340–1346.

[32] Y. Adi, C. Baum, M. Cisse, B. Pinkas, and J. Keshet, "Turning your weakness into a strength: Watermarking deep neural networks by backdooring," in *27th USENIX Security Symposium*, 2018, pp. 1615–1631.

[33] J. Zhang, D. Chen, J. Liao, H. Fang, W. Zhang, W. Zhou *et al.*, "Model watermarking for image processing networks," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 07, 2020, pp. 12 805–12 812.

[34] M. Alam, S. Saha, D. Mukhopadhyay, and S. Kundu, "Deep-lock: Secure authorization for deep neural networks," *arXiv preprint arXiv:2008.05966*, 2020.

[35] Z. Song, W. Jiang, J. Zhan, X. Wen, and C. Bian, "Critical-weight based locking scheme for dnn ip protection in edge computing: work-in-progress," in *Proceedings of the 2021 International Conference on Hardware/Software Codesign and System Synthesis*, 2021, pp. 33–34.

[36] W. Gabriela, "Domain adaptation for visual applications: A comprehensive survey," *arXiv preprint arXiv:1702.05374*, 2017.

[37] B. Sun, J. Feng, and K. Saenko, "Correlation alignment for unsupervised domain adaptation," *Domain Adaptation in Computer Vision Applications*, pp. 153–171, 2017.

[38] J. Zhuo, S. Wang, W. Zhang, and Q. Huang, "Deep unsupervised convolutional domain adaptation," in *Proceedings of the 25th ACM International Conference on Multimedia*, 2017, pp. 261–269.

[39] K. Saito, K. Watanabe, Y. Ushiku, and T. Harada, "Maximum classifier discrepancy for unsupervised domain adaptation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3723–3732.

[40] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, "Domain randomization for transferring deep neural networks from simulation to the real world," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 23–30.

[41] A. Prakash, S. Boochoon, M. Brophy, D. Acuna, E. Cameracci, G. State *et al.*, "Structured domain randomization: Bridging the reality gap by context-aware synthetic data," in *2019 International Conference on Robotics and Automation*, 2019, pp. 7249–7255.

[42] B. Gong, Y. Shi, F. Sha, and K. Grauman, "Geodesic flow kernel for unsupervised domain adaptation," in *CVPR*, 2012, pp. 2066–2073.

[43] R. Gopalan, R. Li, and R. R. Chellappa, "Unsupervised adaptation across domain shifts by generating intermediate data representations," *IEEE TPAMI*, vol. 36, no. 11, 2013.

[44] Y. Dai, J. Liu, Y. Sun, Z. Tong, C. Zhang, and L.-Y. Duan, "Idm: An intermediate domain module for domain adaptive person re-id," in *ICCV*, 2021, pp. 11 864–11 874.

[45] Y. Sun, G. Yang, D. Ding, G. Cheng, J. Xu, and X. Li, "A gan-based domain adaptation method for glaucoma diagnosis," in *International Joint Conference on Neural Networks*, 2020, pp. 1–8.

[46] S. Li, M. Xie, K. Gong, C. H. Liu, Y. Wang, and W. Li, "Transferable semantic augmentation for domain adaptation," in *CVPR*, 2021, pp. 11 516–11 525.

[47] H.-H. Zhao, P. L. Rosin, Y.-K. Lai, and Y.-N. Wang, "Automatic semantic style transfer using deep convolutional neural networks and soft masks," *The Visual Computer*, vol. 36, pp. 1307–1324, 2020.

[48] Y. Deng, F. Tang, W. Dong, C. Ma, X. Pan, L. Wang, and C. Xu, "Stytr2: Image style transfer with transformers," in *CVPR*, 2022, pp. 11 326–11 336.

[49] V. Dumoulin, J. Shlens, and M. Kudlur, "A learned representation for artistic style," *arXiv preprint arXiv:1610.07629*, 2016.

[50] X. Huang and S. Belongie, "Arbitrary style transfer in real-time with adaptive instance normalization," in *ICCV*, 2017, pp. 1501–1510.

[51] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis," in *CVPR*, 2017, pp. 6924–6932.

[52] L. Deng, "The mnist database of handwritten digit images for machine learning research [best of the web]," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 141–142, 2012.

[53] J. J. Hull, "A database for handwritten text recognition research," *IEEE TPAMI*, vol. 16, no. 5, pp. 550–554, 1994.

[54] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng, "Reading digits in natural images with unsupervised feature learning," *NIPS Workshop*, 2011.

[55] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation." PMLR, 2015, pp. 1180–1189.

[56] K. Alex and G. Hinton, "Learning multiple layers of features from tiny images," *Technical Report*, 2009.

[57] A. Coates, A. Ng, and H. Lee, "An analysis of single-layer networks in unsupervised feature learning," in *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*. JMLR Workshop and Conference Proceedings, 2011, pp. 215–223.

[58] G. French, M. Mackiewicz, and M. Fisher, "Self-ensembling for domain adaptation," *arXiv preprint arXiv:1706.05208*, vol. 7, 2017.

[59] X. Peng, B. Usman, N. Kaushik, J. Hoffman, D. Wang, and K. Saenko, "Visda: The visual domain adaptation challenge," *arXiv preprint arXiv:1710.06924*, 2017.

[60] L. Gillick and S. J. Cox, "Some statistical issues in the comparison of speech recognition algorithms," in *International Conference on Acoustics, Speech, and Signal Processing*. IEEE, 1989, pp. 532–535.

[61] J. Blitzer, R. McDonald, and F. Pereira, "Domain adaptation with structural correspondence learning," in *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, 2006, pp. 120–128.

[62] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016, pp. 770–778.

[63] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2223–2232.

[64] P. Soviany, R. T. Ionescu, P. Rota, and N. Sebe, "Curriculum self-paced learning for cross-domain object detection," *Computer Vision and Image Understanding*, vol. 204, p. 103166, 2021.

[65] A. S. Rakin, M. H. I. Chowdhuryy, F. Yao, and D. Fan, "Deepsteal: Advanced model extractions leveraging efficient weight stealing in memories," in *2022 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2022, pp. 1157–1174.

[66] T. Orekondy, B. Schiele, and M. Fritz, "Knockoff nets: Stealing functionality of black-box models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4954–4963.

[67] J. McAuley and J. Leskovec, "Hidden factors and hidden topics: understanding rating dimensions with review text," in *Proceedings of the 7th ACM Conference on Recommender Systems*, 2013, pp. 165–172.

[68] A. Mesaros, T. Heittola, and T. Virtanen, "A multi-device dataset for urban acoustic scene classification," *arXiv preprint arXiv:1807.09840*, 2018.

[69] "Chinaset," *[online] Available:*. [Online]. Available: https://aistudio.baidu.com/aistudio/datasetdetail/25237

[70] S. Candemir, S. Jaeger, J. Musco, Z. Xue, A. Karargyris, S. Antani, G. Thoma, and K. Palaniappan, "Lung segmentation in chest radiographs using anatomical atlases with nonrigid registration," *IEEE Transactions on Medical Imaging*, vol. 33, no. 2, pp. 577–590, 2014.