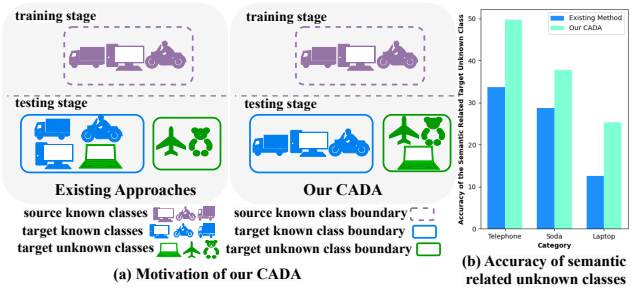
Class-Aware Diversified Augmentation for Open-Set Single Domain Generalization

Jian Hu, Shaogang Gong, Weitong Cai, Junchi Yan

***Abstract*—In Open-Set Single Domain Generalization (OS-SDG), one only has access to a single labeled source domain for training. It assumes that the learned model generalizes well to target samples belonging to the source label space whilst classifies target samples outside the source label space into a single "unknown" class. The current method synthesizes new samples that are semantically unrelated to known classes to simulate target unknown classes. This ignores that unknown classes actually may semantically correlated to known classes, making it difficult to discriminate samples at the margins of class decision boundaries as "unknown". In this work, we introduce a Class-Aware Diversified Augmentation (CADA) method to overcome this problem. Our key idea is to synthesize explicitly new multiple unknown target classes with diversified semantic and learn the inherent correlation among the known and unknown classes, so to both increase the coverage of multiple target unknown classes and to optimize class margin separation. CADA is optimized by enhanced diversity maximization and class-aware minimization. The former synthesizes more novel classes by considering both semantic relationships to known classes and domain shift between the source and target domains. The latter employs class-agnostic clustering with synthesized samples to simulate class correlations among target classes, maximizing class margin separation. Theoretical analysis and experiments on five benchmarks show the efficacy of our CADA.**

***Index Terms*—Transfer Learning, Semi-supervised Learning, Domain Generalization.**



(a) Motivation of our CADA (b) Accuracy of semantic related unknown classes

Fig. 1: (a) Motivation of our CADA. Open-Set Single Domain Generalization [51] synthesizes new unknown class examples that are semantically unrelated to the known classes, mimicing target unknown classes in model training. In practice, target unknown classes can either be close to or overlapping with the known classes in the feature space, therefore difficult to be discriminated, e.g. unknown class "bicycle" is similar to known class "motorcar". CADA explicitly synthesizes unknown classes that are "close" to source known classes in model training in order to improve model generalization in a target domain with both existing known and new unknown classes. (b) Our synthesized unknown classes are semantically similar to known classes whilst CADA can discriminate better than the existing method [51].

I. Introduction

Deep learning in computer vision has shown exceptional performance with large amounts of labeled data [8, 9]. But it assumes that training data (source domain) and test data (target domain) are from the same distribution (i.i.d. assumption). Domain Adaptation (DA) [2, 6] relaxes this assumption by minimizing the distribution gap between domains. However, it needs access target domains during training. Alternatively, Domain Generalization (DG) [19] employs multiple sources for learning a domain-invariant representation in order to generalize the model to unseen target domains. Both DA and DG usually assume source and target domains share the same label space. This is not always true. Multi-Source Open Domain Generalization [32] trains a model across multiple source domains with unaligned label spaces, enabling it to generalize well to unseen domains with unknown classes. It has also been applied to large-scale language model training to improve generalization ability [1], but the performance of such large models in specific scenarios is often unsatisfactory, i.e., medical image analysis [44]. Additionally, due to security and privacy restrictions, collecting labeled data from various domains in these scenarios is also a challenge. A unique approach to minimizing dependency on multiple datasets is Open-Set Single Domain Generalization (OS-SDG) called CrossMatch [51], where a model trained on a single labeled source domain not only generalizes to *unseen* target domains on known classes but also learns to recognize new unknown classes explicitly in the unseen target domain. Tab. I shows comparisons on different DA and DG scenarios.

However, OS-SDG is a harder problem to solve. It needs address both domain shift and label space expansion without accessing target domain training data. CrossMatch [51] synthesizes novel samples with as distinct semantic and domain shifts as possible from the source domain, to simulate unseen target domains during training. But CrossMatch ignores the fact that potential new unknown classes in an unseen target domain may not be well-separated in the feature space from known classes, resulting in target unknown classes being misclassified as target known ones (see Fig. 1(a)). This problem is further aggravated when there is a domain shift between source and

Jian Hu, Shaogang Gong and Weitong Cai ({jian.hu, s.gong, weitong.cai}@qmul.ac.uk) are in Queen Mary University of London, Uk. Junchi Yan (yanjunchi@sjtu.edu.cn) is in Shanghai Jiao Tong University, CN.

Manuscript received April 19, 2021; revised August 16, 2021.

TABLE I: A comparison of different DA and DG settings.

Methods	unaligned label space	only one source domain	inaccessible target domain
Closed-Set Domain Adaptation [35]	✗	✓	✗
Domain Adaptation with category gap [11, 12, 45]	✓	✓	✗
Multi-Source Domain Adaptation [26, 47]	✗	✗	✗
Domain Generalization [19, 25]	✗	✗	✓
Single Source Domain Generalization [29]	✗	✓	✓
Multi-Soucre Open Domain Generalization [32]	✓	✗	✓
Open-Set Single Domain Generalization [51]	✓	✓	✓

target (see Fig. 1(b)). Moreover, since the number of target unknown classes is arbitrary and unknown, CrossMatch treats all simulated target unknown class samples as a single new "unknown" class to the known classes. It ignores any correlations among the unknown classes as well as their relationships to known classes, resulting in suboptimal model learning.

To address the above problems, we introduce Class-Aware Diversified Augmentation (CADA), consisting of two model optimization objectives: enhanced diversity maximization and class-aware minimization. During maximization, diversified unknown classes are first synthesized through unknown maximization. After append synthesized unknown classes into source data, it then synthesizes both known and unknown classes with domain shift to simulate unseen target domains through out-of-distribution maximization. Specifically, The synthesized unknown samples have unique semantics distinct from known classes while still maintaining correlations with the known classes used for their synthesis. After maximization, in order to improve model's open-set generalization ability, synthesized samples are append to source data for class-aware minimization. Model is trained via class-agnostic clustering and modified supervised learning on expand source data, to maximize class margin whilst simultaneously minimize intra-class scattering in the presence of known classes nearby. **Our contributions are as follows:**

(1) We introduce a new Class-Aware Diversified Augmentation (CADA) method to explicitly synthesize potential target unknown class samples that are "close" to known classes when the target domain is unseen in model training. This is designed to address the limitations of the existing OS-SDG model CrossMatch which ignores any potential class similarities between known and unknown classes.

(2) To implement the CADA idea, we formulate a joint learning objective for both enhanced diversity maximization to synthesize diverse unknown class samples that are similar to those of known classes, and class-aware minimization to perform unsupervised clustering of the synthesized samples without knowing the number of target unknown classes.

(3) Theoretical analysis and comprehensive experiments show that CADA outperforms a wide range of existing generalization and adaptation methods, demonstrating its superiority to unseen target domains of different distribution shifts.

II. Related Works

Multi-Source Domain Generalization [25, 32, 33, 41] enhance target performance by learning cross-domain invariance from multiple source domains. Kernel-based methods [25], meta-learning [32], and data augmentation [38, 41] are proposed to address this, but fail to identify novel target classes. [1] leverages meta-learning to extract invariance from source domains to classify known targets and identify unknowns. [33] uses CLIP's generalization with prompt optimization for unknown class recognition. [3] distills a large model to help a small model generalize to unseen unknown classes. However, these methods rely on multiple source domains, which is impractical in real-world applications.

Single Source Domain Generalization [32] relaxes this assumption. It only needs a labeled source domain for training and can be generalized to multiple unseen target domains under the same label space. [49] presents adversarial gradient-based augmentation to address it and achieves good performance. CrossMatch [51] further introduces Open-Set Single Domain generalization. It assumes that the target domain includes novel classes that do not appear in the source. It simulates target unseen samples by synthesization, but synthesized unknown samples are semantically unrelated to known classes, ignores unknown samples closely related to the known classes.

Data Augmentation with Diversity. Data Augmentation [21, 38] is a max-min game that maximization synthesizes samples on the source distribution edge to simulate out-of-distribution target samples, while minimization learns from them to improve generalization ability. CrossMatch [51] synthesizes novel samples beyond source label space with domain shift. But these samples lack semantic diversity, making it difficult to classify boundary classes. Contrastly, CADA synthesizes diverse novel classes semantically related to the known classes to enhance its open-set generalization capability.

III. Class-Aware Diversified Augmentation

Problem Setting. In OS-SDG, the model is trained on an annotated source domain D_s and tested on multiple unannotated target domains $D_t = \{D_{t_1}, D_{t_2}, ...D_{t_H}\}$, which are inaccessible during training. The source domain $D_s = \{x_s^i, y_s^i\}_{i=1}^{n_s}$ has n_s labeled samples, while each target domain $D_{t_h} = \{x_{t_h}^i\}_{i=1}^{n_{t_h}}$ has n_{t_h} unlabeled samples. Target domains include novel categories absent from the source domain. The source label space $\mathcal{C}_s$ is a subset of the target $\mathcal{C}_t$, i.e., $\mathcal{C}_s \subset \mathcal{C}_t$, with novel target classes defined as $\mathcal{C}_t^u = \mathcal{C}_t \setminus \mathcal{C}_s$. During inference, all novel classes $\mathcal{C}_t^u$ are grouped into a single "unknown" class. The label spaces of multiple target domains $\mathcal{C}_{t_1}, ..., \mathcal{C}_{t_H}$ are not fixed but all include $\mathcal{C}_s$. Additionally, there is a distribution shift between source and target domains: $\mathbb{P}_s(x) \neq \mathbb{P}_t(x)$.

Overview. We introduce the Class-Aware Diversified Augmentation (CADA). Fig.2 shows that CADA consists of both enhanced diversity maximization and class-aware minimization. The maximization stage further comprises unknown maximization and out-of-distribution maximization. The former generates unknown samples without domain shift, preserving unique semantics distinct from known classes while maintaining correlations with the known samples used for synthesis. The latter extends this by synthesizing both known and unknown classes under domain shift, simulating unseen target domains. Minimization combines original and synthesized samples to improve open-set generalization ability.

Preliminaries In this section, we review the principle of worst-case problem [34] and adversarial domain generation [38]. In single domain generalization, worst-case problem is employed

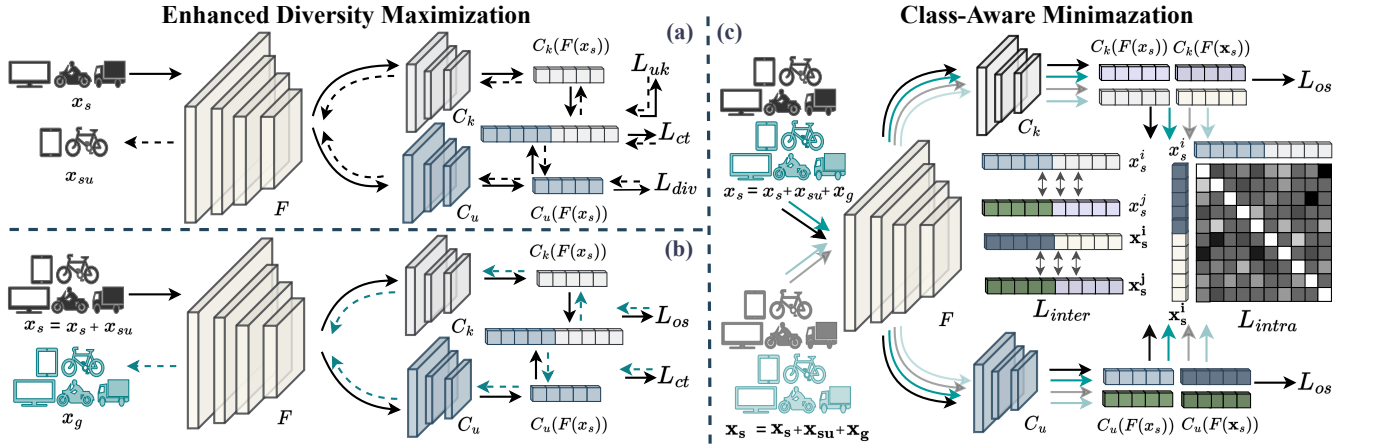


Fig. 2: CADA includes a feature extractor F, a known class classifier C_k, and an unknown class classifier C_u, with enhanced diversity maximization on the left and class-aware minimization on the right. Solid lines represent forward propagation, while dashed lines represent backward propagation. Maximization includes unknown and out-of-distribution maximization. (a) Unknown maximization, guided by unknown loss L_{uk}, semantic consistency loss L_{ct}, and diversified loss L_{div}, synthesizes unknown samples without domain shift x_{su}, which maintains correlation with the corresponding class of the source sample used for synthesizing them. (b) Out-of-distribution maximization appends x_{su} to x_s and synthesizes out-of-distribution samples x_g, to simulate target domain that has both known and unknown classes with open-set loss L_{os} and semantic consistency loss L_{ct}. (c) Minimization improves generalization using both source samples x_s and augmented samples $\mathbf{x_s}$ with open-set loss L_{os},inter-sample loss L_{inter} and intra-sample loss L_{intra}. It explores the generalized class correlation and improves generalization performance. Maximization and minimization are performed alternatively.

to iteratively learn "hard" data points from fictitious target distributions to learn generalization ability by addressing:

$$\min_{\theta} \sup_{D_t} \{\mathbb{E}[L_{\text{ce}}(\theta; D_t)] : d(D_t, D_s) \leq \rho\}, \tag{1}$$

d is used to quantify the similarity between D_s and D_t. ρ corresponds to the maximum allowable cost to transfer D_s to D_t. Meanwhile, θ is the parameters of the model, which are refined through cross-entropy loss L_{ce}. Adversarial Data Augmentation expands the source domain's diversity by creating a domain D_g to approximate the unseen target domains D_t. It redefines the worst-case problem (Eq.1) into a Lagrangian optimization problem with a fixed penalty parameter γ:

$$\min_{\theta} \sup_{D_g} \{\mathbb{E}[L_{\text{ce}}(\theta; D_g)] - \gamma d(D_g, D_s)\}, \tag{2}$$

where $d(D_g, D_s) = L_{\text{ct}}(\theta_g; D_g, D_s) = \|F(x_g) - F(x_s)\|_2^2 + \infty \cdot \mathbf{1}_{\{y \neq y_s\}}$, F is the feature extractor. $\mathbf{1}\{\cdot\}$ is the 0-1 indicator function. It ensures consistent semantics between the synthesized samples and their corresponding original samples used for synthesis. The loss function is:

$$L_{\text{ada}}(\theta, \theta_g; D_s, D_g) = L_{\text{ce}}(\theta; D_g) - \gamma L_{\text{ct}}(\theta_g; D_g, D_s), \tag{3}$$

the training is processed iteratively between two phases: the maximization and the minimization. During maximization, a fictitious target domain D_g is synthesized from D_s by optimizing L_{ada} with learning rate $\eta \geq 0$:

$$x_g \leftarrow x_s + \eta \nabla_{x_s} L_{\text{ada}}(\theta; x_s, x_g), \tag{4}$$

After the maximization phase, the synthesized domain D_g is appended to D_s. In the minimization phase, θ is optimized by minimizing L_{ce} with the updated D_s.

A. Enhanced Diversity Maximization

CrossMatch [51] employs an additional classifier to synthesize unknown samples during above training process to address OS-SDG. In reality, the boundaries between known and unknown classes are artificially defined, with many categories near the boundary having significant correlations. But the unknown classes synthesized by CrossMatch exhibit significant semantic differences from the known classes and do not encompass these boundary unknown classes, leading to poor performance for these classes (Fig. 1 (b)). To address it, during the maximization, we synthesize unknown classes that display correlations with known classes to cover the unknown classes at the classification boundary for better open-set discrimination ability. Samples synthesized by maximization need to address both label space and domain shifts simultaneously from the source domain. Enhanced Diversity Maximization is divided into unknown maximization and out-of-distribution maximization to address them respectively.

1) Unknown Maximization: Unknown maximization synthesizes new unknown classes with different semantics while avoiding domain shift. Traditional classification includes a feature extractor F, and a known class classifier C_k. But source supervised learning cannot distinguish unseen unknown samples. Inspired by [50], we introduce a parallel unknown class classifiers C_u to identify unknown classes. C_u is a linear layer trained from sketch with K-head, and it shares the same feature extractor F as the known classifier C_k. C_u and C_k are used for two objectives simultaneously: classify known classes and distinguish novel samples into a "unknown" classes. For the first objective, augmented features $\mathbf{f}(x_s)$ is defined as:

$$\mathbf{f}(x_s) = [C_k(F(x_s)), \max(C_u(F(x_s)))], \tag{5}$$

where $[\cdot, \cdot]$ is concatenation operation, $\mathbf{f}(x_s)$ is applied to classify known classes with cross-entropy loss L_{ce}. For the second objective, we divide unknown samples into two groups: unrelated to known classes and related to known classes. Unknown samples unrelated to the known classes are distinguished by Eq.11, details are in Sec.III-A2 and III-B. For

unknown samples related to the known classes, we synthesize unknown samples x_{su} without introducing domain shift. It simulates multiple unknown classes at the classification boundary between known and unknown classes using the unknown loss L_{uk}, which is defined as follows:

$$L_{uk}(\theta_{su}, \theta; D_{su}, D_s) = L_{ce}(\mathbf{f}(x_{su}), \mathbf{y_{K+1}}) + L_{ce}(\mathbf{f}^*(x_{su}), K+1), \tag{6}$$

where $\mathbf{y_{K+1}} = (1-\alpha)\mathbb{1}_{K+1} + \alpha/(K+1)$ is the smooth label , K is the number of known classes, α is set as 0.1. $\mathbf{f}^*(x_s)$ is the masked ground truth position features, and is defined as:

$$\mathbf{f}^*(x_s) = [C_k(F(x_s)) \circ (\mathbf{1}_K - \mathbb{1}_{y_s}), \max(C_u(F(x_s)))], \tag{7}$$

where $\mathbf{1}_K$ is the K-dimension all-one vector and $\mathbb{1}_{y_s}$ is K-dimension one-hot encoding with only the y_s-th element as 1. $\circ$ is element-wise product. Eq.6 ensures that the synthesized classes belong to a "unknown" cluster. Meanwhile, since the real unknown samples usually have correlations with known classes instead of being completely unrelated (i.e., "computer" and "laptop" in Fig.1(a)). we further introduce diversified loss L_{div}, which ensures the correlation between the synthesized sample and the corresponding source sample used for synthesization. L_{div} is denoted as:

$$L_{div}(\theta_{su}; D_{su}, D_s) = L_{ce}(C_u(F(x_{su})), y_s). \tag{8}$$

To summary, the overall loss of unknown maximization is:

$$\begin{aligned} L_{um}(\theta, \theta_{su}; D_{su}, D_s) &= L_{ct}(\theta_{su}; D_{su}, D_s) \\ &- \lambda L_{div}(\theta_{su}; D_{su}, D_s) - L_{uk}(\theta_{su}, \theta; D_{su}, D_s), \end{aligned} \tag{9}$$

where λ is the hyper-parameter. By maximizing L_{um}, we synthesize multiple unknown samples with diverse semantics while maintaining some correlation with known classes. This ensures that the synthesized unknown samples belong to different classes, reflecting the actual diversity and interrelation of classes near the known and unknown classification boundary. Similar to Eq.3, We also transform Eq.9 into a Lagrangian relaxation to synthesize unknown classes. Source unknown samples x_{su} are synthesized through the following process:

$$\begin{aligned} x_{su} \leftarrow x_s + \eta \nabla_{x_s} \{ & L_{ct}(\theta_{su}; x_{su}, x_s) - \lambda L_{div}(\theta_{su}; x_{su}, y_s) \\ & - L_{uk}(\theta_{su}, \theta; x_{su}, x_s) \}, \end{aligned} \tag{10}$$

After unknown maximization, synthesized unknown classes in D_{su} have distinct semantic yet without domain shift, and they are appended to D_s for the next stage.

2) Out-of-distribution Maximization: Unknown maximization synthesizes multiple unknown classes with novel semantic to simulate the label space shift. But the target domains are under both label space shift and distribution shift. Out-of-distribution maximization follows unknown maximization, making synthesized samples resemble out-of-distribution samples from unseen targets. We consider the synthesized unknown samples x_{su} as the $(K+1)$-th class and transform the OS-SDG problem into a closed-set Single Domain Generalization problem that contains $K+1$ classes.

Additionally, since a sample is an open-set sample for all classes except its ground truth, we can distinguish unknown classes that are unrelated to both known and synthesized unknown classes by training with x_s and x_{su} using this relationship. For the remaining classes except for its ground truth, $\mathbf{f}^*(x_s)$ is treated as the features of an unknown sample. Using $\mathbf{f}^*(x_s)$ and $\mathbf{f}(x_s)$, we can distinguish semantically unrelated unknown classes by open-set loss L_{os}:

$$L_{os}(x_s, y_s) = L_{ce}(\mathbf{f}(x_s), \mathbf{y_{y_s}}) + L_{ce}(\mathbf{f}^*(x_s), K+1), \tag{11}$$

The first item differentiates known classes and semantically related unknown classes, while the second aligns the masked augmented prediction with the unknown class, enhancing discrimination against semantically unrelated unknown samples without accessing target domains. Since we already possess samples both within and outside the source label space, a natural idea is to synthesize samples that deviate from the source distribution across these spaces. By maximizing Eq. 11 we can generate synthetic classes that incorporate domain shifts at the classification boundary during training, thus enlarging the margins between known and unknown classes. Therefore, we aim to create fictitious target classes by maximizing out-of-distribution characteristics, with our overall objective being:

$$L_{om}(\theta, \theta_g; D_s, D_g) = -L_{ct}(\theta_g; x_g, x_s) + L_{os}(\theta_g; x_g, x_s), \tag{12}$$

where x_g are synthesized fictitious target samples, and $D_g = \{x_g^i\}_{i=1}^{n_g}$ contains n_g samples. The first term guarantees the synthesized samples in D_g maintain semantic connections with the samples in D_s used for their synthesis, the last two terms synthesize samples that aim to confuse the classifier to simulate the domain shift, thereby synthesizing known and unknown samples with domain shift. We apply lagrangian relaxation to synthesize samples as:

$$x_g \leftarrow x_s + \eta \nabla_{x_s} \{ L_{os}(\theta; x_s, y_s) - L_{ct}(\theta_g; x_g, x_s) \}, \tag{13}$$

where D_g contains both known and unknown classes with distribution gap. D_g are appended to D_s after this stage.

B. Class-Aware Minimization

Minimization occurs both during and after maximization. During maximization, minimization focuses solely on supervised training, ensuring that C_k and C_k achieve sufficient discrimination to guide the maximization in generating desired samples. After maximization, the model needs adapt to synthesized known and unknown classes to enhance its cross-domain open-set discrimination ability. However, as the target unknown samples are inaccessible and their categories are agnostic during training, the inherent data structure among unknown classes cannot be fully explored. Furthermore, unknown classes near the classification boundary often correlate with nearby known classes. Learning these connections between known and unknown samples is essential.

1) Learning Inter-Sample Correlation: Although target unknown classes and their number are both agnostic during training, the synthesized known and unknown samples with diverse semantic and domain shift are sufficient for simulating the target domain. Given that all synthesized unknown samples exhibit some correlation with the source known classes used in their synthesis, we introduce a class-agnostic clustering method. This approach utilizes the similarity between synthesized unknown samples and K known classes to cluster the synthesized unknown samples into K clusters, effectively maximize the class margin between known and unknown classes along the classification boundary. we set the number of clusters for unknown samples to be K, and argue the k-th cluster corresponds to the k-th class in the known label space. Compared with other known classes, the k-th unknown cluster exhibits the highest similarity to the k-th known class. Meanwhile, unknown samples that are semantically unrelated to the known classes can be distinguished by minimizing Eq.11.

These two strategies simulates class correlations among target unknown classes without requiring knowing their quantity.

Our class-agnostic clustering includes inter-sample clustering and intra-sample clustering. Inspired by [43], inter-sample clustering first concatenate the outputs of C_k and C_u, and define $\mathbf{z}(x_s^i) = [C_k(F(x_s^i)), C_u(F(x_s^i))]$ for clustering with $2K$ heads. Two output vectors, $\mathbf{z}(x_s^i)$ and $\mathbf{z}(x_s^j)$, are chosen to compute cosine similarity matrix $S(x_s^i, x_s^j)$ for unlabeled input samples x_s^i and x_s^j. Then pseudo inter-sample label $h(x(i), x(j))$ can be obtained by setting a threshold μ on $S(x_s^i, x_s^j)$ (μ=0.9 by default), where

$$h(x_s^i, x_s^j) = \begin{cases} 1, \text{if } S(x_s^i, x_s^j) > \mu \\ 0, \text{otherwise} \end{cases}, \tag{14}$$

Given pseudo inter-sample label $h(x_s^i, x_s^j)$, we employ binary cross entropy loss to train the model as follows:

$$\begin{aligned} L_{inter}(x_s^i, \mathbf{x}_s^j) = \sum_{x_s^i, \mathbf{x}_s^j \in D_s} (L_{bce}(\mathbf{S}(x_s^i, x_s^j), h(x_s^i, x_s^j)) \\ + L_{bce}(\mathbf{S}(\mathbf{x}_s^j, \mathbf{x}_s^j), h(\mathbf{x}_s^j, \mathbf{x}_s^j))), \end{aligned} \tag{15}$$

here, we denote x_s^i and $\mathbf{x}_s^i$ as the original and augmented input samples, respectively. Specifically, the augmentation of $\mathbf{x}_s^i$ is achieved by applying randaugment, color jitter, and random horizontal flip to the samples, while x_s^i is obtained by only applying random horizontal flip to the samples, just like other samples during the training process. By learning inter-sample correlation across different augmentations, it can identify more robust and generalizable class-related characteristics.

2) Learning Intra-Sample Invariance: The inter-sample clustering explores generalizable category relationships, but it fails to fully exploit the invariance inherent within individual samples. Following [14], we address this by maximizing mutual information between augmented versions of the same sample for intra-sample clustering, denoted as:

$$L_{intra}(p(x_s^i), p(\mathbf{x}_s^i)) = \sum_{m=1}^{2K} \sum_{n=1}^{2K} P(x_s^i, \mathbf{x}_s^i) \log \left[\frac{P(x_s^i, \mathbf{x}_s^i)}{P(x_s^i) P(\mathbf{x}_s^i)} \right], \tag{16}$$

here, $p(x_s^i) = \text{Softmax}(z(x_s^i))$, and $P(x_s^i, \mathbf{x}_s^i)$ is the joint probability matrix: $P(x_s^i, \mathbf{x}_s^i) = \frac{1}{2K} \sum_{i=1}^{2K} p(x_i)\, p(\mathbf{x}_s^i)^T$, where m and n represent the m-th row and n-th column, respectively. The formula is optimized using both original samples x_s^i and their augmentations $\mathbf{x}_s^i$. This improves model generalization and produces more uniformly distributed clusters. Additionally, we minimize Eq.11 to enhance discrimination for unknown classes to the related known classes. The loss for class-aware minimization is:

$$\begin{aligned} L_{min}(x_s^i, \mathbf{x}_s^j, y_s^i) = \sum_{x_s^i, \mathbf{x}_s^j, y_s^i \in D_s} (\sigma(L_{inter}(x_s^i, \mathbf{x}_s^j) \\ - L_{intra}(x_s^i, \mathbf{x}_s^i)) + L_{os}(x_s^i, y_s^i)), \end{aligned} \tag{17}$$

σ is a trade-off and is set as 0.1. By minimizing Eq.17, CADA can classify both generalized seen and unseen samples well and explore the correlations among categories.

IV. Theoretical Analysis

The source domain is defined as D_s, they are used to synthesize unknown samples D_{su}. Define $L_e : F \times F \to \mathbb{R}^+ \cup \{\infty\}$ as the cost of perturbing embedding x_s to x_{su} in embedding space. $L_i : X \times X \to \mathbb{R}^+ \cup \{\infty\}$ is the cost of perturbing x_s to x_{su} in input space. The distance between D_s and D_{su} in embedding space is $d_{L_e}(\theta_{su}; D_s, D_{su}) := \inf_{M_F \in \Pi(D_s, D_{su})} \mathbb{E}_{M_F}[L_e(x_s, x_{su})]$, similarly, the distance between D_s and D_{su} in input space is $d_{L_i}(\theta_{su}; D_s, D_{su}) := \inf_{M_x \in \Pi(D_s, D_{su})} \mathbb{E}_{M_x}[L_i(x_s, x_{su})]$. M_F and M_x are measures in the embedding and input space respectively. $\Pi(D_s, D_{su})$ is joint distribution of D_s and D_{su}.

Analysis on unknown maximization. Unknown maximization synthesizes unknown samples D_{su} without introducing domain shift from D_s. D_{su} exhibits semantic differences from the D_s in embedding space, but possesses strong correlations with D_s in input space. Consequently, the goal of the worst-case problem for the unknown maximization stage becomes to synthesize the most challenge samples on boundary between known and unknown classes, which makes the model confused. The relaxed worst-case problem can be rewritten as:

$$\theta^* = \max_{\theta} \inf_{D_{su}} \mathbb{E}(L_{task}(\theta; D_{su})) = -\min_{\theta} \sup_{D_{su}} [-\mathbb{E}(L_{task}(\theta; D_{su}))], \tag{18}$$

where θ is parameters of the model. In unknown maximization, L_{task} is L_{uk}. D_{su} synthesizes unknown samples without domain shift. D_{su} exhibits both semantic differences from D_s in the semantic space, and correlations with the corresponding class D_s used to synthesize D_{su} in input space. Hence, $\{D_{su} : d_{L_i}(\theta_{su}; D_s, D_{su}) \leq \rho, d_{L_e}(\theta_{su}; D_s, D_{su} \geq \eta\}$, we use Lagrangian relaxation under constraint with fixed parameters $\lambda \geq 0$ and $\beta \geq 0$ to solve Eq. 18 as follows:

$$\begin{aligned} \theta^* = -\min_{\theta} \sup_{D_{su}} \{\mathbb{E}[-L_{uk}(\theta; D_{su})] - \lambda[W_{L_i}(\theta_{su}; D_{su}, D)] \\ + \beta[W_{L_e}(\theta_{su}; D_{su}, D_s)]\} = -\min_{\theta} \{\mathbb{E}[\delta_{\lambda,\beta}(\theta_{su}, \theta; x_{su}, x_s)]\}, \end{aligned} \tag{19}$$

where $\delta_{\lambda,\beta}(\theta_{su}, \theta; x_{su}, x_s) = \sup_{x_{su}} \{-L_{uk}(\theta_{su}, \theta; x_{su}, x_s) - \lambda W_{L_i}(\theta_{su}; x_{su}, x_s) + \beta W_{L_e}(\theta_{su}; x_{su}, x_s)\}$. The problem reduces to minimizing $\delta_{\lambda,\beta}$. As shown in [34], $\delta_{\lambda,\beta}$ is smooth with respect to θ if λ and β are large enoughtand the Lipschitz smoothness assumption holds. Gradient can be computed as:

$$\nabla_\theta \delta_{\lambda,\beta}(\theta; x_s) = \nabla_\theta \{\mathbb{E}[-L_{uk}(\theta; x_s^*(x_s, \theta))], \tag{20}$$

where $x_s^*(x_s, \theta) = \text{argmax}_{x_{su}} [\beta W_{L_e}(\theta_{su}; x_{su}, x_s) - \lambda W_{L_i}(\theta_{su}; x_{su}, x_s) - L_{uk}(\theta_{su}, \theta; x_{su}, x_s)] = \arg\max_{x_{su}} [L_{um}(\theta_{su}, \theta; x_{su}, x_s)]$, which is exactly unknown maximization in Eq. 9.

Analysis on out-of-distribution maximization. unknown samples within the source distribution belonging to D_{su} are seen as the $(K+1)$-th class and are append to D_s. Open-Set Single Domain generalization problem becomes Single Domain generalization problem with $K+1$ classes.

Similar to unknown maximization, out-of-distribution maximization synthesizes new samples for better generalization ability. But out-of-distribution maximization synthesizes new samples D_g that out of the source distribution with D_s. D_g exhibits domain shift from D_s in the input space , but possesses strong semantic similarity with D_s in semantic space. Consequently, the synthesised unknown samples encourage classifier to distinguish samples from both D_g and D_s well even if the sample selected from . The relaxed worst-case problem can be rewritten as:

$$\theta^* = \min_{\theta} \sup_{D_g} \mathbb{E}(L_{task}(\theta; D_g)), \tag{21}$$

TABLE II: Classification accuracy (%) on *Digits* with LeNet-5, on *Office31* and *VisDA2017* with ResNet-18. Best are in **bold**.

Methods	Access to D_t	Type	MNIST→ Others			Amazon→ Others			Synthesis → Real World			Real World → Synthesis			Average		
			acc	hs	acc_u	acc	hs	acc_u	acc	hs	acc_u	acc	hs	acc_u	acc	hs	acc_u
OSDAP[31]	Accessible	OSDA	57.3±0.3	54.2±0.4	49.6±0.2	76.1±0.3	79.6±0.2	84.7±0.3	54.7±0.3	59.2±0.5	69.1±0.4	68.6±0.3	50.9±0.2	38.6±0.3	61.7±0.3	55.1±0.4	53.9±0.4
CombEmb[16]	Accessible	OSDA	57.4±0.4	53.3±0.5	48.5±0.4	77.2±0.5	80.6±0.3	85.4±0.4	55.7±0.4	60.5±0.4	71.3±0.5	69.3±0.4	52.0±0.4	39.7±0.3	62.4±0.4	56.3±0.4	55.5±0.4
ERM[17]	Inaccessible	SL	49.2±0.3	18.0±0.4	13.0±0.3	77.9±0.4	40.7±0.3	21.1±0.5	43.3±0.4	30.4±0.5	23.4±0.6	64.5±0.4	25.6±0.4	15.5±0.4	53.9±0.3	28.0±0.4	19.4±0.4
PROSER[50]	Inaccessible	OSR	49.9±0.4	41.2±0.3	33.6±0.5	74.0±0.3	45.6±0.6	32.2±0.4	44.4±0.5	40.0±0.4	35.3±0.5	68.5±0.4	32.3±0.4	20.3±0.3	56.8±0.4	36.2±0.3	27.8±0.5
ADA[38]	Inaccessible	SDG	50.2±0.4	20.1±0.3	15.1±0.4	77.0±0.4	37.2±0.3	24.0±0.4	44.7±0.3	31.7±0.5	24.5±0.2	67.9±0.5	29.5±0.3	18.2±0.4	56.3±0.5	30.7±0.3	21.4±0.4
MEADA[49]	Inaccessible	SDG	52.9±0.4	30.4±0.5	29.8±0.3	76.8±0.4	35.0±0.5	22.2±0.3	45.1±0.4	33.7±0.5	25.7±0.3	68.2±0.4	28.7±0.5	17.6±0.3	56.6±0.4	31.2±0.5	21.6±0.3
CrossMatch[51]	Inaccessible	OS-SDG	51.3±0.4	38.7±0.5	46.1±0.3	76.6±0.4	52.8±0.4	39.3±0.5	44.4±0.3	42.5±0.2	40.8±0.5	69.5±0.4	53.0±0.5	40.9±0.5	57.0±0.3	47.8±0.5	40.8±0.4
CADA	Inaccessible	Ours	51.0±0.4	**49.5±0.3**	**47.2±0.4**	76.0±0.5	**57.7±0.5**	**45.4±0.4**	44.6±0.3	**45.5±0.5**	**47.0±0.3**	69.4±0.5	**56.2±0.3**	**45.1±0.4**	57.0±0.5	**50.8±0.3**	**46.1±0.4**

TABLE III: Classification accuracy (%) on *Office-Home* with ResNet-18.

Methods	Access to D_t	Type	Art → Others			Clipart→ Others			Product→ Others			Real World→ Others			Average		
			acc	hs	acc_u	acc	hs	acc_u	acc	hs	acc_u	acc	hs	acc_u	acc	hs	acc_u
OSDAP[31]	Accessible	OSDA	45.6±0.3	43.4±0.4	49.8±0.5	52.8±0.4	52.3±0.3	51.7±0.4	41.4±0.5	46.6±0.3	54.9±0.4	53.5±0.3	47.1±0.4	41.6±0.4	48.3±0.5	48.4±0.3	49.5±0.4
CombEmb[16]	Accessible	OSDA	46.5±0.5	47.7±0.4	49.2±0.5	51.9±0.4	52.7±0.5	53.7±0.4	54.2±0.4	55.8±0.3	55.3±0.3	66.3±0.3	59.1±0.5	52.8±0.4	54.7±0.4	53.8±0.5	52.8±0.4
ERM[17]	Inaccessible	SL	65.0±0.4	31.1±0.5	20.5±0.6	64.1±0.4	35.8±0.6	24.7±0.5	60.5±0.4	36.3±0.4	26.3±0.5	66.6±0.4	33.9±0.5	23.2±0.5	64.1±0.4	34.3±0.4	23.7±0.5
PROSER[50]	Inaccessible	OSR	62.6±0.4	47.5±0.3	37.6±0.5	60.0±0.5	38.8±0.3	28.2±0.4	59.9±0.5	42.1±0.5	31.9±0.5	64.9±0.5	49.7±0.6	39.7±0.4	61.9±0.4	44.5±0.5	34.3±0.4
ADA[38]	Inaccessible	SDG	68.3±0.4	32.9±0.5	22.1±0.4	65.1±0.4	42.1±0.4	31.2±0.3	60.5±0.5	34.7±0.4	24.6±0.5	67.1±0.3	34.9±0.3	22.9±0.5	65.2±0.4	36.2±0.5	25.4±0.4
MEADA[49]	Inaccessible	SDG	68.3±0.6	33.3±0.5	22.4±0.4	65.3±0.5	42.1±0.4	31.3±0.5	60.4±0.4	35.7±0.4	25.6±0.5	67.0±0.4	34.7±0.4	23.7±0.5	65.0±0.3	36.4±0.4	25.7±0.5
CrossMatch[51]	Inaccessible	OS-SDG	65.9±0.4	53.2±0.3	45.3±0.5	62.9±0.4	48.9±0.5	37.8±0.4	58.4±0.3	45.3±0.5	37.7±0.6	67.1±0.4	50.8±0.4	41.3±0.3	63.6±0.5	49.6±0.5	40.5±0.3
CADA	Inaccessible	Ours	65.8±0.4	**54.9±0.3**	**46.4±0.5**	59.0±0.3	**53.4±0.3**	**48.3±0.4**	60.1±0.3	**52.4±0.4**	**45.8±0.4**	65.0±0.3	**56.3±0.4**	**49.1±0.5**	62.5±0.4	**54.2±0.3**	**47.4±0.4**

TABLE IV: Classification accuracy (%) on *PACS* with ResNet-18.

Methods	Access to D_t	Type	Art Paint→ Others			Cartoon→ Others			Photo→ Others			Sketch→ Others			Average		
			acc	hs	acc_u	acc	hs	acc_u	acc	hs	acc_u	acc	hs	acc_u	acc	hs	acc_u
OSDAP[31]	Accessible	OSDA	36.5±0.3	36.3±0.4	76.6±0.5	32.7±0.4	32.3±0.4	67.1±0.4	36.4±0.5	36.0±0.4	74.3±0.3	34.3±0.4	33.8±0.5	32.9±0.3	35.0±0.3	34.6±0.4	62.7±0.5
CombEmb[16]	Accessible	OSDA	48.4±0.4	53.5±0.5	72.2±0.4	52.3±0.3	56.4±0.5	67.4±0.5	42.6±0.3	49.1±0.3	75.1±0.4	46.4±0.3	50.3±0.4	61.3±0.4	47.4±0.3	52.3±0.4	69.0±0.4
ERM[17]	Inaccessible	SL	46.2±0.4	32.7±0.3	23.5±0.5	49.0±0.4	23.5±0.4	15.0±0.4	34.7±0.4	26.7±0.5	20.2±0.3	41.3±0.4	30.7±0.4	22.7±0.5	42.8±0.4	28.4±0.3	20.3±0.3
PROSER[50]	Inaccessible	OSR	48.6±0.4	39.4±0.5	30.8±0.5	49.3±0.3	35.0±0.3	25.2±0.4	36.5±0.5	33.8±0.3	30.0±0.5	42.7±0.6	41.6±0.4	39.8±0.3	44.3±0.5	37.4±0.4	31.4±0.5
ADA[38]	Inaccessible	SDG	47.7±0.5	35.9±0.5	26.8±0.4	49.3±0.4	26.8±0.4	17.2±0.5	37.0±0.4	29.2±0.5	22.4±0.4	43.2±0.5	33.7±0.4	25.7±0.3	44.3±0.4	31.4±0.5	22.9±0.4
MEADA[49]	Inaccessible	SDG	48.1±0.4	36.5±0.5	27.4±0.3	48.8±0.4	27.6±0.5	18.0±0.3	36.9±0.3	28.7±0.5	21.8±0.3	42.8±0.4	32.6±0.5	24.4±0.3	44.1±0.4	31.3±0.4	22.9±0.5
CrossMatch[51]	Inaccessible	OS-SDG	48.5±0.4	41.0±0.5	33.2±0.5	49.6±0.3	37.5±0.5	30.1±0.4	34.2±0.3	33.0±0.4	31.1±0.5	43.2±0.3	**45.1±0.5**	50.6±0.3	43.9±0.4	39.2±0.3	36.2±0.5
CADA	Inaccessible	Ours	49.2±0.4	**43.0±0.5**	**35.8±0.3**	49.9±0.5	**42.2±0.4**	**34.2±0.3**	34.3±0.4	**35.1±0.4**	**37.2±0.4**	42.1±0.4	45.0±0.3	**57.2±0.5**	43.9±0.3	**41.3±0.3**	**41.1±0.4**

where θ is parameters of the model. In out-of-distribution maximization, L_{task} is L_{os}. D_g generates both unknown and known samples out of the source distribution. D_{su} not only exhibits semantic differences from D_s in the input space, but also maintains semantic correlations with D_s in the semantic space. Hence, $\{D_g : W_{L_i}(D_s, D_g) \geq \rho, W_{L_e}(D_s, D_g) \leq \eta\}$. It is hard to solve Eq. 21, so we use Lagrangian relaxation under constraint with fixed parameters $\lambda \geq 0$ and $\beta \geq 0$:

$$\begin{aligned}\theta^* = \min_{\theta} \sup_{D_g} \{\mathbb{E}[L_{os}(\theta; D_g)] - \lambda[W_{L_i}(\theta_{s_g}; D_s, D_g)] \\ -\beta[W_{L_e}(\theta_{s_g}; D_s, D_g)]\} = \min_{\theta} \{\mathbb{E}[\delta_{\lambda,\beta}(\theta; x_s)]\},\end{aligned} \tag{22}$$

where $\delta_{\lambda,\beta}(\theta; x_s) = \sup_{x_g} \{L_{os}(\theta; x_g) - \lambda W_{L_i}(x_s, x_g) - \beta W_{L_e}(x_s, x_g)\}$. The problem becomes minimizing $\delta_{\lambda,\beta}$. $\delta_{\lambda,\beta}$ is smooth w.r.t. θ if λ, β are large enough and the assumption of Lipschitzian smoothness holds. The gradient is computed as:

$$\nabla_\theta \delta_{\lambda,\beta}(\theta; x_s) = \nabla_\theta \{\mathbb{E}[L_{os}(\theta; x_s^*(x_s, \theta))], \tag{23}$$

where $x_s^*(x_s, \theta) = \text{argmax}_{x_g}[L_{os}(\theta, \theta_{s_g}; x_s, x_g) - \lambda W_{L_i}(\theta_{s_g}; x_s, x_g) - \beta W_{L_e}(\theta_{s_g}; x_s, x_g)] \leq \text{argmax}_{x_{su}}[L_{os}(\theta, \theta_{s_g}; x_s, x_g) - \beta W_{L_e}(\theta_{s_g}; x_s, x_g)] = \arg\max_{x_g}[L_{om}(\theta, \theta_{s_g}; x_s, x_g)]$, which is exactly out-of-distribution maximization.

V. Experiments

A. Setup

1) Dataset: Experiments are conducted on 7 datasets. **Digits** dataset includes: MNIST [18], USPS [13], SVHN [46], MNIST-M and SYN [7]. **Office-Home** dataset [37] contains Art, Clipart, Product and Real World domains. **Office31** [30] dataset has Amazon, Dslr, and Webcam domains. **VisDA-2017** [27] dataset comprises from Synthesis and Real World. **PACS** [19] dataset has 4 domains: Art Paint, Cartoon, Sketch and Photo. Face Anti-spoofing tasks with **CASIA-MFSD** [48] (C), **Replay-Attack** [4] (R) and **HQ-WMCA** [10] (H).

2) Evaluation Protocol: For the Digits dataset, we follow CrossMatch [51] and select MNIST as the source domain, consisting of numbers 0 to 4. The remaining four datasets are used as target domains with a common unknown label space. The target unknown label space $\mathcal{C}_t^u$ consists of unknown categories ranging from 5 to 9. For Office-Home dataset, we follow CrossMatch [51] and designate the first 15 categories as the shared label space, and the remaining 50 categories as the target unknown label space. Each domain is sequentially selected as the source domain, with the remaining three domains serving as the target domains. On the VisDA-2017 dataset, we evaluate our method using two domains as the source in turn. The first five classes in alphabetical order are considered as the source domain label space, and all 12 classes are considered as the target space. We further conduct experiments on Office31 and PACS datasets with inconsistent target domain label spaces. On the Office31 dataset, Amazon is seen as the source domain and the remaining two as the target. The source domain contains 10 classes shared by Office31 and Caltech256, while target unknown label space $\mathcal{C}_t^u$ of Webcam includes the last 10 classes sorted alphabetically, and target label space of Dslr includes all the 31 categories. For the PACS dataset, each of the four domains is used as the source in turn, with the others as targets. The source domain label space consists of the first three classes alphabetically, while each target domain has a different set of classes ($\mathcal{C}_t$). The first target domain includes 5 classes, the second has 6, and the last contains 7 classes. For the Face Anti-spoofing, C and R are used as source domains in turn, and H domain is the target.

3) Pytorch Implementation Details: LeNet-5 is used as the backbone for Digits dataset, while a ResNet18 pre-trained is used on ImageNet as the baseline for the rest. Following SHOT [22], we place a BN layer after the FC layer inside the

TABLE V: Experiments on Face Anti-Spoofing

Method	C to H		I to H	
	HTER (%) ↓	AUC (%) ↑	HTER (%) ↓	AUC (%) ↑
PatchCNN [42]	39.54	64.54	35.03	73.24
PDEN [22]	35.76	69.10	31.03	74.35
LD [42]	38.33	65.12	32.43	73.57
PCGRL [15]	27.24	78.81	28.03	79.37
Ours	31.45	72.36	30.55	75.83

TABLE VII: Comparison on Office-Home and VisDA2017

Methods	Venue	Office–Home			VisDA2017		
		acc	h_s	acc_u	acc	h_s	acc_u
NormAuG+PROSER [28, 50]	TIP24	60.5	51.2	43.8	45.3	40.3	40.8
MEDIC [1]	ICCV23	58.2	44.8	35.8	53.3	45.1	37.6
ODG-CLIP [33]	CVPR24	61.3	53.0	46.1	54.7	48.9	42.9
CADA	Ours	**62.5**	**54.2**	**47.4**	**57.0**	**50.8**	**49.1**

bottleneck layer and employ a weight normalization layer in the last FC layer. Mini-batch SGD is adpoted with momentum 0.9 and weight decay 1e-3, and set the learning rate as 1e-4. Batch size is 64. We use cropping [40] and RandAugment [5] to augment the training data x_s to $\mathbf{x}_s$. CADA compares with other methods following their original procedures under our settings except DA method accessing target data during training. Time and space complexity are both O(N).

4) Comparative Evaluations and Performance Metrics: CADA is compared against various baseline and SOTA methods includes supervised learning (SL) method Empirical Risk Minimization (ERM) [17], Open-Set DA method Open-Set Domain Adaptation by Back propagation (OSDAP) [31], Open-Set recognition (OSR) method PROSER [50] and CombEmb [16], state-of-the-art SDG methods Adversarial Data Augmentation (ADA) [38] and Maximum-Entropy Adversarial Data Augmentation (MEADA) [49] , and state-of-the-art OS-SDG method CrossMatch (CrossMatch) [51] on five datasets. In this paper, we use acc,hs and acc_u as our three main metrics. Since in our setting, almost half data are unknown samples, while per-class accuracy (acc) is the mean accuracy averaged over all classes in all $K+1$ classes, unknown samples are seen as the $K+1$ class. Therefore, treating unknown samples, which account for over half of the dataset, equally with the rest of the known classes is unfair. While h-score (hs), $hs = \frac{2*acc_u*acc_k}{acc_u+acc_k}$, is the harmonic mean of average per-class accuracy in known and unknown space, which give equal importance weight to known and unknown classes. This setting is more consistent with our practical situation, as the hs score will only be high when both known and unknown accuracies are high. Here, acc_k and acc_u are per-class accuracy of known and unknown label space.

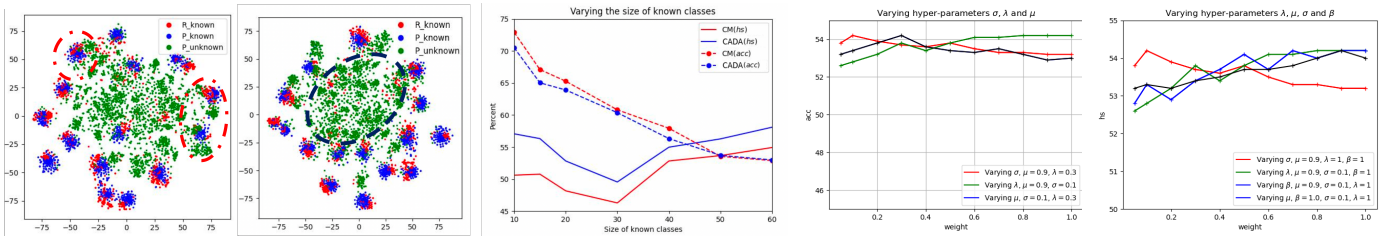
B. Results and Further Analysis

1) Experiment Result: Table II, III and IV reports the performance on the Digits, Office31, VisDA-2017, Office-Home and PACS datasets under the OS-SDG setting, respectively. OSDA performs well on acc_u. However, it uses both source and target data in training. Thanks to L_{os}, PROSER can distinguish some semantic unrelated unknown samples, but fails to classify target known ones well due to distribution shift. SDG methods are effective in improving accuracy on known classes. But they struggle on unknown classes. This leads to low accuracy on unknown classes and low hs. The SOTA OS-SDG method, CrossMatch, improves the discrimination of unknown samples by synthesizing them, but its assumption that these samples are entirely unrelated to known classes limits its open-set discriminative power. Our CADA outperforms CrossMatch significantly in hs and acc_u through all the datasets. It shows that CADA significantly improves generalization over existing methods on unknown classes while maintaining accuracy on known classes. More analysis is in supplemental materials.

2) Component Effectiveness Evaluation: We conduct an ablation study in Table VI to evaluate L_{div}, L_{uk}, L_{intra}, and L_{inter}. The first row is the baseline, a CADA variant using only L_{os} without these four components. Although it retains strong discriminative power for known classes, its acc_u and h_s remain relatively low, indicating limited ability to handle unknown classes. Comparing the first and last rows shows that while L_{os} can only distinguish unrelated unknown samples, the full CADA model can differentiate both related and unrelated unknowns, validating our approach. The difference between the first and second rows highlights that L_{intra} and L_{inter} help capture class correlations, improving performance. The comparison between the fourth and last rows demonstrates that L_{div} enhances the diversity of synthesized unknown samples, contributing to better results. Additionally, from the first, second, and last rows, we observe that L_{uk} strengthens discriminative power for unknown samples by generating diverse, known-related unknowns. The last four rows highlight how L_{intra} and L_{inter} explore class relationships.

3) Applicability and Comparison with Other Methods: We evaluate CADA's applicability on anti-spoofing task in Table V. Our method outperforms SDG methods [20, 39, 42], which can be attributed to its strong open-set discrimination capability. However, compared to PCGRL [15], an OS-SDG method specifically designed for this scenario, our method performs less effectively. This is because our CADA primarily focuses on relative relationships between classes without capturing fine-grained, attribute-level features. Such limitations hinder performance on face anti-spoofing tasks, which heavily rely on local detail features. Table VII compare CADA with more SOTA approaches. It shows that even SOTA SDG methods, when augmented with open-set recognition modules, experience significant performance drops when target domain data is inaccessible. This highlights both the challenge and the practical significance of our OS-SDG setting. In real-world scenarios, open-set domain generalization methods often assume multiple source domains—an assumption that rarely holds true. When these methods are applied in a single-source setting, their performance deteriorates markedly. Notably, even ODG-CLIP, which leverages the strong generalization ability of CLIP, underperforms compared to our proposed CADA. This further substantiates the superiority of our approach and the challenges inherent in the OS-SDG setting.

4) Visualization: We conduct t-SNE [36] under the OS-SDG setting with Real World domain → other task, Fig. 3a show the comparison between CrossMatch and CADA. Red points are source known classes, while blue and green are target known and unknown classes. CrossMatch performs better but sometimes misclassifies unknown samples into known clusters (red circles) due to difficulty in identifying semantically related unknowns. In contrast, CADA separates target known samples more accurately and effectively distinguishes

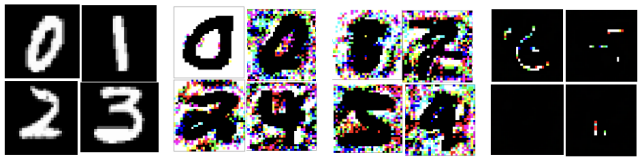


(a) t-SNE for CrossMatch (left) and CADA (right) (b) Number of known classes (c) Hyper-parameter λ, θ, μ (d) Convergence plot

Fig. 3: Ablation study under OS-SDG setting. (a) Feature visualization on "Real World→Others" task. (b) Number of categories known on *Office-Home* dataset. (c) Varying hyper-parameters λ, θ and μ. (d) Convergence plot on *Office-Home* dataset.

TABLE VI: Ablation study of variants with our CADA on *Office-Home* dataset.

CADA's variant				settings on Office-Home											
				Art → Others			Clipart→ Others			Product→ Others			Real World→ Others			Average		
L_{div}	L_{uk}	L_{intra}	L_{inter}	acc	hs	acc_u	acc	hs	acc_u	acc	hs	acc_u	acc	hs	acc_u	acc	hs	acc_u
		✓	✓	65.3	43.7	32.3	58.2	45.2	36.5	60.4	42.0	31.7	65.9	47.4	36.4	62.4	44.6	34.2
		✓	✓	65.3	44.9	34.2	58.0	44.2	35.8	**60.5**	42.4	32.7	**66.0**	47.6	37.3	62.4	44.8	34.9
	✓			65.1	54.5	46.2	56.6	49.9	44.2	59.5	50.1	42.8	64.9	55.1	47.3	61.5	52.4	45.1
	✓	✓	✓	64.8	53.2	44.5	58.3	52.0	46.5	57.9	50.7	44.6	64.1	56.4	**49.7**	61.3	53.1	46.3
✓	✓		✓	64.0	53.1	45.3	58.83	51.84	46.3	58.5	48.8	41.8	63.6	55.8	**49.7**	61.2	52.4	45.8
✓	✓	✓		64.8	54.4	46.2	58.3	51.9	46.2	59.6	50.3	42.9	64.7	**56.4**	49.4	61.9	53.2	46.2
✓	✓	✓	✓	**65.8**	**54.9**	**46.4**	**59.0**	**53.4**	**48.3**	60.1	**52.4**	**45.8**	65.0	56.3	49.1	**62.5**	**54.2**	**47.4**



(a) original known samples (b) augmented known samples (c) augmented unknown samples (d) bad cases

Fig. 4: Visualization of synthesized samples.

TABLE VIII: Impact of different class ratios.

Source Known	Source Unknown	OOD Known	OOD Unknown	hs
1	1	1	1	54.9
1	0	1	0	25.6
1	1	1	0	36.8
1	0	1	1	50.8
1	0.5	1	0.5	52.4
1	0.5	0.5	0.5	51.3

TABLE IX: Accuracy vs. Complexity Comparison

Method	hs	Accuracy (Unknown)	Training Time (hours)	Inference Time (ms)	FLOPs (G)
CrossMatch	47.8%	40.5%	3.8	14.7	2.1
CADA (Ours)	50.8%	47.4%	5.2	14.9	2.2

both related and unrelated unknown samples (black circle).

5) Varying number of known classes: Fig. 3b illustrates the adaptation task from the Real World to other domains, with the number of known classes ranging from 10 to 60. The red curves represent the performance of CrossMatch, while the blue curves correspond to CADA. Both models exhibit sensitivity to the number of known classes, with performance generally declining as the number increases. However, CADA consistently outperforms CrossMatch, demonstrating superior differentiation of unknown samples. This advantage leads to a higher harmonic mean score (hs) in terms of classifiability and separability while maintaining comparable accuracy levels.

6) Hyper-parameter sensitivity: In Fig. 3c, we conduct an ablation study on the Office-Home dataset under the OS-SDG setting, exploring the impact of varying hyper-parameters λ, θ, and μ on model performance. The evaluation metric used is hs. The figure shows three curves representing different hyper-parameter variations: the red curve varies σ with $\mu = 0.9$ and $\lambda = 0.3$, the green curve varies λ with $\mu = 0.9$ and $\sigma = 0.1$, and the blue curve varies μ with $\sigma = 0.1$ and $\lambda = 0.3$. The results show that CADA maintains stable performance across different settings, demonstrating its robustness to these hyper-parameters. Specifically, varying σ (red curve) causes mild fluctuations, with a slight peak around a weight of 0.3, suggesting an optimal region for σ. Adjusting λ (green curve) shows an increasing trend, indicating its importance for regularization. Changes in μ (blue curve) result in a relatively stable trend, highlighting the model's resilience to variations in this parameter. Overall, Fig. 3c shows that CADA is robust to hyper-parameter changes, validating the model's stability.

7) Impact of different class ratios: As shown in Table VIII, the first row represents the full CADA model, achieving the best performance by generating sufficient known and unknown samples to enhance generalization and open-set recognition. The second row, which removes all unknown samples, sharply reduces open-set recognition ability, lowering h_s despite strong known-class performance. In the third row, removing only out-of-distribution unknowns slightly improves performance over the second row but remains far below the full model. The fourth row removes source unknowns, causing a moderate performance drop, as Eq.(6) relies on class relationships, but the impact is limited. The last two rows, with adjusted generation ratios, show performance declines, highlighting the importance of each module in our approach.

8) Accuracy vs complexity comparison: We added an "Accuracy vs. Complexity" comparison with baseline CrossMatch, as shown in Table IX. Our method improves unknown accuracy by +6.9% and achieves a 3% higher hs compared to CrossMatch. Although the two-stage maximization increases

training time by approximately 1.5 hours, inference time and FLOPs remain similar to the baseline.

9) Iteration of training: The convergence plot is depicted in Fig. 3d. For the Office-Home dataset, the training includes 2 stages with 200 epochs. The result exhibits convergence stability, our training includes two phases. Specifically, the first phase aims to synthesize the fictitious target domain to simulate the real target domain. During this phase, we first utilizes supervised learning with L_{os} to learn open-set recognition ability. The loss curve converges stablely. Then we conduct unknown maximization and minimization to synthesize classes with both domain shift and label space shift to simulate the inaccessible target domain. The loss curse is dynamically changing. During the second phase, synthesized samples appends to the source domain, model is modified with both class-agnostic clustering and modified supervised learning with L_{os} to learn generalized open-set discrimination ability, the loss curve converges stablely.

10) Synthesized samples: Fig. 4 displays the synthesized samples—both known and unknown—from the MNIST dataset. The synthesized known samples exhibit a distinct domain gap compared to the original images, yet they successfully preserve semantic integrity. In contrast, the synthesized unknown samples, specifically from digits 0, 2, 3, and 4, demonstrate both domain and semantic variations when compared to their original counterparts. Despite these differences, there remains a significant correlation with the original samples, indicating that the essential characteristics of the digits are still recognizable. This balance highlights the effectiveness of the synthesis process in generating new, yet contextually related samples that expand the training dataset while maintaining a connection to the original data.

VI. Conclusion

This work presents Class-Aware Diversified Augmentation, which synthesizes more realistic unknown samples semantically correlated with the source known classes to simulate unseen target domains for generalization. Theoretical analysis and experiments on benchmarks show its superiority.

Future Works and Limitations CADA has some limitations for future research. First, our method requires a two-stage sample generation. Although it produces more diverse samples than previous methods, reducing generation steps and resource consumption remains a challenge. Second, our method currently constrains semantics through simple inter-class relations; in the future, inspired by [23, 24], we aim to incorporate part-whole relational properties to construct finer-grained inter-class relationships for better synthesization.

References

[1] Shirsha Bose, Ankit Jha, Hitesh Kandala, and Biplab Banerjee. Beyond boundaries: A novel data-augmentation discourse for open domain generalization. *Transactions on Machine Learning Research*, 2023.

[2] Xinyang Chen, Sinan Wang, Mingsheng Long, and Jianmin Wang. Transferability vs. discriminability: Batch spectral penalization for adversarial domain adaptation. In *International conference on machine learning*, pages 1081–1090. PMLR, 2019.

[3] Zining Chen, Weiqiu Wang, Zhicheng Zhao, Fei Su, Aidong Men, and Hongying Meng. Practicaldg: Perturbation distillation on vision-language models for hybrid domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23501–23511, 2024.

[4] Ivana Chingovska, André Anjos, and Sébastien Marcel. On the effectiveness of local binary patterns in face anti-spoofing. In *2012 BIOSIG-proceedings of the international conference of biometrics special interest group (BIOSIG)*, pages 1–7. IEEE, 2012.

[5] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 702–703, 2020.

[6] Abolfazl Farahani, Sahar Voghoei, Khaled Rasheed, and Hamid R Arabnia. A brief review of domain adaptation. *Advances in Data Science and Information Engineering: Proceedings from ICDATA 2020 and IKE 2020*, pages 877–894, 2021.

[7] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *International conference on machine learning*, pages 1180–1189. PMLR, 2015.

[8] Kai Han, An Xiao, Enhua Wu, Jianyuan Guo, Chunjing Xu, and Yunhe Wang. Transformer in transformer. *Advances in Neural Information Processing Systems*, 34:15908–15919, 2021.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning. *Image Recognition*, 7, 2015.

[10] Guillaume Heusch, Anjith George, David Geissbühler, Zohreh Mostaani, and Sébastien Marcel. Deep models and shortwave infrared information to detect face presentation attacks. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 2(4):399–409, 2020.

[11] Jian Hu, Hongya Tuo, Chao Wang, Lingfeng Qiao, Haowen Zhong, Junchi Yan, Zhongliang Jing, and Henry Leung. Discriminative partial domain adversarial network. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVII 16*, pages 632–648. Springer, 2020.

[12] Jian Hu, Haowen Zhong, Fei Yang, Shaogang Gong, Guile Wu, and Junchi Yan. Learning unbiased transferability for domain adaptation by uncertainty modeling. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXI*, pages 223–241. Springer, 2022.

[13] Jonathan J. Hull. A database for handwritten text recognition research. *IEEE Transactions on pattern analysis and machine intelligence*, 16(5):550–554, 1994.

[14] Xu Ji, Joao F Henriques, and Andrea Vedaldi. Invariant information clustering for unsupervised image classification and segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9865–9874, 2019.

[15] Fangling Jiang, Qi Li, Weining Wang, Min Ren, Wei Shen, Bing Liu, and Zhenan Sun. Open-set single-domain generalization for robust face anti-spoofing. *International Journal of Computer Vision*, pages 1–22, 2024.

[16] Geeho Kim, Junoh Kang, and Bohyung Han. Open-set representation learning through combinatorial embedding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19744–19753, 2023.

[17] Vladimir Koltchinskii. *Oracle inequalities in empirical risk minimization and sparse recovery problems: École D'Été de Probabilités de Saint-Flour XXXVIII-2008*, volume 2033. Springer Science & Business Media, 2011.

[18] Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard, and Lawrence D Jackel. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551, 1989.

[19] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE international conference on com-*

puter vision, pages 5542–5550, 2017.

[20] Lei Li, Ke Gao, Juan Cao, Ziyao Huang, Yepeng Weng, Xiaoyue Mi, Zhengze Yu, Xiaoya Li, and Boyang Xia. Progressive domain expansion network for single domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 224–233, 2021.

[21] Shuang Li, Kaixiong Gong, Binhui Xie, Chi Harold Liu, Weipeng Cao, and Song Tian. Critical classes and samples discovering for partial domain adaptation. *IEEE Transactions on Cybernetics*, 53(9):5641–5654, 2022.

[22] Jian Liang, Dapeng Hu, and Jiashi Feng. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In *International Conference on Machine Learning*, pages 6028–6039. PMLR, 2020.

[23] Yi Liu, De Cheng, Dingwen Zhang, Shoukun Xu, and Jungong Han. Capsule networks with residual pose routing. *IEEE Transactions on Neural Networks and Learning Systems*, 2024.

[24] Yi Liu, Dingwen Zhang, Qiang Zhang, and Jungong Han. Part-object relational visual saliency. *IEEE transactions on pattern analysis and machine intelligence*, 44(7):3688–3704, 2021.

[25] Krikamol Muandet, David Balduzzi, and Bernhard Schölkopf. Domain generalization via invariant feature representation. In *International conference on machine learning*, pages 10–18. PMLR, 2013.

[26] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1406–1415, 2019.

[27] Xingchao Peng, Ben Usman, Neela Kaushik, Judy Hoffman, Dequan Wang, and Kate Saenko. Visda: The visual domain adaptation challenge. *arXiv preprint arXiv:1710.06924*, 2017.

[28] Lei Qi, Hongpeng Yang, Yinghuan Shi, and Xin Geng. Normaug: Normalization-guided augmentation for domain generalization. *IEEE Transactions on Image Processing*, 33:1419–1431, 2024.

[29] Fengchun Qiao, Long Zhao, and Xi Peng. Learning to learn single domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12556–12565, 2020.

[30] Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting visual category models to new domains. In *Computer Vision–ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part IV 11*, pages 213–226. Springer, 2010.

[31] Kuniaki Saito, Shohei Yamamoto, Yoshitaka Ushiku, and Tatsuya Harada. Open set domain adaptation by backpropagation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 153–168, 2018.

[32] Yang Shu, Zhangjie Cao, Chenyu Wang, Jianmin Wang, and Mingsheng Long. Open domain generalization with domain-augmented meta-learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9624–9633, 2021.

[33] Mainak Singha, Ankit Jha, Shirsha Bose, Ashwin Nair, Moloud Abdar, and Biplab Banerjee. Unknown prompt the only lacuna: Unveiling clip's potential for open domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13309–13319, 2024.

[34] Aman Sinha, Hongseok Namkoong, Riccardo Volpi, and John Duchi. Certifying some distributional robustness with principled adversarial training. *arXiv preprint arXiv:1710.10571*, 2017.

[35] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7167–7176, 2017.

[36] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

[37] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5018–5027, 2017.

[38] Riccardo Volpi, Hongseok Namkoong, Ozan Sener, John C Duchi, Vittorio Murino, and Silvio Savarese. Generalizing to unseen domains via adversarial data augmentation. *Advances in neural information processing systems*, 31, 2018.

[39] Fei Wang, Xing Zhang, Yong Jiang, Li Kong, and Xiaotong Wei. Patchcnn: An explicit convolution operator for point clouds perception. *IEEE Geoscience and Remote Sensing Letters*, 18(4):726–730, 2020.

[40] Meng Wang, Changzhi Luo, Richang Hong, Jinhui Tang, and Jiashi Feng. Beyond object proposals: Random crop pooling for multi-label image recognition. *IEEE Transactions on Image Processing*, 25(12):5678–5688, 2016.

[41] Xiran Wang, Jian Zhang, Lei Qi, and Yinghuan Shi. Generalizable decision boundaries: Dualistic meta-learning for open set domain generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11564–11573, 2023.

[42] Zijian Wang, Yadan Luo, Ruihong Qiu, Zi Huang, and Mahsa Baktashmotlagh. Learning to diversify for single domain generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 834–843, 2021.

[43] Jianlong Wu, Keyu Long, Fei Wang, Chen Qian, Cheng Li, Zhouchen Lin, and Hongbin Zha. Deep comprehensive correlation mining for image clustering. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8150–8159, 2019.

[44] Zhiling Yan, Kai Zhang, Rong Zhou, Lifang He, Xiang Li, and Lichao Sun. Multimodal chatgpt for medical applications: an experimental study of gpt-4v. *arXiv preprint arXiv:2310.19061*, 2023.

[45] Kaichao You, Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Universal domain adaptation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2720–2729, 2019.

[46] Netzer Yuval. Reading digits in natural images with unsupervised feature learning. In *Proceedings of the NIPS Workshop on Deep Learning and Unsupervised Feature Learning*, 2011.

[47] Kun Zhang, Mingming Gong, and Bernhard Schölkopf. Multi-source domain adaptation: A causal view. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29, 2015.

[48] Zhiwei Zhang, Junjie Yan, Sifei Liu, Zhen Lei, Dong Yi, and Stan Z Li. A face antispoofing database with diverse attacks. In *2012 5th IAPR international conference on Biometrics (ICB)*, pages 26–31. IEEE, 2012.

[49] Long Zhao, Ting Liu, Xi Peng, and Dimitris Metaxas. Maximum-entropy adversarial data augmentation for improved generalization and robustness. *Advances in Neural Information Processing Systems*, 33:14435–14447, 2020.

[50] Da-Wei Zhou, Han-Jia Ye, and De-Chuan Zhan. Learning placeholders for open-set recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2021.

[51] Ronghang Zhu and Sheng Li. Crossmatch: Cross-classifier consistency regularization for open-set single domain generalization. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022.