AOVANet: Adaptive One-vs-All Network for Universal Domain Adaptation

1st Yuntao Du
C-FAIR & School of software
Shandong University
Jinan, China
yuntaodu@sdu.edu.cn

2nd Huiming Li
C-FAIR & School of software
Shandong University
Jinan, China
202535700@mail.sdu.edu.cn

3rd Mujie zhang
School of computer science
Nanjing University
Nanjing, China
mujiezhang@smail.nju.edu.cn

4th Mingcai chen
school of computer science
Nanjing University of Posts and Telecommunications
City, Country
email address or ORCID

5th Lizhen Cui*
C-FAIR & School of software
Shandong University
Jinan, China
clz@sdu.edu.cn

***Abstract*—Universal Domain Adaptation (UniDA) aims at handling both domain and category shifts between different domains. Its primary goal is to transfer the learned knowledge of the source domain to the target domain and, moreover, identify "unknown" classes in the target domain. Recently, methods leveraging multiple one-vs-all classifiers with open-set entropy minimization have emerged to detect unknown classes, eliminating the need to set thresholds or proportionally reject unknown classes manually. However, this method overlooks the fact that a target sample contributes differently to the one-vs-all classifiers, leading to sub-optimal adaptation performance. Besides, we experimentally found that the performance of this method is heavily influenced by the number of categories in the source domain, with larger class numbers resulting in degraded results. To address these limitations, we propose a simple yet effective solution: Adaptive One-Vs-All Network (AOVANet). This method dynamically assigns weights to each one-vs-all classifier based on its importance to a given sample, enabling the model to focus more effectively on the more relevant classifiers adaptively. We conducted extensive experiments on both UniDA and open-set domain adaptation settings on three cross-domain image classification datasets. AOVANet achieves significant performance gains of** 5.6%, 0.4%, **and** 7.1% **on Office-31, Office-Home, and VisDA than OVANet, validating its effectiveness.**

***Index Terms*—Domain Adaptation, Open-set DA**

I. Introduction

Domain adaptation aims to transfer knowledge from a labeled source domain to an unlabeled target domain. While both domains share the same task, they typically originate from different data distributions. Traditional unsupervised domain adaptation assumes that the source and target domains have identical label sets, but this assumption does not always hold in practice. Based on the category relationship of the source domain and the target domain, domain adaptation can be divided into Closed-set adaptation(CDA) [1]–[4], Open-set Domain Adaptation (ODA) [5]–[7], Partial Domain Adaptation (PDA) [8], and Universal Domain Adaptation (UniDA) [9]. Among them, universal domain adaptation (UniDA) is introduced to handle cases where the label distributions between the source and target domains differ, but the specific discrepancies are unknown. In such situations, both domains may contain private classes. As a result, the model must not only perform distribution matching to address the domain shift across the common label sets, but also identify and label unknown classes as "unknown", without prior knowledge of their existence.

Existing methods typically employ three distinct strategies to address the challenge of unknown class detection in UniDA. The first approach involves setting explicit thresholds to identify unknown classes, as seen in models like UAN [9], DANCE [10], and CPR [11]. These methods rely on manually defined thresholds without prior data from the target domain, introducing potential subjective bias and undermining accuracy. The second strategy uses categorical rejection of unknown classes at a fixed ratio, as demonstrated by ROS [6]. However, this method is also limited, as its effectiveness diminishes when the proportion of unknown samples fluctuates.

To address these limitations, a new set of methods uses adaptive threshold learning to identify unknown classes more effectively [12], [13]. Among these, OVANet [12] is a pioneering approach. It assumes that the minimum inter-class distance serves as a reliable threshold for determining whether a sample belongs to a class. Specifically, OVANet trains a one-vs-all classifier for each class in the source domain, where each classifier determines whether a sample belongs to the corresponding class. Additionally, OVANet introduces open-set entropy minimization for unlabeled target samples, minimizing the entropy across all one-vs-all classifiers for each target sample. This approach enables the model to assign unlabeled target samples to either known or unknown classes. By eliminating the need for manual threshold setting, one-vs-all classifier-based methods like OVANet have become an effective solution in UniDA.

Despite its promising results, OVANet has two major short-

*, Corresponding author

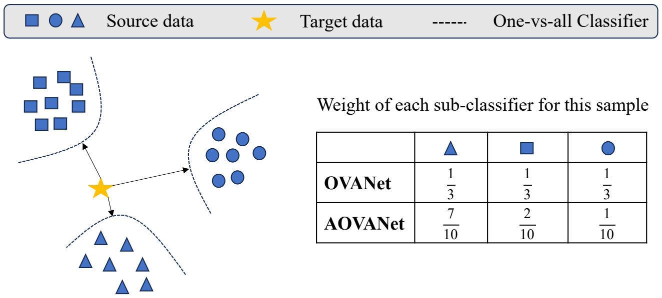


Fig. 1: The illustration of our method. A target sample contributes differently to the one-vs-all classifiers. OVANet assigns equal weight to each one-vs-all classifier, while AOVANet operates with an adaptive model weight strategy.

comings. First, it fails to account for the fact that a target sample contributes differently to each one-vs-all classifier during model adaptation. As illustrated in Figure 1, OVANet minimizes the entropy across all one-vs-all classifiers for target samples, implicitly assuming that all samples are equally important to each classifier. However, a given sample exhibits varying distances to different class boundaries. Classes farther from the sample tend to have lower entropy due to the larger distance, meaning that more weight should be assigned to the class whose boundaries are closer to the sample. In contrast, OVANet treats each class equally, which leads to suboptimal performance.

Secondly, our experiments show that OVANet's performance deteriorates significantly as the number of source classes increases. As illustrated in Figure 2, OVANet's performance drops rapidly with the increasing number of source domain categories in the Amazon-to-Webcam task from the Office-31 dataset, particularly when the number exceeds 12. This decline can be attributed to OVANet assigning equal importance to each open-set classifier. As the number of source categories grows, the attention given to each category decreases, especially for the most relevant class, which results in lower classification accuracy.

To address these issues, we propose a simple yet effective method called **A**daptive **O**ne-**V**s-**A**ll **Net**work (**AOVANet**), which enables adaptive learning based on one-vs-all classifiers for UniDA. Our approach aims to identify an index that quantifies the importance of each one-vs-all classifier and incorporates it as a weight in the open-set entropy minimization. This weighting mechanism ensures that the model focuses on refining decision boundaries for the most relevant classifiers. Specifically, we use the probability output from the closed-set classifier as the weight, as it adaptively reflects the contribution of each one-vs-all classifier with higher relevance assigned larger weights. Our method is both simple and effective, requiring no additional hyperparameters due to the adaptive weighting scheme. Moreover, experimental results demonstrate that our method experiences less performance degradation as the number of known classes increases.

The main contribution is summarized as follows:

- We found that a target sample contributes differently to one-vs-all classifiers in UniDA and propose an adaptive learning method to highlight the effect of the more relevant classifier.
- We introduce AOVANet, and design a dynamic weight allocation mechanism, supported by a closed-set classifier, which enables adaptive adjustments across different one-vs-all classifiers.
- We conduct extensive experiments in both universal and open-set domain adaptation settings on multiple image classification datasets. The results validate the effectiveness of the proposed method.

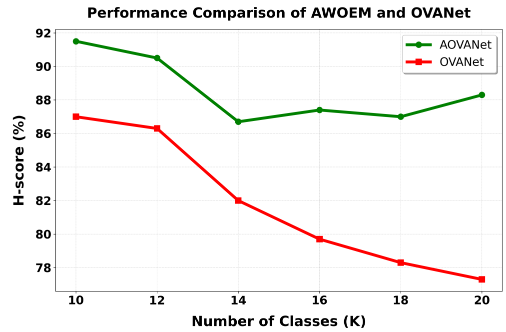


Fig. 2: The performance changes between OVANet and AOVANet with the increasing number of source classes.

II. Related Work

A. Universal Domain Adaptation

Universal Domain Adaptation (UniDA) addresses both domain-shift and category-shift between source and target domains. It aims to transfer labeled knowledge from the source domain to the target domain, particularly when the target domain lacks labels. Adversarial methods rely on a feature extractor and domain discriminator to extract domain-invariant features. UAN [9] introduced generative adversarial concepts to UniDA, using entropy and domain similarity to assess sample uncertainty and assign weights to samples. Uni3DA [14] maximizes the weight difference between high-confidence samples and a threshold. CMU [15] combines entropy with confidence and consistency measures to quantify sample uncertainty. I-UAN [16] employs the Pseudo-Margin Vector (PMV) and Target Pseudo-Margin Register (TPMR) for shared class estimation and updates during training. UMAN [17] extends I-UAN to multi-source domains by employing classifiers for all source categories. However, these methods often require manually set thresholds, which leads to erroneous rejection of unknown categories.

B. Entropy Minimization

Entropy, in a general sense, is a measure of uncertainty or randomness. Entropy minimization, therefore, involves reducing this uncertainty or randomness. Entropy minimization is often used in the fields of information theory, machine learning, and statistical mechanics. In machine learning, entropy minimization is utilized to exploit unlabeled data in semi-supervised

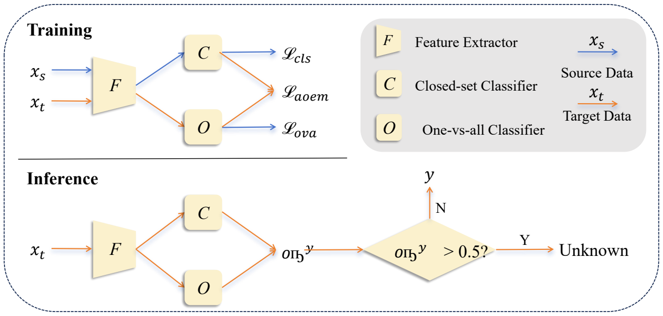


Fig. 3: Overview of AOVANet. It is composed of three components: 1) a feature extractor $F(\cdot)$, 2) a closed-set classifier $C(\cdot)$, and 3) one-vs-all classifiers $O(\cdot)$ consisting of $|C_s|$ binary classifiers $O^j(\cdot)$.

learning [18] and domain adaptation [4], to enforce the decision boundaries in the unlabeled data traverse low-density regions.

Consider a sample's output through a multi-class classifier represented as $p = \{p_1, p_2, ..., p_{|L|}\}$, where $\sum_{i=1}^{|L|} p_i = 1$ and $|L|$ denotes the number of classes. The entropy of this output, defined as $H(p) = -\sum_{i=1}^{|L|} p_i \log(p_i)$, serves as a measure of the model's prediction confidence, where lower entropy indicates higher confidence. For open-set problems, where traditional entropy minimization techniques are inadequate, OVANet [12] introduced open-set entropy minimization. This approach computes the entropy across all binary classifiers in a one-vs-all network and employs the average entropy as a loss function to optimize the model.

III. Problem Statement

Consider a labeled source dataset $\mathcal{D}_s = \{(x_s, y_s) : x_s \in \mathcal{X}, y_s \in C_s\}$, where C_s represents the source label set, $\mathcal{X}$ represents the input space, and x_s is sampled from the marginal distribution p_s. Additionally, consider an unlabeled dataset $\mathcal{D}_t = \{x_t : x_t \in \mathcal{X}\}$, with each x_t sampled from the marginal distribution p_t. Let C_t denote the target label set. In UniDA, there is no prior knowledge related to the relationship between C_s and C_t [9]. For simplicity, the shared label set is denoted as $C = C_s \cap C_t$ with $|C| = K$, and the private label sets for the source and target domains are defined as $C'_s = C_s \setminus C_t$ and $C'_t = C_t \setminus C_s$, respectively. The goal of UniDA is to correctly classify samples $x_t \in C$, while detecting samples $x_t \in C'_t$ as unknown.

IV. AOVANet

A. Approach Overview

As shown in Figure 3, our model has three components: 1) a feature extractor $F(\cdot)$, 2) a closed-set classifier $C(\cdot)$ which outputs a probability vector $p \in R^{|C_s|}$, and 3) one-vs-all classifiers $O(\cdot)$ consisting of $|C_s|$ binary classifiers $O^j(\cdot)$, $j \in \{1, ..., |C_s|\}$. First, we introduce the one-vs-all classification loss, which is used to train the one-vs-all classifiers on the source domain. Next, we describe how the models are adapted using the proposed adaptive open-set entropy minimization loss.

B. One-vs-All Classification Loss

In this subsection, we first review the one-vs-all classification (OVA) loss. The one-vs-all classifiers are able to detect unknown samples in the target domain by delineating a class boundary between inliers and outliers for each class. Specifically, each classifier is trained with its corresponding samples from this class as positive and samples from other classes as negative. For instance, the open-set classifier for class j, denoted as $O^j(\cdot)$, outputs a two-dimensional vector $z_j = \text{Softmax}(O^j(F(x))) \in \mathbb{R}^2$, where each dimension corresponds to the sample's likelihood of being an inlier or an outlier, respectively. The function $o^j(x) = z_j^0$ represents the probability of x being an inlier for class j, while $\overline{o}^j(x) = 1 - o^j(x) = z_j^1$ signifies the probability of x being an outlier.

To train one-vs-all classifiers, for each $O^j(\cdot)$, the source domain samples of the category j are positive samples, while those of other categories are negative samples. However, this approach creates an imbalance between positive and negative sample classes. Therefore, we follow OVANet and adopt the Hard Negative Classifier Sampling (HNCS), which randomly selects one sample from the corresponding class to act as a positive sample and the most challenging negative sample from other classes as the negative sample. This addresses the issue of class imbalance and saves computational resources. We use HNCS as the strategy for training the one-vs-all classifiers, with the specific loss function as follows:

$$\mathcal{L}_{ova}(x_i^s, y_i^s) = -log(o^{y_i^s}(x_i^s)) - \min_{j \neq y_i^s} log(\overline{o}^j(x_i^s)) \quad (1)$$

For a source domain sample, Equ.1 maximizes the positive output for the correct label's open-set classifier and the lowest negative output for other categories.

C. Adaptive Open-Set Entropy Minimization

As OVANet do not consider the importance of different one-vs-all classifiers. We propose the adaptive open-set entropy minimization for better adaptation.

To enhance the separation in low-density areas of the unlabeled target domain, OVANet [12] proposed Open-set Entropy Minimization, depicted in the upper portion of Figure 4. This method computes the entropy for each open-set classifier and averages the entropy across $|C_s|$ classifiers to derive the final open-set entropy minimization loss, namely,

$$\begin{aligned}\mathcal{L}_{oem}(x_i^t) = -\sum_{j=1}^{|C_s|} \times [o^j(x_i^t) log(o^j(x_i^t)) \\ + \overline{o}^j(x_i^t) log(\overline{o}^j(x_i^t))]\end{aligned} \quad (2)$$

However, OVANet fails to consider that a target sample contributes differently to each one-vs-all classifier during model adaptation. Specifically, OVANet minimizes the entropy across all one-vs-all classifiers for target samples, implicitly assuming equal importance for all classes. In reality, a given sample may exhibit varying distances to different class boundaries. Classes farther from the sample may have lower entropy due to the large distance, so more weight should be given to the class whose

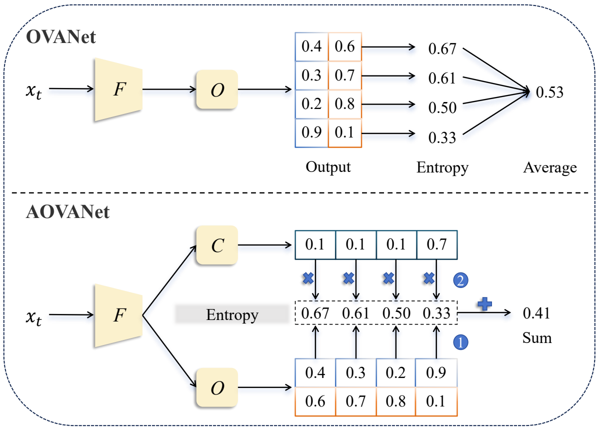


Fig. 4: The differences between OVANet and AOVANet for model adaptation on target samples.

boundaries are closer to the sample. By treating each class equally, OVANet leads to sub-optimal performance. Moreover, this issue exacerbates as the number of source domain categories increases, causing a decrease in the H-score.

To address this, we propose adjusting the model weights adaptively based on the importance of each open-set classifier for a target sample. This adaptive weighting mechanism can improve the model's ability to distinguish unknown categories and enhance overall transfer performance. Accordingly, as shown in Figure 4, we introduce the Adaptive Open-set Entropy Minimization (AOEM) loss. The computation of AOEM loss is as follows:

$$\mathcal{L}_{aoem}(x_i^t) = -\sum_{j=1}^{|C_s|} w(j) \times [o^j(x_i^t) log(o^j(x_i^t)) + \overline{o}^j(x_i^t) log(\overline{o}^j(x_i^t))] \tag{3}$$

where $w(j)$ represents the adaptive weight for each open-set classifier. To compute this weight, we follow previous self-calibrated work [19] and adopt the prediction probability of the closed-set classifier as the weight:

$$w(j) = p(\hat{y}^j | x_i^t) \tag{4}$$

Where $p(\hat{y}^j|x_i^t)$ means the j-th dimension of $p(\hat{y}|x_i^t)$. With this strategy, if a target sample is closer to an open-set classifier, the prediction probability of the closed-set classifier for that class is likely to be larger. Therefore, the closed-set classifier's prediction probability serves as a good estimator for the adaptive weight, with a larger weight indicating a more relevant open-set classifier.

D. Learning Objective

The overall optimization objective of the model can be formulated by integrating the AOEM loss, the loss of the closed-set classifier $C(\cdot)$, and the loss of one-vs-all classifiers. The overall loss is shown in the Equ.5,

$$\begin{aligned} \mathcal{L}_{all}(x, y) &= \mathbb{E}_{(x_i^s, y_i^s) \sim \mathcal{D}_s} \mathcal{L}_{src}(x_i^s, y_i^s) + \lambda \mathbb{E}_{x_i^t \sim \mathcal{D}_t} \mathcal{L}_{aoem}(x_i^t) \\ \mathcal{L}_{src}(x_i^s, y_i^s) &= \mathcal{L}_{cls}(x_i^s, y_i^s) + \mathcal{L}_{ova}(x_i^s, y_i^s) \end{aligned} \tag{5}$$

where λ is a hyper-parameter; $\mathcal{L}_{src}$ is the loss in the source domain, which includes the closed-set classifier loss L_{cls} and the one-vs-all classifier loss $\mathcal{L}_{ova}$. The loss L_{cls} is calculated using cross-entropy loss.

During the inference stage, following previous work, the closed-set classifier is first applied to predict a closed-set label $\hat{y} \in \{1, ..., |C_s|\}$. Then, if the corresponding open-set classifier $O^{\hat{y}}$ predicts an inlier, then the output class is $\hat{y}$; otherwise, the output is "unknown".

V. Experiment

A. Setup

Datasets. This paper conducts experiments on three benchmark datasets: Office-31 [20], Office-Home [21] and VisDA [22]. We follow previous work [9], [10] and divide the datasets into three parts: source domain private labels C_s', target domain private labels C_t', and shared classes between source and target domains C. This subsection adopts the same strategy for categorizing the datasets as these studies. To display the data settings in the experimental results table, we use the format $(C/C_s'/C_t')$, such as (10 / 10 / 11), to explicitly show the number of classes in these three parts. We also consider a special case of UniDA, i.e., $C_s' = 0$, which is called open-set domain adaptation (ODA) [23], [24].

Evaluation Metric. In the context of UniDA, it is essential to accurately distinguish between known and unknown categories. An effective evaluation metric should balance the classification performance of known categories with the rejection effectiveness of unknown categories. To this end, we utilize the H-score [15] to assess model performance. An increase in the H-score reflects an enhancement in overall model performance. Specifically, the H-score calculates the harmonic mean of the accuracies for known categories (acc_c) and unknown categories (acc_t), thereby providing a comprehensive measure of the model's ability to identify unknown categories:

$$\text{H-score} = \frac{2 \times acc_c \times acc_t}{acc_c + acc_t}, acc_c = \frac{1}{K}\sum_{j=1}^{K} acc_j \tag{6}$$

The acc_c represents the average accuracy across the known categories, and k_{com} is the total number of known categories. The H-score is an essential metric for evaluating performance in UniDA tasks, offering the unique advantage of increasing only when both the accuracy of known categories (acc_c) and unknown categories (acc_t) are high. This property effectively reflects the model's overall performance.

Besides, the accuracy of known category classification (Acc Close), the accuracy of unknown categories (UNK), and the distinguishability between known and unknown categories (AUROC) are also adopted.

Implementation. The experiments were conducted using PyTorch on a single NVIDIA Tesla V100 GPU. In alignment with prior research [12], [15], [25], we utilized ResNet50, pre-trained on the ImageNet [26] dataset, as our backbone network. We modified this network by removing its final fully connected layer to transform it into a feature extractor F. For the closed-set

TABLE I: UniDA H-score(%) on the Office-31 dataset (10/10/11)

Method	A2W	A2D	W2A	W2D	D2A	D2W	Avg.
UAN	58.6	59.7	60.3	71.4	60.1	70.6	63.5
CMU	67.3	68.1	72.2	80.4	71.4	79.3	73.1
I-UAN	79.4	71.5	83.0	80.7	81.0	81.5	79.5
ROS	71.3	71.4	79.2	95.3	81.0	94.6	82.1
DANCE	71.5	78.6	72.2	87.9	79.9	91.4	80.3
DCC	78.5	88.5	75.9	88.6	70.2	79.3	80.2
PCL	80.5	82.8	65.7	93.5	68.7	78.8	78.3
OVANet	77.3	84.5	81.8	95.8	76.9	94.7	85.2
OVANet*	80.9	85.4	82.5	**97.5**	82.5	92.3	86.9
SNAIL	80.6	82.4	86.4	94.2	84.2	**96.5**	87.4
CPR	81.4	84.4	**91.3**	96.8	85.5	93.4	88.8
AOVANet	**88.3**	**89.2**	88.6	95.4	**87.9**	95.3	**90.8**

TABLE II: ODA H-score(%) on the Office-31 dataset (10/0/11)

Method	A2W	A2D	W2A	W2D	D2A	D2W	Avg.
UAN	46.8	38.9	54.9	53.0	68.0	68.8	55.1
DCC	54.8	58.3	85.3	80.9	67.2	89.4	72.6
DANCE	78.8	84.9	68.3	88.9	79.1	78.8	79.8
ROS	71.7	65.8	82.0	98.2	87.2	94.8	83.3
OSBP	77.5	81.0	72.9	91.0	78.2	95.0	82.6
CPR	89.4	**90.4**	88.6	92.7	86.7	**98.5**	91.1
OVANet	87.0	85.1	88.4	**99.4**	84.9	98.4	91.1
OVANet*	90.2	89.7	86.8	82.6	**99.8**	96.9	91.0
AOVANet	**91.5**	85.1	**91.4**	95.5	91.2	96.4	**91.9**

classifier C and the one-vs-all classifier O, we implemented a single fully connected layer for each. The primary distinction between them lies in the output dimensions: $|C_s|$ for the closed-set classifier and $2 \times |C_s|$ for the one-vs-all classifier. For hyper-parameter selection, a fixed optimal hyper-parameter λ of 0.1 was consistently applied.

Baselines. To rigorously evaluate our method, we selected benchmark methods for both UniDA and ODA. These include: Open Set Back Propagation (OSBP) [5]; Techniques that integrate domain discriminators and weighting mechanisms such as UAN [9], CMU [15], I-UAN [16], and SNAIL [27]; Domain Consensus Knowledge Mining (DCC) [25]; self-supervised learning approaches like DANCE [10] and ROS [6]; OVANet [12], employing one-vs-all classifiers; and class prototype-based methods such as SPA [28], CPR [11], and PCL [29]. Significantly, when SPA is integrated with OVANet, it is referred to as OVANet*. We will not compare with UniAM [30] and the recent work by Deng et al. [31] because they utilized large-scale pre-trained models (such as CLIP [32] and ViT [33]), which gives them an unfair advantage over our work and other baselines.

B. Results on UniDA

Office-31. Table I showcases the comparison results on the Office-31 dataset in the UniDA scenario, where the highest H-scores for each task are highlighted in bold and the second highest scores are highlighted in underlined. The experimental design involved defining 10 classes as private classes for the source domain, 10 classes as shared classes across domains, and 11 classes as private classes for the target domain, with a degree of sharing between the domains denoted by $\xi = 0.32$. Our method demonstrated superior performance in three of the six tasks, achieving an average H-score of 90.8%, which is an improvement of 2.0% over the highest scores of competing methods. Particularly notable is the Amazon to Webcam task, where our method surpassed other leading scores by 6.9%, marking the most substantial improvement for a single task.

Office-Home. Table III presents the results of our method applied to UniDA experiments on the Office-Home dataset. The experimental setup included 5 classes exclusive to the source domain, 10 shared classes, and 50 exclusive classes for the target domain, resulting in a sharing degree between the source and target domains of $\xi = 0.15$. Our method achieved the highest average performance on the Office-Home dataset in this division. However, under the (10/5/50) configuration, our method surpassed the second-highest method, OVANet, by only 0.4, which may be due to the larger domain shifts.

VisDA. We conducted UniDA experiments exclusively on the VisDA dataset, a large-scale testbed for unsupervised domain adaptation across visual domains. We selected six shared categories, and three categories each that are private to the source and target domains. The experimental results are presented in Table III. Our method outperformed OVANet [12] by 7.1% and surpassed SNAIL [27], achieving the best result.

C. Results on ODA

Office31. Table II displays the comparison results on the Office-31 dataset in the ODA scenario. In this scenario, the source domain did not have private categories but included 10 shared categories, while the target domain had 11 private categories, resulting in an increased sharing degree between the source and target domains to $\xi = 0.48$. Despite the smaller gains in the (10/0/11) setup compared to the (10/10/11) setup in Table I, our method still attained the optimal H-score, outperforming OVANet and CPR [11]. Moreover, our method achieved the highest scores in two tasks and the second highest in another.

Office-Home. Table IV presents the outcomes of the ODA experiments on the Office-Home dataset. In this configuration, the source and target domains share 25 categories, with the source domain's labels being a subset of the target domain. The target domain contains 40 private categories, leading to a sharing degree between the source and target domains of $\xi = 0.38$. Our method achieved the highest average performance on the Office-Home dataset in this division. Under the (25/0/40) configuration, our method improves better than the (10/5/50) configuration in Table III, demonstrating its advantages in dealing with more source classes effectively.

D. Insightful Analysis

Ablation Study. The results of the ablation study are displayed in Table VI, examining the individual contributions of the AOEM loss and One-vs-All Classification (OAC) loss in the ODA scenario for the Webcam to Amazon task on the Office-31 dataset. This experiment systematically evaluates the impact of removing each module separately. The model

TABLE III: UniDA H-score (%) on the Office-Home (10/5/50) and VisDA(6/3/3) dataset

Method	Office-Home(10/5/50) A2C	A2P	A2R	C2A	C2P	C2R	P2A	P2C	P2R	R2A	R2C	R2P	Avg.	VisDA (6/3/3)
OSBP	39.6	45.1	46.2	45.7	45.2	46.8	45.3	40.5	45.8	45.1	41.6	46.9	44.5	27.3
UAN	51.6	51.7	54.3	61.7	57.6	61.9	50.4	47.6	61.5	62.9	52.6	65.2	56.6	30.5
SNAIL	55.9	57.9	63.1	52.5	55.4	56.4	66.8	53.5	61.1	64.3	53.8	63.2	58.6	59.8
CMU	56.0	56.9	59.2	67.0	64.3	67.8	54.7	51.1	66.4	68.2	57.9	69.7	61.6	34.6
DANCE	**61.0**	60.4	64.9	65.7	58.8	61.8	**73.1**	61.2	66.6	67.7	62.4	63.7	63.9	42.8
I-UAN	54.1	63.1	65.2	70.5	68.3	73.2	61.9	51.8	63.8	69.8	55.6	70.7	64.0	-
ROS	54.0	**77.6**	**85.3**	62.1	71.0	76.4	68.8	52.4	83.2	71.6	57.8	79.2	70.0	50.1
DCC	58.0	54.1	58.0	**74.6**	70.6	**77.5**	64.3	**73.6**	74.9	**81.0**	75.1	80.4	70.2	43.0
PCL	52.9	71.7	84.5	70.8	**72.9**	82.1	66.8	43.8	**84.2**	76.4	**84.2**	76.5	70.3	-
OVANet	60.0	**77.6**	79.2	69.7	68.4	75.2	69.5	55.3	80.1	75.7	62.2	78.7	71.0	53.1
AOVANet	54.9	76.8	78.8	73.9	71.8	75.4	69.1	54.2	83.6	78.2	58.9	**81.1**	**71.4**	**60.2**

TABLE IV: ODA H-score (%) on the Office-Home dataset (25/0/40)

Method	A2C	A2P	A2R	C2A	C2P	C2R	P2A	P2C	P2R	R2A	R2C	R2P	Avg.
DCC	56.1	**67.5**	66.7	49.6	66.5	64.0	55.8	53.0	**70.5**	61.6	57.2	**71.9**	61.7
DANCE	**61.9**	61.3	63.7	**64.2**	58.6	62.6	**67.4**	**61.0**	65.5	65.9	**61.3**	64.2	63.0
OVANet	58.4	66.3	**69.3**	60.3	65.1	67.2	58.8	52.4	68.7	67.6	58.6	66.6	63.3
AOVANet	58.3	65.3	66.1	63.8	**67.7**	**69.3**	58.9	52.5	70.2	**70.1**	58.0	68.6	**64.1**

TABLE V: Comparison between our method and the new weight method.

Task	Our Method	Weight based on one-vs-all classifier
A2D	0.89	0.85
A2W	0.88	0.78
D2A	0.88	0.64
D2W	0.95	0.96
W2A	0.89	0.79
W2D	0.95	0.97
Average	**0.91**	**0.83**

employing solely the AOEM loss yielded the lowest H-score, illustrating its dependence on the foundational OAC loss. Without the integration of OAC loss, the classifiers do not effectively assimilate knowledge from the source domain, leading to suboptimal transfer performance. Conversely, the model utilizing only OAC loss registered an H-score of 79.1%, an accuracy of 93.3%, an UNK score of 83.1%, and an AUROC of 89.4. Incorporating AOEM loss significantly enhanced the H-score to 91.4%, accuracy to 95.6%, UNK score to 90.6%, and AUROC to 96.3. These substantial improvements across all metrics underscore the critical role of AOEM loss in boosting the model's overall performance.

Besides, to show the effectiveness of weighting strategy, we design a new weight method, where the output of a one-vs-all classifier followed by softmax is adopted as a weight method. The comparison results are shown in Table V. As we can see, the proposed method achieves better results than a random weight or one-vs-all classifier-based method, which could show its effectiveness.

Feature Visualization. We employed T-SNE to visualize feature distributions under the ODA scenario for the Amazon to DSLR task on the Office-31 dataset. Figure 5a illustrates the feature space distribution for both the source and target domains. In this figure, red sample points denote source domain data, blue points indicate known class data from the target domain, and green points represent unknown class data from

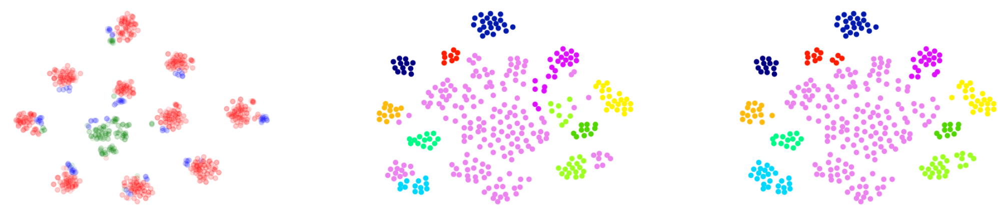
TABLE VI: Ablation study results (%) on Office-31: Webcam to Amazon task (10/0/11)

OAC	AOEM	H-score	Acc close	UNK	AUROC
✗	✓	42.1	94.2	85.6	55.0
✓	✗	79.1	93.3	83.1	89.4
✓	✓	**91.2**	**95.6**	**90.6**	**96.3**

the target domain. These distributions show that known class samples in the target domain closely align with corresponding categories in the source domain, forming clusters that facilitate classification. Additionally, unknown class samples are clearly separated, aiding the identification of these categories. Figures 5b and 5c present the distributions of true labels and predicted values in the target domain, respectively, with different colors marking various categories: pink for unknown classes and other colors for known classes. It is evident that most samples are accurately classified, although occasional misclassifications do occur.

VI. Conclusion

In this paper, we focus on UniDA. Considering that previous one-vs-all classifier-based methods, e.g., OVANet, have not considered that a sample contributes differently to the one-vs-all classifiers, we propose to adapt the model with an adaptive model weight. Specifically, we propose AOVANet, which utilizes the prediction probability of close-set classifiers as weights for the one-vs-all classifiers. This strategy could adaptively evaluate the relevance of each one-vs-all classifier, with a more relevant classifier having a larger weight. Despite its simplicity, the results on three cross-domain image classification datasets show that AOVANet significantly outperforms the baseline method, OVANet, especially when the source domain has a large number of known classes. We believe that this simple yet effective method can significantly advance the application of UniDA in real-world scenarios.



(a) Source domain and target domain. (b) True label of target Domain. (c) Predication of target Domain.

Fig. 5: T-SNE visualization for ODA: Amazon to DSLR on the Office-31 dataset.

VII. Acknowledgements

This work was supported by the CAAI-Lenovo Blue Sky Research Fund (No. 2025CAAI-LENOVO-11), the Shandong Provincial Natural Science Foundation(No. ZR2025QC1570), and the National Natural Science Foundation of China (62402293).

References

[1] H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, and M. Marchand, "Domain-adversarial neural networks," *ICML*, 2014.

[2] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," in *Proceedings of International conference on machine learning*. PMLR, 2015, pp. 1180–1189.

[3] M. Long, Y. Cao, J. Wang, and M. Jordan, "Learning transferable features with deep adaptation networks," in *Proceedings of International conference on machine learning*. PMLR, 2015, pp. 97–105.

[4] M. Long, H. Zhu, J. Wang, and M. I. Jordan, "Unsupervised domain adaptation with residual transfer networks," *Advances in neural information processing systems*, vol. 29, 2016.

[5] K. Saito, S. Yamamoto, Y. Ushiku, and T. Harada, "Open set domain adaptation by backpropagation," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 153–168.

[6] S. Bucci, M. R. Loghmani, and T. Tommasi, "On the effectiveness of image rotation for open set domain adaptation," in *Proceedings of European conference on computer vision*. Springer, 2020, pp. 422–438.

[7] J. Jang, B. Na, D. H. Shin, M. Ji, K. Song, and I.-C. Moon, "Unknown-aware domain adversarial learning for open-set domain adaptation," *Advances in Neural Information Processing Systems*, vol. 35, pp. 16 755–16 767, 2022.

[8] Z. Cao, L. Ma, M. Long, and J. Wang, "Partial adversarial domain adaptation," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 135–150.

[9] K. You, M. Long, Z. Cao, J. Wang, and M. I. Jordan, "Universal domain adaptation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 2720–2729.

[10] K. Saito, D. Kim, S. Sclaroff, and K. Saenko, "Universal domain adaptation through self supervision," *Advances in neural information processing systems*, vol. 33, pp. 16 282–16 292, 2020.

[11] S. Hur, I. Shin, K. Park, S. Woo, and I. S. Kweon, "Learning classifiers of prototypes and reciprocal points for universal domain adaptation," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2023, pp. 531–540.

[12] K. Saito and K. Saenko, "Ovanet: One-vs-all network for universal domain adaptation," in *Proceedings of the ieee/cvf international conference on computer vision*, 2021, pp. 9000–9009.

[13] Y. Lu, M. Shen, A. J. Ma, X. Xie, and J.-H. Lai, "Mlnet: Mutual learning network with neighborhood invariance for universal domain adaptation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 4, 2024, pp. 3900–3908.

[14] Y. Ren, Y. Cong, J. Dong, and G. Sun, "Uni3da: Universal 3d domain adaptation for object recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 33, no. 1, pp. 379–392, 2022.

[15] B. Fu, Z. Cao, M. Long, and J. Wang, "Learning to detect open classes for universal domain adaptation," in *Proceedings of Computer Vision–ECCV 2020: 16th European Conference*. Springer, 2020, pp. 567–583.

[16] Y. Yin, Z. Yang, X. Wu, and H. Hu, "Pseudo-margin-based universal domain adaptation," *Knowledge-Based Systems*, vol. 229, p. 107315, 2021.

[17] Y. Yin, Z. Yang, H. Hu, and X. Wu, "Universal multi-source domain adaptation for image classification," *Pattern Recognition*, vol. 121, p. 108238, 2022.

[18] Y. Grandvalet and Y. Bengio, "Semi-supervised learning by entropy minimization," *Advances in neural information processing systems*, vol. 17, 2004.

[19] G. Yu, J. Zhu, J. Yao, and B. Han, "Self-calibrated tuning of vision-language models for out-of-distribution detection," *NeurIPS*, 2024.

[20] K. Saenko, B. Kulis, M. Fritz, and T. Darrell, "Adapting visual category models to new domains," in *ECCV*. Springer, 2010, pp. 213–226.

[21] H. Venkateswara, J. Eusebio, S. Chakraborty, and S. Panchanathan, "Deep hashing network for unsupervised domain adaptation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 5018–5027.

[22] 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.

[23] P. Panareda Busto and J. Gall, "Open set domain adaptation," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 754–763.

[24] Q. Wang, F. Meng, and T. P. Breckon, "Progressively select and reject pseudo-labelled samples for open-set domain adaptation," *IEEE Transactions on Artificial Intelligence*, 2024.

[25] G. Li, G. Kang, Y. Zhu, Y. Wei, and Y. Yang, "Domain consensus clustering for universal domain adaptation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 9757–9766.

[26] J. Deng, W. Dong, R. Socher, L. J. Li, and F. F. Li, "Imagenet: a large-scale hierarchical image database," in *Proceedings of 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[27] Z. Han, W. Su, R. He, and Y. Yin, "Snail: Semi-separated uncertainty adversarial learning for universal domain adaptation," in *Proceedings of Asian Conference on Machine Learning*. PMLR, 2023, pp. 436–451.

[28] J. N. Kundu, S. Bhambri, A. R. Kulkarni, H. Sarkar, and V. Jampani, "Subsidiary prototype alignment for universal domain adaptation," *Advances in Neural Information Processing Systems*, vol. 35, pp. 29 649–29 662, 2022.

[29] X. Shan, T. Ma, and Y. Wen, "Prediction of common labels for universal domain adaptation," *Neural Networks*, 2023.

[30] D. Zhu, Y. Li, J. Yuan, Z. Li, K. Kuang, and C. Wu, "Universal domain adaptation via compressive attention matching," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2023, pp. 6974–6985.

[31] B. Deng and K. Jia, "Universal domain adaptation from foundation models: A baseline study," 2023.

[32] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, and et al., "Learning transferable visual models from natural language supervision," in *ICML*, vol. 139. PMLR, 18–24 Jul 2021, pp. 8748–8763.

[33] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," 2021.