# Toward Improving Robustness and Accuracy in Unsupervised Domain Adaptation

**Aishwarya Soni and Tanima Dutta**

Department of Computer Science and Engineering, Indian Institute of Technology (BHU) Varanas, India
{aishwaryasoni.rs.cse19, tanima.cse}@iitbhu.ac.in

## Abstract

Adversarial robustness in the context of Unsupervised Domain Adaptation (UDA) is particularly challenging due to the lack of labels in the target domain. Pseudo labels are often used to make adversarial robust models but compromise robustness and accuracy, falling short of the performance due to noise and inaccuracies in these pseudo labels. The main challenges in achieving robustness and accuracy include ensuring reliable pseudo labels and developing effective training methods that bring alignment between clean and adversarial examples of target data. To address these challenges, we propose a novel training method within the self-training paradigm *Consistent Attention Mapping with Self Pseudo Label Refinement (CAM+SPLR)*. It begins with the pre-training of the UDA model, resulting in a UDA pre-trained model, which is initialized into two separate models: the Anchor model and the TargetNet model. The Anchor model encourages the attention maps of clean images and their adversarial counterparts to be similar, while the TargetNet model simultaneously performs self-training using Adversarial target data and refining the pseudo labels. CAM+SPLR improves both semantically relevant key features and pseudo-labels through a two-step stochastic gradient descent process during training. We conducted extensive experiments on benchmark datasets, including OfficeHome, PACS, and VisDA, demonstrating significant improvements in both robustness and accuracy. Our method achieves an average accuracy improvement of 6% and 8.1% and an average robustness improvement of 10.2% and 4.9%, compared to state-of-the-art methods on the PACS and VisDA datasets.

## Introduction

Unsupervised Domain Adaptation (UDA) (Ganin and Lempitsky 2015; Wilson and Cook 2020) has achieved significant success in transferring learned representations from a labelled source domain to an unlabeled target domain. Existing methods (Ganin et al. 2016; Long et al. 2018, 2017; Saito et al. 2018; Sun and Saenko 2016; Xu et al. 2019; Choi et al. 2019; Yang et al. 2021b) focus on improving the model's ability to generalize and accurately map these representations to the target domain. However, in real-world applications, it's crucial for UDA models to be secure against potential threats like adversarial examples (AEs). UDA are particularly vulnerable to adversarial attacks, where even small perturbations can significantly degrade their performance. Despite this vulnerability, the robustness of UDA methods against adversarial attacks has been largely overlooked, as highlighted in recent studies (Lo and Patel 2022; Zhu et al. 2023). This gap presents a major challenge for deploying UDA models in safety-critical applications.

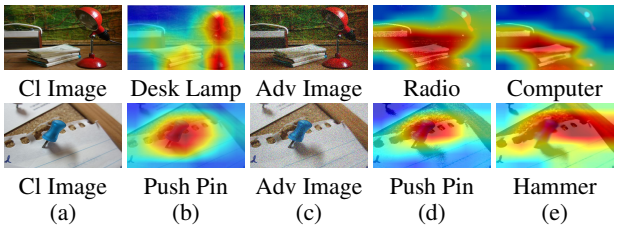


Figure 1: (a) Clean image, (b) Corresponding attention map where red regions indicate areas contributing more to the model's classification, (c) Adversarial example of (a), (d) Attention map of adversarial example illustrating the attention shift relative to the attention map of the corresponding clean image, as depicted in (b), (e) Attention shift due to noisy labels, focusing on the background or semantically irrelevant features. (Cl: Clean, Adv: Adversarial)

Adversarial Training (AT) is a proven strategy in deep neural networks (DNNs) for defending against adversarial attacks. Several AT approaches have been developed (Szegedy et al. 2013; Goodfellow, Shlens, and Szegedy 2014; Madry et al. 2017; Dong et al. 2018; Lo and Patel 2021) to enhance the robustness of DNNs by training models on adversarial examples using a max-min optimization process. However, this approach typically relies on ground-truth labels to generate adversarial examples, which limits its application in UDA where labeled target data are unavailable. Moreover, the self-training method is effective in using adversarial examples from the target data to enhance robustness. These adversarial examples are generated using pseudo labels predicted by a pre-trained UDA model. However, there are still significant challenges in self-training to direct the use of pseudo labels.

When pseudo labels are used to generate adversarial examples, the noise and inaccuracies in these labels can cause the model to overfit specifically to these adversarial exam-

ples in target domain. This overfitting results in suboptimal performance on unseen adversarial examples. Moreover, in self-training, model may start focusing on misleading irrelevant features introduced by adversarial perturbations, driven by noisy or inaccurate pseudo labels, rather than concentrating on meaningful, semantically relevant features.

In Figure 1, we visualized the attention map of a clean image and its corresponding adversarial image using a ResNet-50 backbone pre-trained on ImageNet and fine-tuned on source-target data for domain adaption task. The clean image is originally labeled as "Desk Lamp," and the model correctly focuses on key features such as the lamp's shape and light source. However, after adversarial perturbations are added, the model's attention may shift to irrelevant features due to the erratic patterns introduced by the perturbation. This shift occurs in the case of "Desk Lamp" but not in the case of "Push Pin." As a result, the model incorrectly classifies the image as "Radio" instead of "Desk Lamp" due to the shift, while it correctly classifies the image as "Push Pin" in the case of "Push Pin" as shown in column (d). When adversarial examples are generated and trained with incorrect or noisy pseudo labels, model may focus on irrelevant features, as seen in column (e). This results in incorrect classifications, such as "Computer" for "Desk Lamp" and "Hammer" for "Push Pin." This shift in focus impairs model's performance on adversarial and clean examples by prioritizing peripheral regions or semantically irrelevant features.

In this paper, we propose a method called Consistent Attention Mapping with Self Pseudo Label Refinement (CAM+SPLR), which redesigns the self-training method in UDA. Our method aims to enhance both adversarial robustness and clean accuracy in UDA models by focusing on two key aspects: 1) Enhance robust feature representation using *Consistent Attention Mapping (CAM)* by emphasizing semantically informative features. 2) Improving the pseudo-label quality of the target data using Self Pseudo Label Refinement (SPLR) by incorporating two Stochastic Gradient Descent (SGD) step updates using both labelled source data and unlabelled target data. We begin by training a UDA model using both source and target data, resulting in a UDA pre-trained model. This model is then initialized into two separate models: the Anchor model and the TargetNet model. The Anchor Model, which is frozen and processes clean target images, generates attention maps that help to reconstruct robust feature representations. This is achieved by encouraging the attention maps of clean examples and their adversarial counterparts, processed by the TargetNet model, to be similar by activating semantically relevant key areas. Meanwhile, the TargetNet model generates pseudo labels and performs self-training using adversarial examples of the target data, along with self-refinement of the pseudo labels. The simultaneous updation of pseudo labels, along with the minimization of attention loss during self-training, is achieved through a two-step-stochastic Gradient Descent (SGD) process in each epoch. In the first step, the TargetNet model's parameters are updated using conventional SGD, with a combined loss consisting of cross-entropy loss and attention loss. In the second step, the TargetNet model's updated parameters are again updated using conventional SGD, with a combined loss consisting of semi-supervised loss and unsupervised data augmentation loss.

Our contributions are as follows:

• We first introduce the novel self-training method Consistent Attention Mapping with Self Pseudo Label Refinement(CAM+SPLR), which simultaneously improves the robustness and accuracy of the UDA model.

• We propose the Consistent Attention Mapping (CAM) method for the self-training pipeline (CAM+SPLR). This effectively prevents the model from focusing on less informative regions influenced by adversarial perturbations and noisy pseudo-labels. It activates semantically relevant key areas with attention mapping, which enhances learning more discriminative features.

• We further proposed the Self Pseudo Label Refinement (SPLR) method that prevent the model overfitting due to inevitable noisy pseudo labels. This method employs a two-step gradient update process during training to progressively refine the pseudo labels without relying on any additional model.

• We achieve improvement in robustness and gain in standard accuracy across multiple datasets (OfficeHome (Wang et al. 2021), PACS (Li et al. 2017), VisDA (Peng et al. 2017)) compared to state-of-the-art methods (DART (Wang et al. 2024), SRoUDA (Zhu et al. 2023), and ARTUDA (Yang et al. 2021a)). Specifically, we observed remarkable average robustness gain at $\epsilon = 2/255$ of 5.2%, 4.9%, and 10.2% on the OfficeHome, VisDA, and PACS datasets, respectively . Additionally, average accuracy improved by 0.9%, 8.1%, and 6% over the UDA baseline Domain-Adversarial Neural Network (DANN) (Ganin et al. 2016) on the OfficeHome, VisDA, and PACS datasets, respectively.

The remainder of the paper first introduces the proposed methodology, covering its concepts and architecture, and then presents our experimental setup and results. Finally, we concludes the paper. Preliminaries are given in the Supplementary.

## Methodology

This section outlines our proposed methodology, detailing the training pipeline of the TargetNet model with the help of the frozen Anchor model. Figure 2 illustrates the CAM+SPLR training process for the TargetNet model using adversarial examples derived from pseudo-labeled target data. Key semantic features are progressively enhanced throughout the training, while the pseudo labels are self-refined based on the model's performance on both the source and target data.

### Pre-training of DANN

We employ a pre-trained UDA model to generate accurate pseudo-labels for training the TargetNet model. To achieve this, we utilize the DANN method (Ganin et al. 2016), which optimizes two classifiers simultaneously: (1) a label classifier and (2) a domain discriminator. The label classifier predicts the labels and is used during both training and testing, while the domain discriminator differentiates between the source and target domains during training. This joint optimization promotes the learning of domain-invariant features.

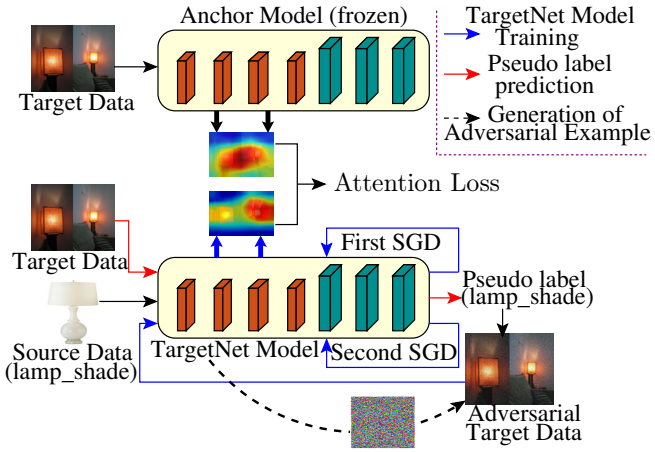


Figure 2: A schematic representation of CAM+SPLR training, where the TargetNet model is trained on adversarial examples generated from pseudo-labeled target data to enhance the key semantic features and self-refining pseudo labels. SGD: Stochastic Gradient descent.

It minimizes cross-entropy loss, $\mathcal{L}_{CE}(\cdot,\cdot)$ and domain discriminator loss, $\mathcal{L}_{dd}(\cdot,\cdot)$ as;

$$\min\left(\mathcal{L}_{CE}(F_s(x_s), y_s) + \omega \mathcal{L}_{dd}(x_s, x_t)\right), \tag{1}$$

where $\omega$ is a hyperparameter, often referred to as a regularization parameter, which controls the trade-off between the classification loss and the discriminator or domain loss. We set $\omega = 0.1$ throughout the entire training process and use the DANN method for UDA pre-training. However, our proposed method can be easily integrated with any of the existing UDA baselines.

## Consistent Attention Mapping (CAM)

In this section, our goal is to help the TargetNet model learn robust feature representations by focusing on semantically relevant regions of the input. We achieve this by enforcing consistency between the Attention Maps of clean target examples and their adversarial counterparts. Specifically, we align the attention maps to maintain focus on key areas of the input, ensuring that adversarial perturbations do not divert attention away from critical regions. Let $S$ denote the set of indices corresponding to the activation layer pairs that are prioritized in this attention alignment. Then, we can define the attention loss ($L_{\text{ATT}}$) as:

$$\mathcal{L}_{ATT} = \sum_{z \in S} \left\| \frac{P_z^{\text{ADV}}}{\|P_z^{\text{ADV}}\|_2} - \frac{P_z^{\text{CLEAN}}}{\|P_z^{\text{CLEAN}}\|_2} \right\|_2, \tag{2}$$

where CLEAN and ADV denote clean examples and their adversarial examples, respectively. $P_z^{\text{CLEAN}} = \text{vec}\left(F\left(B_z^{\text{CLEAN}}\right)\right)$ and $P_z^{\text{ADV}} = \text{vec}\left(F\left(B_z^{\text{ADV}}\right)\right)$ are the $z$-th pair of clean examples and their adversarial examples attention maps in vectorized form. This alignment encourages the model to maintain a consistent focus on semantically important regions, even under adversarial perturbation driven by inaccurate labels.

## Consistent Attention Mapping with Self Pseudo Label Refinement

Figure 2 illustrates the method of CAM+SPLR of the TargetNet model in the presence of the Anchor model. The TargetNet model possesses the UDA feature extractor and classifier architecture, and the re-designed self-training is performed using adversarial examples of target data. During self-training, the model concurrently enhances the feature representation using attention mappings and refines the pseudo labels. To achieve this, we first pass a mini-batch of unlabeled clean target data $x_t$ to the TargetNet model to get pseudo-label predictions $y_t$. The model utilizes pseudo-labels with confidence scores ($y_t^i \geq T$) to guide the training process, where $T$ represents the predefined confidence threshold. Subsequently, adversarial examples $\hat{x}_t$ are generated from these pseudo-labeled target data, and the TargetNet model is trained using these adversarial examples. During training, we perform two gradient descent steps in each epoch. In the first step, we calculate the attention loss $\mathcal{L}_{ATT}$ as defined in Eq. 2, along with the cross-entropy loss $\mathcal{L}_{CE}$. To calculate these losses, a batch of clean target examples is passed through the Anchor model, while the corresponding adversarial examples are passed through the TargetNet model. We then extract attention maps from the different layers in both models. We aim to leverage the attention supervision from the Anchor model to align the TargetNet's features with semantically relevant regions by minimizing the attention loss between clean target examples and their adversarial counterparts. This attention loss strengthens the model's ability to learn robust feature representations. In addition, during supervision, the weights of the Anchor model remain fixed and have not been updated. Furthermore, to facilitate accurate classification of the TargetNet Model, we calculate the Cross-Entropy Loss between the soft label prediction ($\hat{p}_t$) from the TargetNet Model and the hard pseudo labels ($y_t$):

$$\mathcal{L}_{CE} = CE(\hat{p}_t, y_t), \tag{3}$$

where $\hat{x}_t$ represents the adversarial example of the target data, $y_t = \arg\max(G_t(x_t))$ is the hard pseudo-label for the clean target data $x_t$ and $\hat{p}_t = G_t(\hat{x}_t)$ is the soft label prediction for the target adversarial examples outputs from the TargetNet model. Initially, the pseudo-labels may be noisy, so the TargetNet model is progressively updated to refine them. The combined loss $\mathcal{L}_1(\theta_M)$ is defined as the sum of the attention loss $\mathcal{L}_{\text{ATT}}$ and the cross-entropy loss $\mathcal{L}_{CE}$:

$$\mathcal{L}_1(\theta_M) = \mathcal{L}_{CE} + \lambda_1 \cdot \mathcal{L}_{ATT}, \tag{4}$$

where $\lambda_1$ is a hyperparameter used to balance the two terms in loss. With this calculated loss $\mathcal{L}_1(\theta_M)$, we update the **first gradient** step such as:

$$\textbf{First SGD: } \theta_M{}' = \theta_M - \eta_1 \cdot \nabla \mathcal{L}_1(\theta_M). \tag{5}$$

In the second gradient step, we generate new predictions on labelled source data and unlabelled target data using the updated model parameters. The parameters are then updated again by optimizing the second objective function, which includes Unsupervised Data Augmentation Loss (UDA) and Semi-Supervised Loss (SSL) terms:

$$\mathcal{L}_2(\theta'_M) = \mathcal{L}_{\text{UDA}} + \lambda_2 \mathcal{L}_{\text{SSL}}. \tag{6}$$

Here, $\mathcal{L}_{\text{UDA}}$ denotes the Unsupervised Adversarial Consistency Loss, and $\mathcal{L}_{\text{SSL}}$ represents the Semi-Supervised Loss for refining pseudo-labels. The hyperparameter $\lambda_2$ balances these two losses. In the second gradient step, $\mathcal{L}_{\text{SSL}}$ refines the pseudo-labels for target data, while $\mathcal{L}_{\text{UDA}}$ enforces consistency between predictions on clean and adversarial target examples, promoting the learning of robust, generalizable features. This consistency constraint improves performance by ensuring stable predictions across data variations. The $\mathcal{L}_{\text{UDA}}$ loss is computed as follows:

$$\mathcal{L}_{\text{UDA}} = CE(G_t(x_s), y_s) + \beta_k \mathbb{E}[-p_t \log(\hat{p}_t)], \tag{7}$$
$$\beta_k = \beta_0 \cdot \min(1, \frac{k+1}{a}).$$

The first term is the cross-entropy loss calculated on the labeled source data and second term the consistency regularization term, enforcing that the model's predictions on clean target examples $p_t$ remain aligned with their adversarial counterparts $\hat{p}_t$. $\beta_k$ serves as a warm-up coefficient that progressively increases its value until a constant $a$ is reached during training. This warm-up process ensures that pseudo-labels do not overly influence the model in the early stages of training. Further, the semi-supervised loss is a critical component of the second gradient update, specifically designed to refine pseudo-labels by leveraging feedback from the model's performance. This $\mathcal{L}_{\text{SSL}}$ loss is used to evaluate the model's performance after the first gradient update and is computed as:

$$\mathcal{L}_{\text{SSL}} = \Delta CE \cdot CE(\hat{p_t}, \hat{y_t}), \tag{8}$$

where $\Delta CE$ represents the difference in cross-entropy loss calculated on the labeled source data before and after the first gradient update. $CE(\hat{p_t}, y_t)$ is the cross entropy loss on the pseudo-labels, where $\hat{p_t}$ are the soft predictions and $y_t$ are the hard pseudo-labels for the clean target examples. In addition, to reduce the variance in $SSL$, we subtract the previous moving average of $\Delta CE$ from the current $\Delta CE$ when calculating the SSL term. This approach helps stabilize the learning process, ensuring consistent and reliable feedback during pseudo-label refinement. Then we do a second gradient update by using this combined loss $\mathcal{L}_2(\theta_M)$, which will act as a feedback signal:

$$\textbf{Second SGD:} \quad {\theta_M}'' = \theta'_M - \eta_2 \cdot \nabla \mathcal{L}_2(\theta'_M), \tag{9}$$

where $\eta_2$ is the learning rate for the second gradient step. The algorithm of our CAM+SPLR is summarized in the supplementary of Algorithm 1.

## Experiments

**Dataset.** We evaluate our method on the three multi-domain datasets: **OfficeHome** (Wang et al. 2021) which has four domains across 65 categories *i.e.,* Art (Ar, 2427 images), ClipArt (Cl, 4365 images), Product(Pr, 4439 images) and RealWorld (Re, 4357 images), **PACS** (Li et al. 2017) has four domains with seven categories, namely Photo (Ph, 1670 images), Art Painting (Ar, 2048 images), Cartoon (Ca, 2344 images) and Sketch (Sk, 3929 images), and **VisDA** (Peng et al. 2017) is a large dataset having two domains with 12 categories namely, Synthetic images (Syn, 152409) and Real images (Re, 55400).

| S → T | Syn→Re | | Cl→Re | | Ph→Ar | |
|---|---|---|---|---|---|---|
| **Method** | Clean | PGD | Clean | PGD | Clean | PGD |
| DANN | 67.5 | 0.3 | **68.0** | 0.0 | **89.0** | 0.0 |
| CAM+SPLR (L-AT) | 64.9 | 52.8 | 59.2 | 53.8 | 74.5 | 67.5 |
| CAM+SPLR (M-AT) | 66.1 | 54.7 | 61.1 | 54.5 | 79.2 | 69.9 |
| CAM+SPLR (H-AT) | 67.4 | 56.2 | 63.3 | 56.4 | 81.5 | 72.6 |
| CAM+SPLR (A-AT) | **69.5** | **57.8** | 65.8 | **57.5** | 83.6 | **74.5** |

Table 1: Experimental results on attention maps used with CAM+SPLR and UDA baseline DANN (Ganin et al. 2016). (L: Low Level, M: Mid Level, H: High Level, A: All Levels, AT: Attention Map.)

**Comparison.** Here, *1.) UDA baseline:* a simple UDA model trained without considering robustness, *2.) UDA+AT:* initially trains a DANN (Ganin et al. 2016) to predict pseudo-labels for the unlabeled target data, followed by applying standard adversarial loss for training the target model on the pseudo-labels, *3.) UDA+Trades:* it is similar to the UDA+AT, but the standard adversarial loss is replaced with Trades loss (Zhang et al. 2019), and *4.) UDA+Mart:* employ Mart loss instead of standard adversarial loss (Wang et al. 2019). *5.) Comparison with state-of-the-art:* ARTUDA (Yang et al. 2021a), *6.)* SRoUDA (Zhu et al. 2023), and *7.)* DART (Wang et al. 2024).

**Implementation details.** We use ResNet-50 (He et al. 2016) (pre-trained using ImageNet) as the backbone feature extractor for all the datasets in the experiments. During pre-training of DANN, we use Transfer-Learning-library (TLL) (Junguang Jiang 2020) to set up the experimental environment for UDA and follow the training hyperparameters as in (Wang et al. 2024). Next, we incorporate data augmentation techniques, like resizing to $224 \times 224$ pixels, random horizontal flips, and rotations, for target training data. The training batch size is set to 32 with Adam optimizer and a learning rate of $0.001$. We train the CAM+SPLR to 25K iteration using two stochastic gradient descent steps in every training epoch. We consider adversarial perturbation under $L_\infty$ norm, with adversarial examples generated using 10 steps of PGD with $\epsilon = 2/255$ and $\epsilon = 8/255$. Finally, we evaluate all the methods on the target data using standard accuracy and robustness computed on 20-step PGD (PGD 20) attack with $\epsilon = 2/255$ on all the datasets except $8/255$ on the VisDA dataset and PACs dataset. Results presented in the tables/graphs are expressed as % accuracy or robustness. Related work and additional results are provided in the supplementary materials and the Experiment section, including Table 1, which presents performance comparisons under black-box robustness settings; Figures 1 and 2 illustrate hyperparameter analyses; and Figure 4, highlights robustness evaluations in the feature space.

**CAM+SPLR with attention maps at different feature levels.** We use ResNet-50 as the backbone and leverage different stages of convolution blocks within the activation layers to generate attention maps. The levels from low to high correspond to the attention maps generated by

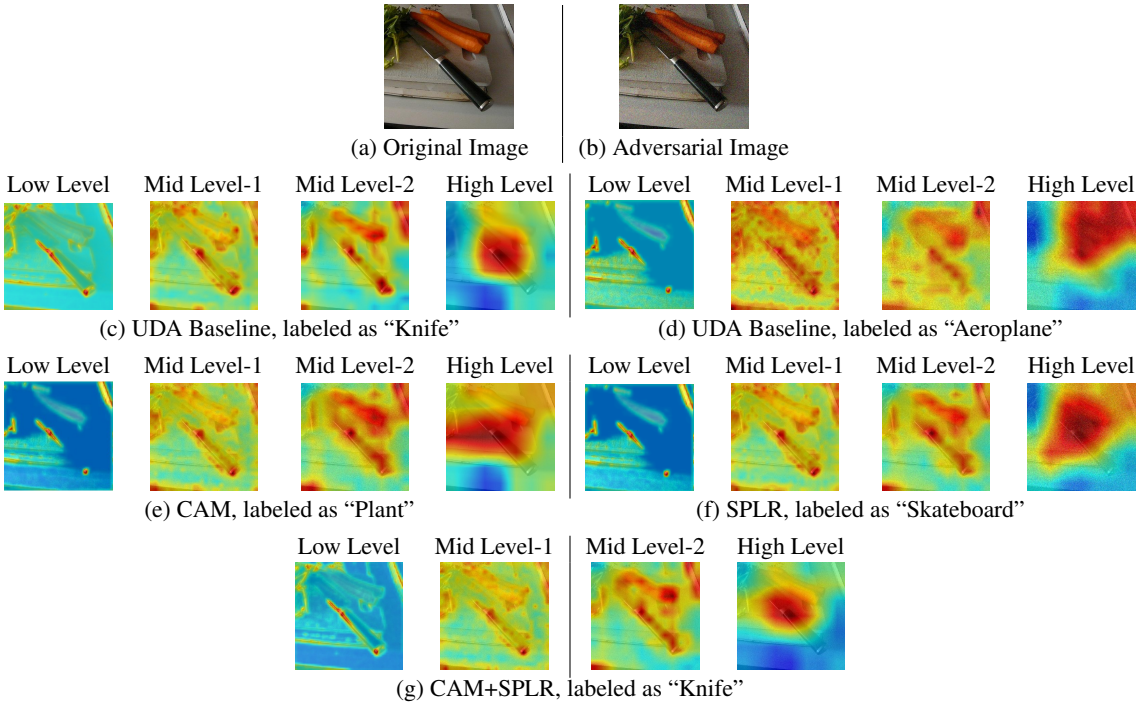


Figure 3: (a) Clean image and (b) corresponding adversarial example. (c) and (d) depict attention maps of clean and adversarial examples at different feature levels along with their predicted labels by the UDA baseline. (e) and (f) show attention maps generated by CAM and SPLR methods, respectively, for adversarial examples at different feature levels and their predicted labels. (g) presents attention maps produced by the CAM+SPLR method and their corresponding predicted labels.

four groups of convolutional structures (Zagoruyko and Komodakis 2017) in ResNet-50, namely conv2_x, conv3_x, conv4_x, and conv5_x. These groups extract features at varying levels: low-level features, mid-level features, and high-level features. In Figure 3, we presented attention maps that illustrate the effectiveness of our defence method against PGD20 attacks. ResNet-50, initially pre-trained on the ImageNet dataset and later fine-tuned on the VisDA dataset for the syn-to-real domain adaptation task, referred to as the UDA baseline model. Our analysis reveals that combining CAM and SPLR significantly improves the ability to accurately capture the outline and texture of a knife.

If only CAM method is used, it labels the "Knife" as "Plant" due to noise and inaccuracies in the pseudo label. If only SPLR is used alone, it labels the "Knife" as "Skateboard" due to the attention shift of the TargetNet model as it focuses on irrelevant features, which leads to incorrect prediction. However, the combined CAM+SPLR approach consistently generates precise predictions by activating relevant features across the entire knife. Moreover, we conducted an ablation study to identify which feature level of attention map contributes most to enhancing robustness and accuracy. This study examines the behaviour of the CAM+SPLR approach when applying attention maps at different feature levels: low, mid, and high. The evaluation was performed under a PGD20 attack with $\epsilon = 8/255$ across three source-target domain pairs: Synthetic→Real, Clipart→ RealWorld, and Photo→Art, using the VisDA, OfficeHome, and PACS datasets. If only CAM is used alone, it labels the "Knife" as "Plant" due to noise and inaccuracies in the pseudo label. If only SPLR is used alone, it labels the "Knife" as "Skateboard" due to the attention shift of the TargetNet model as it focuses on irrelevant features, leading to incorrect predictions. However, the combined CAM+SPLR approach consistently generates precise predictions by activating relevant features across the entire knife. Moreover, we conducted an ablation study to identify which feature level of the attention map contributes most to enhancing robustness and accuracy. This study examines the behavior of the CAM+SPLR approach when applying attention maps at different feature levels: low, mid, and high. The evaluation was performed under a PGD20 attack with $\epsilon = 8/255$ across three source-target domain pairs: Synthetic → Real, Clipart → RealWorld, and Photo → Art, using the VisDA, OfficeHome, and PACS datasets. As shown in Table 1, our proposed method, which incorporates attention mapping at all feature levels, achieved approximately $2-3\%$ improvement in accuracy and robustness compared to using only high-level attention mapping. Additionally, it outperformed low-level and mid-level attention mapping by $3-5\%$ and $7-8\%$, respectively, in terms of accuracy and robustness.

**Comparison on PGD** $20$ **attack at** $\epsilon = 2/255$**.** We present a comprehensive comparison of various methods in Table 2 for the PACS dataset, Table 3 for the OfficeHome dataset and Table 4 for the VisDA dataset are provided in the experiment section of the supplementary. In the experiment, we

| S → T | Ph → Ar | | Ph → Ca | | Ph → Sk | | Ca → Ar | | Ca → Ph | | Ca → Re | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Method** | Clean | PGD | Clean | PGD | Clean | PGD | Clean | PGD | Clean | PGD | Clean | PGD |
| DANN | **89.0** | 5.5 | 80.5 | 11.5 | 74.2 | 24.0 | **84.8** | 0.5 | 92.3 | 1.0 | 78.0 | 25.8 |
| UDA+AT | 82.1 | 59.0 | 85.2 | 77.1 | 78.0 | 75.3 | 76.2 | 55.0 | 93.1 | 80.2 | 80.1 | 77.2 |
| UDA + Trades | 82.0 | 63.0 | 84.2 | 76.5 | 78.6 | 75.2 | 78.5 | 58.0 | 92.2 | 82.1 | 79.9 | 77.6 |
| UDA + Mart | 80.5 | 62.7 | 81.2 | 77.6 | 77.2 | 75.8 | 76.2 | 57.8 | 91.7 | 82.6 | 79.1 | 78.2 |
| ARTUDA | 85.9 | 60.1 | 87.5 | 78.1 | 74.9 | 70.4 | 76.5 | 53.3 | 89.4 | 75.0 | 80.3 | 74.9 |
| SRoUDA | 76.1 | 56.4 | 82.4 | 71.7 | 71.9 | 63.7 | 72.0 | 50.9 | 90.3 | 79.9 | 76.7 | 72.3 |
| DART | 85.2 | 58.0 | 89.4 | 80.5 | 82.5 | 79.9 | 77.4 | 54.6 | 94.2 | 79.8 | 84.9 | 81.0 |
| **Ours** | 87.6 | **63.4** | **92.5** | **82.6** | **83.2** | **81.9** | 81.8 | **61.5** | **94.9** | **83.5** | **85.3** | **82.6** |

| S → T | Ar → Ca | | Ar → Ph | | Ar → Sk | | Sk → Ar | | Sk → Ca | | Sk → Ph | | Avg Accuracy | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Method** | Clean | PGD | Clean | PGD | Clean | PGD | Clean | PGD | Clean | PGD | Clean | PGD | Clean | PGD |
| DANN | 84.2 | 12.5 | 97.9 | 2.7 | 84.9 | 0.5 | 68.0 | 45 | 72.1 | 14.2 | 71.3 | 0.1 | 81.4 | 11.9 |
| UDA+AT | 85.0 | 76.3 | 95.0 | 83.1 | 84.8 | 81.1 | 62.1 | 37.5 | 72.5 | 64.0 | 88.8 | 73.6 | 81.9 | 70.0 |
| UDA+Trades | 85.1 | 77.0 | 98.3 | 82.2 | 86.0 | 83.3 | 69.0 | 44.2 | 76.6 | 63.8 | 89.6 | 76.4 | 83.3 | 71.6 |
| UDA+Mart | 84.9 | 78.1 | 95.5 | 82.5 | 85.2 | 83.7 | 67.5 | 45.6 | 71.8 | 63.1 | 88.3 | 77.1 | 81.6 | 72.1 |
| ARTUDA | 88.1 | 76.0 | 95.0 | 78.5 | 80.3 | 61.5 | 49.5 | 31.7 | 38.1 | 28.5 | 48.9 | 40.4 | 74.5 | 60.7 |
| SRoUDA | 84.2 | 75.8 | 94.1 | 81.5 | 77.3 | 73.2 | 24.8 | 22.4 | 72.4 | 62.3 | **91.9** | 73.1 | 76.2 | 65.3 |
| DART | 89.1 | 79.1 | **98.9** | 81.4 | 89.5 | 86.4 | 71.9 | 53.1 | 78.4 | 69.2 | 87.8 | 76.8 | 85.7 | 73.3 |
| **Ours** | **90.5** | **83.1** | 97.5 | **85.0** | **90.8** | **86.7** | **73.5** | **56.9** | **81.6** | **73.8** | 89.4 | **79.2** | **87.4** | **76.7** |

Table 2: Comparison on PGD 20 attack at $\epsilon = 2/255$ using PACS dataset. Avg accuracy calculated using both (top and bottom).

used DANN as the UDA baseline and considered the white-box attack using PGD 20 with perturbation size $\epsilon = 2/255$ to assess model robustness and derive various observations. From the results, we observe that the proposed CAM+SPLR method significantly enhances both adversarial robustness and accuracy under the UDA scenario and outperforms other UDA baselines and state-of-the-art methods. Specifically, CAM+SPLR achieves substantial improvements in model robustness and accuracy across all datasets as shown in Figure 4. Some interesting observations include enhancement in average robustness ranging from $1.4\%$ to $45.9\%$ for the OfficeHome dataset, $11.9\%$ to $76.7\%$ for the PACS dataset, and $0.4\%$ to $74.0\%$ for the VisDA dataset. Further, our method achieves better average accuracy, $1\%$ to $7\%$, as compared to the UDA baseline DANN method for all the datasets.

**Comparison on PGD** 20 **attack at** $\epsilon = 8/255$**.** We present a comparison of different methods on the VisDA dataset in Table 3. We used DANN as the UDA baseline and considered the white-box attack using PGD 20 with $\epsilon = 8/255$ to assess model robustness and accuracy. Our method achieves average robustness $0.4\%$ to $62.1\%$ on the VisDA Dataset. In comparison with the SRoUDA (Zhu et al. 2023) method, our method improved the average accuracy and robustness of $23.3\%$ and $24.8\%$, respectively. Our proposed CAM+SPLR method significantly improves both the adversarial robustness and accuracy in comparison to the other state-of-the-art methods, as shown in Table 3. Additionally, we conduct experiments on the Real→Synthetic source-target pair of the VISDA dataset under various adversarial attacks (FGSM, PGD10, PGD20, CW$\infty$), as shown in Table 4.

**Component ablation of CAM+SPLR.** We evaluate the effectiveness of the individual components in our CAM+SPLR method by conducting experiments on the PACS dataset, specifically on the Photo→Clipart and Photo→Sketch source target pairs. The results, presented in Table 6, compare the following scenarios: 1) Perform standard adversarial training (UDA+AT) on the TargetNet model, 2) Training the TargetNet model with CAM excluding SPLR method, 3) Training the TargetNet model with SPLR excluding CAM method, and 4)Training with our proposed CAM+SPLR method. We perform the experiment on perturbation size of $\epsilon = 8/255$. We observed that during standard adversarial training (UDA+AT), the TargetNet model's performance in terms of robustness and accuracy is approximately 10 to 14% lower compared to our pro-

| S → T | Syn → Re | | Re → Syn | | Avg Accuracy | |
|---|---|---|---|---|---|---|
| **Method** | Clean | PGD | Clean | PGD | Clean | PGD |
| DANN | 67.5 | 0.3 | 78.5 | 0.5 | 73.0 | 0.4 |
| UDA+AT | 49.6 | 29.3 | 52.8 | 33.5 | 51.2 | 31.4 |
| Trades | 51.7 | 36.1 | 57.4 | 45.6 | 54.6 | 40.9 |
| Mart | 50.8 | 38.4 | 56.0 | 46.8 | 53.4 | 42.6 |
| ARTUDA | 54.9 | 42.7 | 59.5 | 47.2 | 57.2 | 45.0 |
| SRoUDA | 51.3 | 35.4 | 61.6 | 49.3 | 56.5 | 42.4 |
| DART | 65.6 | 51.2 | 73.8 | 60.1 | 69.7 | 55.7 |
| **Ours** | **69.5** | **57.8** | **79.1** | **66.4** | **74.3** | **62.1** |

Table 3: An illustration of comparison on PGD 20 attack at $\epsilon = 8/255$ using VisDA dataset.

| Method | Clean | FGSM | PGD 10 | PGD 20 | CW$\infty$ |
|---|---|---|---|---|---|
| DANN | 78.5 | 19.5 | 1.2 | 0.4 | 0.06 |
| UDA+AT | 52.8 | 45.0 | 37.4 | 33.5 | 30.8 |
| ARTUDA | 59.6 | 54.3 | 50.4 | 47.2 | 44.7 |
| SRoUDA | 61.6 | 57.2 | 53.8 | 49.3 | 46.5 |
| DART | 73.8 | 68.6 | 64.5 | 60.1 | 57.3 |
| **Ours** | **79.1** | **69.7** | **67.8** | **66.4** | **61.5** |

Table 4: Robustness comparison against different adversarial attacks (FGSM, PGD 10, PGD 20, CW$\infty$) for various methods using VISDA dataset (Real→synthetic).

| UDA | DAN | | JAN | | DANN | |
|---|---|---|---|---|---|---|
| S→T | Re→Cl | | Re→Cl | | Re→Cl | |
| **Method** | Clean | PGD | Clean | PGD | Clean | PGD |
| Baseline | 53.0 | 0.0 | 54.3 | 0.0 | 55.5 | 0.0 |
| ARTUDA | 49.3 | 28.2 | 50.4 | 34.8 | 54.7 | 33.9 |
| SRoUDA | 48.9 | 31.7 | 49.5 | 34.1 | 51.6 | 36.2 |
| DART | 52.8 | 34.4 | 53.9 | 36.7 | 54.1 | 38.5 |
| **Ours** | **55.1** | **39.6** | **56.3** | **41.2** | **57.5** | **43.4** |

Table 5: UDA Baseline comparison on PGD 20 attack at $\epsilon = 8/255$ on Realworld→Clipart of OfficeHome dataset.

posed method. Additionally, both CAM without SPLR and SPLR without CAM demonstrated improvements in robustness and accuracy over the DANN and UDA+AT methods.

| S→T | Photo→Clipart | | Photo →Sketch | |
|---|---|---|---|---|
| **Method** | **Clean** | **PGD** | **Clean** | **PGD** |
| DANN | 80.2 | 0.7 | 74.5 | 0.6 |
| UDA + AT | 79.5 | 61.2 | 69.5 | 59.6 |
| CAM w/o SPLR | 83.3 | 64.6 | 75.5 | 63.2 |
| SPLR w/o CAM | 84.8 | 68.9 | 77.1 | 67.7 |
| **CAM+SPLR** | **86.7** | **71.5** | **80.5** | **74.3** |

Table 6: Results on component ablations of CAM+SPLR at perturbation size $\epsilon = 8/255$ on PACs dataset

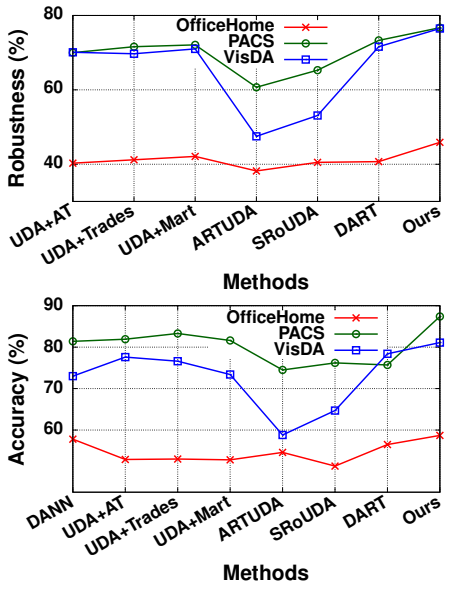


Figure 4: Comparison of robustness and accuracy on the OfficeHome, VisDA, PACS dataset.

**Comparison with using different UDA baseline.** We leverage different UDA baseline methods to initialize the pre-trained model, which is trained on source-target pairs for the domain adaptation task. To compare this, we use the DAN (Long et al. 2015), JAN (Long et al. 2017) and DANN (Ganin et al. 2016) as the UDA-baseline on the source target pair of OfficeHome dataset, specifically

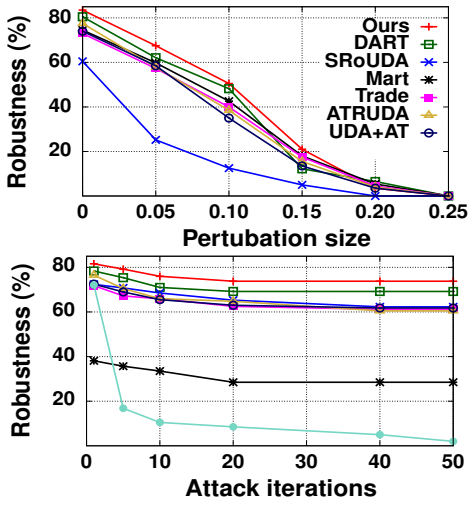


Figure 5: Robustness as a function of perturbation size and attack iteration for different methods on PACS (Sketch→Cartoon).

Realworld→Clipart. We compare our method with the existing state-of-the-art methods: ARTUDA, SRoUDA, and DART, as shown in Table 5. We observe that our method outperforms all the UDA baselines in comparison to other state-of-the-art. However, the accuracy and robustness vary with the UDA baselines. Thus, a better UDA baseline can help improve the robustness and accuracy of our method.

**Evaluation on attack budget.** We evaluate the scalability of our CAM+SPLR method to various attack budgets through two aspects: 1) increasing the perturbation size and 2) increasing the number of attack iterations while keeping the $\epsilon = 2/255$. In Figure 5(c), it is evident that increasing the perturbation budget size directly impacts the robustness accuracy. We conducted evaluations on the PACS dataset (Sketch→Cartoon) and observed that all methods experienced a decrease in robustness with the perturbation size. However, our method consistently outperformed other methods across all perturbation sizes. When the perturbation sizes became sufficiently large, robustness dropped to zero for all. Additionally, Figure 5(d) demonstrates that 20 attack iterations can achieve the strongest PGD attack within the given perturbation size of $2/255$ observed with all method also including DANN method.

## Conclusion

In this paper, we tackle the challenge of enhancing both robustness and accuracy in unsupervised domain adaptation (UDA) models. We introduce a novel self-training method called Consistent Attention Mapping with Self Pseudo Label Refinement (CAM+SPLR), which incorporates a two-step gradient update process. This method is specifically designed to improve adversarial robustness and accuracy in UDA by encouraging the attention maps of both clean and adversarial examples to be consistent. Additionally, it refines the pseudo labels to minimize inaccuracies, further boosting the model's performance.

## Acknowledgments

We thank SERB MTR/2021/604 and ECR/2017/002419 for supporting this research.

## References

Choi, J.; Jeong, M.; Kim, T.; and Kim, C. 2019. Pseudo-Labeling Curriculum for Unsupervised Domain Adaptation. arXiv:1908.00262.

Dong, Y.; Liao, F.; Pang, T.; Su, H.; Zhu, J.; Hu, X.; and Li, J. 2018. Boosting adversarial attacks with momentum. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 9185–9193. IEEE.

Ganin, Y.; and Lempitsky, V. 2015. Unsupervised Domain Adaptation by Backpropagation. In Bach, F.; and Blei, D., eds., *Proceedings of the 32nd International Conference on Machine Learning*, volume 37, 1180–1189.

Ganin, Y.; Ustinova, E.; Ajakan, H.; Germain, P.; Larochelle, H.; Laviolette, F.; March, M.; and Lempitsky, V. 2016. Domain-Adversarial Training of Neural Networks. *Journal of Machine Learning Research*, 17(59): 1–35.

Goodfellow, I. J.; Shlens, J.; and Szegedy, C. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep Residual Learning for Image Recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770–778. IEEE.

Junguang Jiang, B. F. M. L., Baixu Chen. 2020. Transfer-Learning-library.

Li, D.; Yang, Y.; Song, Y.-Z.; and Hospedales, T. M. 2017. Deeper, Broader and Artier Domain Generalization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 5542–5550. IEEE.

Lo, S.-Y.; and Patel, V. 2022. Exploring Adversarially Robust Training for Unsupervised Domain Adaptation. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, 4093–4109. Springer.

Lo, S.-Y.; and Patel, V. M. 2021. Multav: Multiplicative adversarial videos. In *2021 17th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, 1–6.

Long, M.; Cao, Y.; Wang, J.; and Jordan, M. 2015. Learning Transferable Features with Deep Adaptation Networks. In *Proceedings of the International Conference on Machine Learning (ICML)*, 97–105. PMLR.

Long, M.; CAO, Z.; Wang, J.; and Jordan, M. I. 2018. Conditional Adversarial Domain Adaptation. In *Advances in Neural Information Processing Systems*, 1–11. Curran Associates, Inc.

Long, M.; Zhu, H.; Wang, J.; and Jordan, M. I. 2017. Deep Transfer Learning with Joint Adaptation Networks. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2208–2217. PMLR.

Madry, A.; Makelov, A.; Schmidt, L.; Tsipras, D.; and Vladu, A. 2017. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*.

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

Saito, K.; Watanabe, K.; Ushiku, Y.; and Harada, T. 2018. Maximum Classifier Discrepancy for Unsupervised Domain Adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Sun, B.; and Saenko, K. 2016. Deep CORAL: Correlation Alignment for Deep Domain Adaptation. In *Computer Vision – ECCV 2016 Workshops*, 443–450. Cham: Springer International Publishing.

Szegedy, C.; Zaremba, W.; Sutskever, I.; Bruna, J.; Erhan, D.; Goodfellow, I.; and Fergus, R. 2013. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*.

Wang, J.; Tian, K.; Ding, D.; Yang, G.; and Li, X. 2021. Unsupervised Domain Expansion for Visual Categorization. *ACM Transactions on Multimedia Computing Communications and Applications (TOMM)*. In press.

Wang, Y.; Hazimeh, H.; Ponomareva, N.; Kurakin, A.; Hammoud, I.; and Arora, R. 2024. DART: A Principled Approach to Adversarially Robust Unsupervised Domain Adaptation. *arXiv preprint arXiv:2402.11120*.

Wang, Y.; Zou, D.; Yi, J.; Bailey, J.; Ma, X.; and Gu, Q. 2019. Improving Adversarial Robustness Requires Revisiting Misclassified Examples. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 1–14.

Wilson, G.; and Cook, D. J. 2020. A Survey of Unsupervised Deep Domain Adaptation. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 11(5).

Xu, R.; Li, G.; Yang, J.; and Lin, L. 2019. Larger Norm More Transferable: An Adaptive Feature Norm Approach for Unsupervised Domain Adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 1426–1435.

Yang, J.; Li, C.; An, W.; Ma, H.; Guo, Y.; Rong, Y.; Zhao, P.; and Huang, J. 2021a. Exploring robustness of unsupervised domain adaptation in semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 9194–9203.

Yang, J.; Shi, S.; Wang, Z.; Li, H.; and Qi, X. 2021b. ST3D: Self-Training for Unsupervised Domain Adaptation on 3D Object Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 10368–10378.

Zagoruyko, S.; and Komodakis, N. 2017. Paying More Attention to Attention: Improving the Performance of Convolutional Neural Networks via Attention Transfer. In *Proceedings of the International Conference on Learning Representations (ICLR)*. OpenReview.net.

Zhang, H.; Yu, Y.; Jiao, J.; Xing, E.; El Ghaoui, L.; and Jordan, M. 2019. Theoretically principled trade-off between robustness and accuracy. In *International conference on machine learning*, 7472–7482. PMLR.

Zhu, W.; Yin, J.-L.; Chen, B.-H.; and Liu, X. 2023. SRoUDA: Meta Self-Training for Robust Unsupervised Domain Adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 3852–3860. The MIT Press.