

SHUFFLE PATCHMIX AUGMENTATION WITH CONFIDENCE-MARGIN WEIGHTED PSEUDO-LABELS FOR ENHANCED SOURCE-FREE DOMAIN ADAPTATION

Prasanna Reddy Pulakurthi¹, Majid Rabbani¹, Jamison Heard¹, Sohail Dianat¹,
Celso M. de Melo², and Raghuveer Rao²

¹Rochester Institute of Technology, Rochester, NY, USA

²DEVCOM Army Research Laboratory, Adelphi, MD, USA

ABSTRACT

This work investigates Source-Free Domain Adaptation (SFDA), where a model adapts to a target domain without access to source data. A new augmentation technique, Shuffle PatchMix (SPM), and a novel reweighting strategy are introduced to enhance performance. SPM shuffles and blends image patches to generate diverse and challenging augmentations, while the reweighting strategy prioritizes reliable pseudo-labels to mitigate label noise. These techniques are particularly effective on smaller datasets like PACS, where overfitting and pseudo-label noise pose greater risks. State-of-the-art results are achieved on three major benchmarks: PACS, VisDA-C, and DomainNet-126. Notably, on PACS, improvements of **7.3%** (**79.4%** to **86.7%**) and **7.2%** are observed in single-target and multi-target settings, respectively, while gains of **2.8%** and **0.7%** are attained on DomainNet-126 and VisDA-C. This combination of advanced augmentation and robust pseudo-label reweighting establishes a new benchmark for SFDA. The code is available at: <https://github.com/PrasannaPulakurthi/SPM>.

Index Terms— Source-Free Domain Adaptation, Classification, Contrastive Learning, Pseudo-Labels, Self-Training

1. INTRODUCTION

Deep neural networks have achieved remarkable success in tasks where the training and test data share similar distributions. However, domain shifts can degrade network performance considerably [1]. To address this challenge, Domain Adaptation (DA) techniques have been developed to improve model performance in the target domain by leveraging knowledge from a related source domain. The most studied DA paradigm, Unsupervised Domain Adaptation (UDA), adapts models from source to target domains where labeled target data is unavailable. A key strategy in UDA involves aligning feature distributions between source and target domains to reduce domain discrepancies [2, 3, 4, 5].

In many real-world scenarios, access to source data during adaptation is restricted due to privacy, security, or logistical constraints. This has led to Source-Free Domain Adaptation (SFDA), a subfield of UDA where a pre-trained source model is adapted to a target domain without source data. The lack of source data makes adaptation more challenging, requiring innovative techniques to align the model to the target domain while retaining knowledge learned from the source domain.

Advancements, such as AdaContrast [6], have demonstrated the effectiveness of self-training by pseudo-labeling, jointly trained with contrastive learning on strongly augmented target data to exploit the pairwise relationships among target samples. Despite their effectiveness, these methods heavily depend on the quality of pseudo-labels. If noisy pseudo-labels are treated equally, errors can be amplified, negatively impacting adaptation performance.

To address these limitations, this work proposes two key innovations: (1) **Confidence-Margin Reweighting** strategy that leverages both confidence (probability of the top-1 class prediction) and margin (difference between top-1 and top-2 class probabilities) of the pseudo-labels for self-training, which prioritizes reliable pseudo-labels to reduce the impact of label noise, and (2) **Shuffle PatchMix (SPM)** that is a novel augmentation strategy to enhance target domain representation by introducing diverse and challenging augmentations. Together, these contributions lead to significant performance gains on three benchmark datasets.

2. RELATED WORK

Domain Adaptation: In UDA, several methods have been proposed to reduce domain discrepancies by aligning the feature distributions between source and target domains, MMD [7], MCC [8], MDD [9], and CMD [10]. Additionally, GAN-based methods use adversarial training to align the distributions in feature space [2, 4] and image space [11]. However, all these methods require access to the source data. In contrast, SFDA methods adapt to unlabeled target data without source samples. Several significant works have been proposed to address SFDA, including leveraging entropy minimization by TENT [12], class prototypes by BAIT [13],

This research was supported by DEVCOM Army Research Laboratory under contract W911QX-21-D-0001.

latent-prototype generation USFDA [14], and self-training by pseudo-labeling by SHOT [15]. Significant improvements are achieved by AdaContrast [6] using self-training by pseudo-labels along with contrastive learning.

Learning with Noisy Pseudo-Labels: The limitation of previous methods such as [6] is that all the pseudo-labels equally contribute to the loss without considering the noise associated with the pseudo-labels. Various approaches were proposed to address noisy pseudo-labels. In semi-supervised learning, FixMatch [16], a fixed confidence threshold was used to filter pseudo-labels, while FreeMatch [17] improved upon this by dynamically adapting the threshold based on prediction uncertainty. A meta-learning-based method [18] reweighted the loss but relied on a noise-free validation set, making it unsuitable for the SFDA setting. In SFDA, NEL [19] method employed a pseudo-label refinement framework that relied on an ensemble of pseudo-label predictions generated from different augmentations. While this method improved the robustness of pseudo-labels, it incurred a significant computational cost due to the need for multiple augmentations and predictions. UPA [20] filters low-confidence pseudo-labels via uncertainty awareness, achieving the best results on VISDA-C. In contrast, our reweighting strategy assigns continuous weights to pseudo-labels by leveraging both confidence and margin from aggregated neighboring predictions, ensuring more stable and reliable adaptation.

Data Augmentation: Data augmentation techniques are extensively utilized in domain adaptation to enhance model generalization, while contrastive learning aids in learning robust features. Recently, mixing-based data augmentation techniques have gained popularity, such as Mixup [21], CutMix [22], Manifold-Mixup [23], Tokenmix [24], and TransMix [25]. In CNN-based UDA tasks, several methods [26, 27, 28] also use mixing-based strategies by linearly mixing the source and target domain data. In the SFDA setting, notable mixing-based methods such as ProxyMix [29] construct a proxy source domain by selecting confident target samples. Most recently, Improved SFDA [30] introduced a learnable data augmentation via a teacher-student framework, whereas SF(DA)² [31] generates an augmentation graph in feature space to achieve the best results on DomainNet-126.

Our proposed SPM augmentation differs from prior patch-shuffling methods [24, 25] by seamlessly blending overlapping patches, effectively reducing blocking artifacts. In contrast to [24, 25], which mix patches across different images and were designed for supervised training, SPM operates intra-image and is inherently suited for source-free domain adaptation. By generating challenging adaptations, SPM outperforms previous methods to set a new state-of-the-art.

3. METHOD

In the SFDA setting, the source model $g_s(\cdot)$ is trained on labeled source data $\{\mathbf{x}_s^i, y_s^i\}_{i=1}^{n_s}$, where $\mathbf{x}_s^i \in \mathcal{X}_s$ represent the

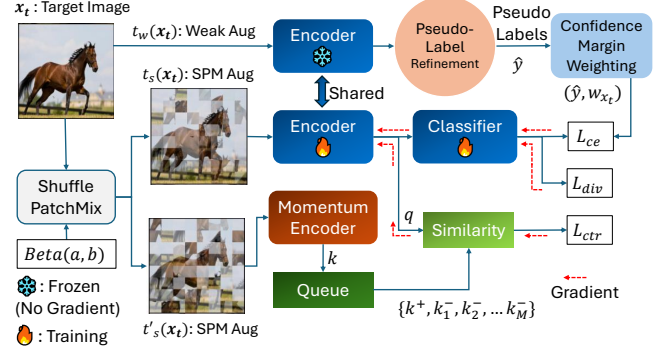


Fig. 1. An overview of the adaptation method.

source images and $y_s^i \in \mathcal{Y}_s$ represent the source labels. During adaptation, the target images $\{\mathbf{x}_t^i\}_{i=1}^{n_t} \in \mathcal{X}_t$ are available, while the underlying target labels $\{y_t^i\}_{i=1}^{n_t} \in \mathcal{Y}_t$ are accessed only for evaluation. SFDA aims to train a model $g_t(\cdot) = h_t(f_t(\cdot))$, comprising of an encoder network $f_t(\cdot)$ and a classification network $h_t(\cdot)$, to classify the target data. During adaptation, the target model's parameters θ_t are initialized with the source model's parameters θ_s .

Adaptation Method Overview: As illustrated in Figure 1, the proposed adaptation method builds on the framework introduced in [6], with two key modifications: the integration of the SPM augmentation into the strong augmentation phase, and reweighting the self-training loss using pseudo-label confidence and margin estimation. The process begins by generating two strong augmentations, $t_s(\mathbf{x}_t)$ and $t'_s(\mathbf{x}_t)$, using SPM, along with a weak augmentation, $t_w(\mathbf{x}_t)$. The weakly augmented image is processed through the encoder $f_t(\cdot)$ to extract target features, which are then refined through the pseudo-labeling process to produce pseudo-labels, $\hat{y}_t$. The weight associated with each pseudo-label is determined using the proposed Confidence-Margin method. These refined pseudo-labels, $\hat{y}_t$, along with their weights, $w_{\mathbf{x}_t}$, are used to train the model $g_t(\cdot)$ to classify the strongly augmented data, $t_s(\mathbf{x}_t)$. The training optimizes a composite loss function that includes classification, diversity, and contrastive losses. This approach enables effective adaptation to the target domain by leveraging the diverse augmentations introduced by SPM and appropriately weighting the contribution of the pseudo-labels.

Shuffle PatchMix (SPM): This work introduces SPM, a novel augmentation strategy that creates strong augmentations for contrastive learning. As illustrated in Figure 2, SPM first divides the target image into patches, denoted as $\mathbf{x}_t^p$. These patches are randomly shuffled to form $\mathbf{x}_t^{sp}$. The original and shuffled patches are linearly combined using a scalar parameter λ , which is randomly generated from a Beta distribution. The Beta distribution has been successfully applied in the mixing-based augmentation method [21, 27]. Additional standard strong augmentations from [32] are applied.

The strength of patch mixing can be controlled with hyperparameters such as the number of patches ν , and param-

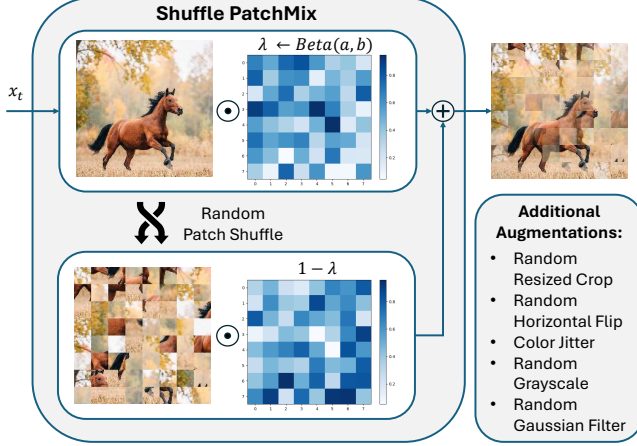


Fig. 2. Overview of the proposed Shuffle PatchMix method.

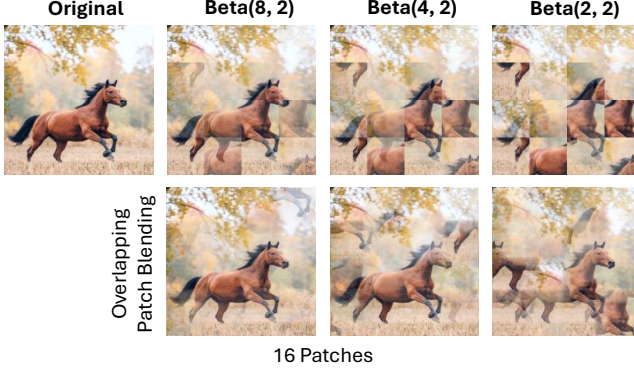


Fig. 3. (a) Different mixing strength SPM images (top). (b) SPM images with overlapping patch blending (bottom).

ters of the Beta distribution a and b . The mixing parameter λ follows a Beta distribution, $\lambda \sim \text{Beta}(a, b)$, where $a, b > 0$, with the probability density function (PDF) given by:

$$f(\lambda; a, b) = \frac{\lambda^{a-1}(1-\lambda)^{b-1}}{\int_0^1 u^{a-1}(1-u)^{b-1} du} \quad \text{for } 0 \leq \lambda \leq 1. \quad (1)$$

The effects of varying patch mixing strengths, determined by different values of a and b , are illustrated in Figure 3(a).

Adaptive Mixing Strength: Our simulations demonstrate that gradually decreasing the mean of the random variable λ during the training (i.e., increasing the mixing strength) allows the model to better adapt to the SPM augmentation distribution. To achieve this, the parameter a in the Beta distribution $\text{Beta}(a, b)$ is progressively reduced from a start value a_s to an end value a_e ($a_e \leq a_s$) during the training.

Mitigating Blocking Artifacts: As shown in Figure 3(a), patch shuffling introduces blocking artifacts. To address this, an overlapping patch blending approach is proposed, where patches are extracted with a linear dimension 30% larger than the intended size and blended linearly with neighboring patches. This effectively reduces blocking artifacts, resulting in smoother transitions, as illustrated in Figure 3(b). Additional visualizations are provided in the supplement ([link](#)).

Although applying SPM to all strongly augmented images can maximize diversity, two potential risks are introduced: reduced exposure to standard strong augmentations [32], and the generation of overly unrealistic samples that may hinder model training. Therefore, SPM is applied to a large fraction ρ of the standard strong augmentations but not the entire set.

Pseudo-label Refinement: During adaptation, pseudo-labels are generated for the weakly augmented unlabeled target data $t_w(x_t)$ using the target model initialized with source weights, allowing knowledge transfer while gradually adapting to the target domain. This is performed on a per-batch basis through a nearest-neighbor soft voting strategy similar to [6] using a memory queue Q_w representing the target feature space. For each target image, its weakly augmented version is encoded into a feature vector, which is then used to find the nearest neighbors in the target feature space. The pseudo-label is refined by averaging the probabilities of these neighbors, followed by an argmax operation.

To enable the nearest-neighbor search, the memory queue Q_w is maintained by storing features and probability predictions of the weakly augmented target samples, updated at each mini-batch. The feature space is stabilized using a slowly changing momentum model $g'_t(\cdot)$. The momentum model’s parameters θ'_t are initialized with source weights θ_s , and are updated with momentum $m = 0.999$ at each mini-batch step according to the update rule: $\theta'_t \leftarrow m\theta'_t + (1 - m)\theta_t$.

Loss Reweighting by Pseudo-Label Confidence-Margin Estimation: The pseudo-labels are obtained by averaging predictions of nearest neighbors and serve as self-supervised signals to classify strongly augmented images. Uniformly weighting all pseudo-labels can hinder adaptation, as noisy labels negatively affect the training process.

To address this, we introduce the Confidence-Margin reweighting strategy, where **Confidence**, defined as the probability of the top prediction (p_{top1}), and **Margin** (Δ), defined as the difference between the top two prediction probabilities ($\Delta = p_{\text{top1}} - p_{\text{top2}}$) for the pseudo-labels. If all the neighbors predominantly belong to the same class, the pseudo-label is considered highly reliable and should receive a high weight. This reliability is reflected in both the confidence and margin. A high margin corresponds to a more reliable pseudo-label, as it indicates a clear distinction between the most likely class and the second most likely class. Conversely, a low margin signals greater uncertainty and reduced reliability.

The weights are computed as follows:

$$w_{x_t} = \underbrace{p_{\text{top1}}}_{\text{confidence}} \underbrace{\Delta}_{\text{margin}} \exp(\Delta). \quad (2)$$

All symbols in Equation 2 are scalars, evaluated independently for each target image x_t . The weighting function multiplies the confidence p_{top1} and the margin Δ , then scales their product by the exponential of margin $\exp(\Delta)$. The term $\exp(\Delta)$ supplies a smooth, margin-aware gain that exponentially enlarges the contribution of samples with large mar-

gins, whose pseudo-labels are statistically the most trustworthy. Samples with small margins are down-weighted, so that unreliable predictions exert little influence on the gradients. This continuous weighting eliminates the need for hand-tuned confidence thresholds and focuses learning on reliable targets, leading to faster and more stable adaptation. Since pseudo-labels tend to be highly noisy in the early training stages, the reweighting strategy is gradually introduced as training progresses to ensure stability.

Weighted Classification Loss: The refined pseudo-labels $\hat{y}_t$ generated from the weakly augmented data, along with their corresponding weights w_{x_t} , are used to supervise the model’s prediction for the strongly augmented version of the target data, as illustrated in Figure 1. The proposed weighted classification loss is formulated as:

$$L_{ce} = -\mathbb{E}_{x_t \in \mathcal{X}_t} \left[w_{x_t} \sum_{c=1}^C \hat{y}_t^c \log p_q^c \right], \quad (3)$$

where C is the number of classes and p_q^c is the predicted probability of class c for the strongly-augmented image $t_s(x_t)$.

Overall Loss: The proposed weighted classification is jointly optimized with two additional losses from [6]: the contrastive L_{ctr} loss and the diversity L_{div} loss. The total loss function is defined as:

$$L_t = L_{ce} + L_{ctr} + L_{div}. \quad (4)$$

The diversity loss (L_{div}) serves as a regularization term to prevent model collapse by promoting diverse predictions. The contrastive loss (L_{ctr}) utilizes the SPM module to generate two augmentations, which are processed by the target and momentum encoders. A memory queue stores past keys, optimizing the model by pulling positive pairs closer while pushing negative pairs apart. Some of the images from the negative pairs may share the same pseudo-label; these same-class negative pairs are excluded. For more details refer to [6].

4. EXPERIMENTAL SETUP

Datasets: The PACS [33], VisDA-C [34], and DomainNet-126 [35] datasets are used to evaluate the proposed method. For DomainNet-126, seven domain shifts were constructed from four domains (Real, Sketch, Clipart, Painting) following the protocol described in [6], with the top-1 accuracy (%) and the average of the seven shifts reported. For VisDA-C, the per-class top-1 accuracies (%) and their average are compared. For PACS, the evaluation is performed following the single-target and multi-target adaptation protocol outlined in [19]. Since [6] had not originally reported results for PACS, the publicly available version of the code (after being verified on the published results in [6]) was used to generate the baseline PACS results in Tables 1 and 2, as they outperformed the best-reported results in NEL [19].

Backbone: Following the standard DA protocols, all the experiments use ResNet-101 backbone for VisDA-C, ResNet-50 for DomainNet-126, and ResNet-18 for PACS.

Table 1. Classification accuracy (%) on PACS for the single-target setting with ResNet-18. Legend: **P**: Photo, **A**: Art-Painting, **C**: Cartoon, and **S**: Sketch. The highest accuracies are in **bold**. * indicates our reproduced results using [6].

Method	P→A	P→C	P→S	A→P	A→C	A→S	Avg.
NEL [19]	82.6	80.5	32.3	98.4	84.3	56.1	72.4
AdaContrast [6]*	81.3	72.2	66.7	98.7	79.7	77.9	79.4
SPM (Ours)	89.7	82.3	74.5	99.1	87.9	86.4	86.7

Table 2. Classification accuracy (%) on PACS for the multi-target setting with ResNet-18 backbone. The highest value is **bolded**. * indicates our reproduced results using [6].

Multi-Target UDA		P → A, C, S			A → P, C, S			
Method	SF	A	C	S	P	C	S	Avg.
1-NN	×	15.2	18.1	25.6	22.7	19.7	22.7	20.7
ADDA [2]	×	24.3	20.1	22.4	32.5	17.6	18.9	22.6
DSN [36]	×	28.4	21.1	25.6	29.5	25.8	26.8	25.8
ITA [37]	×	31.4	23.0	28.2	35.7	27.0	28.9	29.0
KD [38]	×	24.6	32.2	33.8	35.6	46.6	57.5	46.6
NEL [19]	✓	80.1	76.1	25.9	96.0	82.8	49.8	68.4
AdaContrast [6]*	✓	70.1	77.9	62.9	95.9	72.7	72.9	75.4
SPM (Ours)	✓	85.2	89.2	66.4	97.7	76.4	81.0	82.6

Hyperparameters: All hyperparameters are similar to those in [6], except for the following deviations, which were found to yield the optimal performance. The learning rate was fixed at 2×10^{-4} for the SGD optimizer, the number of nearest neighbors was set to 3, and the model was trained for 100 epochs for PACS and 50 epochs for DomainNet-126 and VisDA-C. The starting value a_s of the Beta distribution was set to 8 for DomainNet-126 and PACS, and 4 for VisDA-C, with a ρ value of 0.8 used across all experiments. SPM randomly selected the number of patches ν from $\{2^2, 4^2, 8^2, 16^2\}$ in each mini-batch to enhance diversity.

5. RESULTS

PACS Results: Table 1 and Table 2 summarize the performance of our proposed method on the PACS dataset under single-target and multi-target settings. For the single-target setting, our method achieves an average accuracy of **86.7%**, outperforming the baseline AdaContrast [6] by a significant margin of **7.3%**. The most notable improvements over baseline are observed in P→C (**82.3%** vs. **72.2%**) and P→S (**74.5%** vs. **66.7%**), demonstrating superior generalization across domain shifts. In the multi-target setting, our method achieves the highest average accuracy of **82.6%**, surpassing AdaContrast [6] by **7.2%**. Notable improvements are observed in challenging domain shifts, including P→A (**85.2%** vs. **70.1%**) and A→S (**81.0%** vs. **72.9%**), demonstrating the model’s ability to adapt effectively to multiple target domains.

VisDA-C Results: Table 3 presents the classification accuracy of the VisDA-C dataset. Our method outperforms the best-performing method [20] by **0.7%**, achieving the highest average accuracy of **89.4%**. Our method achieves the best or

Table 3. Classification accuracy (%) across various methods on VisDA-C (ResNet-101). SF stands for Source-Free. The highest values are shown in **bold** and the second highest are underlined.

Method	SF	plane	bycl	bus	car	horse	knife	mcycl	person	plant	sktbrd	train	truck	Avg.
DANN [3]	×	81.9	77.7	82.8	44.3	81.2	29.5	65.1	28.6	51.9	54.6	82.8	7.8	57.4
CDAN [4]	×	85.2	66.9	83.0	50.8	84.2	74.9	88.1	74.5	83.4	76.0	81.9	38.0	73.9
SWD [39]	×	90.8	82.5	81.7	70.5	91.7	69.5	86.3	77.5	87.4	63.6	85.6	29.2	76.4
MCC [8]	×	88.7	80.3	80.5	71.5	90.1	93.2	85.0	71.6	89.4	73.8	85.0	36.9	78.8
CAN [40]	×	97.0	87.2	82.5	74.3	97.8	96.2	90.8	80.7	96.6	96.3	87.5	59.9	87.2
FixBi [26]	×	96.1	87.8	90.5	90.3	96.8	95.3	92.8	88.7	97.2	94.2	90.9	25.7	87.2
Source only	-	57.2	11.1	42.4	66.9	55.0	4.4	81.1	27.3	57.9	29.4	86.7	5.8	43.8
MA [41]	✓	94.8	73.4	68.8	74.8	93.1	95.4	88.6	84.7	89.1	84.7	83.5	48.1	81.6
BAIT [13]	✓	93.7	83.2	84.5	65.0	92.9	95.4	88.1	80.8	90.0	89.0	84.0	45.3	82.7
SHOT [15]	✓	95.3	87.5	78.7	55.6	94.1	94.2	81.4	80.0	91.8	90.7	86.5	59.8	83.0
AdaContrast [6]	✓	97.0	84.7	84.0	77.3	96.7	93.8	91.9	84.8	94.3	93.1	<u>94.1</u>	49.7	86.8
SF(DA) ² [31]	✓	96.8	89.3	82.9	<u>81.4</u>	96.8	95.7	90.4	81.3	95.5	93.7	88.5	<u>64.7</u>	88.1
Improved SFDA [30]	✓	<u>97.5</u>	91.4	87.9	79.4	<u>97.2</u>	97.2	<u>92.2</u>	83.0	<u>96.4</u>	94.2	91.1	53.0	88.4
UPA [20]	✓	97.0	<u>90.4</u>	82.6	65.0	96.7	<u>96.7</u>	91.0	87.0	96.8	96.5	89.2	75.0	<u>88.7</u>
SPM (Ours)	✓	98.1	87.9	<u>86.7</u>	86.2	97.7	94.8	93.3	<u>85.6</u>	95.9	<u>95.6</u>	95.5	55.3	89.4

Table 4. Classification accuracy (%) on 7 domain shifts of the DomainNet-126 dataset (ResNet-50). Legend: **R**: Real, **C**: Clipart, **P**: Painting, and **S**: Sketch.

Method	SF	R→C	R→P	P→C	C→S	S→P	R→S	P→R	Avg.
MCC [8]	×	44.8	65.7	41.9	34.9	47.3	35.3	72.4	48.9
Source only	-	55.5	62.7	53.0	46.9	50.1	46.3	75.0	55.6
TENT [12]	✓	58.5	65.7	57.9	48.5	52.4	54.0	67.0	57.7
SHOT [15]	✓	67.7	68.4	66.9	60.1	66.1	59.9	80.8	67.1
AdaContrast [6]	✓	<u>70.2</u>	<u>69.8</u>	<u>68.6</u>	58.0	65.9	61.5	80.5	67.8
UPA [20]	✓	68.6	69.5	67.6	60.9	66.8	61.5	<u>80.9</u>	68.0
SF(DA) ² [31]	✓	67.7	59.6	67.8	83.5	60.2	68.8	70.5	68.3
SPM (Ours)	✓	74.2	71.9	72.5	<u>62.4</u>	68.1	<u>66.4</u>	81.8	71.1

second-best performance in **8 out of 12** classes.

DomainNet-126 Results: Table 4 summarizes classification accuracy on the DomainNet-126 dataset. Our method achieves the highest average accuracy of **71.1%** and outperforms the SOTA [31] by **2.8%**. In addition, our method achieves the best performance on **5 of 7** domain shifts.

Ablation Studies: An ablation study was conducted on DomainNet-126 and PACS to assess the impact of SPM augmentation, patch overlap, and the reweighting strategy. The results, summarized in Table 5, demonstrate the effectiveness of each component in improving classification accuracy.

The baseline AdaContrast achieves an accuracy of **67.8%** on DomainNet-126 and **79.4%** on PACS. When the reweighting strategy is applied, accuracy improves to **69.1%** as unreliable pseudo-labels are down-weighted, while higher weights are assigned to more reliable ones during self-training. The SPM augmentation enhances performance by introducing challenging augmentations that encourage the network to learn class-invariant features, increasing accuracy to **70.2%**. Additionally, patch overlapping mitigates blocking artifacts, resulting in a slight but consistent improvement to **70.4%**. Since the strength of SPM augmentation increases progressively during training, the model is trained for an extended period, as detailed in Section 4, allowing for better adaptation to varying augmentation distributions. When all components

Table 5. Ablation studies of sub-components of the proposed method measured by classification accuracy (%) on DomainNet-126 and PACS.

Baseline [6]	SPM Augmentation	Patch Overlap	Reweighting Strategy	DomainNet-126 Avg.	PACS Avg.
✓	×	×	×	67.8	79.4
✓	×	×	✓	69.1	81.8
✓	✓	×	×	70.2	83.8
✓	✓	✓	×	70.4	84.4
✓	✓	✓	✓	71.1	86.7

are combined, the highest accuracy of **71.1%** on DomainNet-126 and **86.7%** on PACS is achieved, demonstrating their complementary benefits in improving SFDA performance.

The largest improvement is observed in the smaller dataset, PACS, due to the following reasons. First, its limited size makes the model more prone to overfitting, so SPM is particularly effective in enhancing data diversity through patch mixing. Second, the impact of noisy pseudo-labels is more pronounced in smaller datasets, making Confidence-Margin reweighting essential for reducing the influence of uncertain labels while emphasizing confident predictions.

6. CONCLUSION

This paper introduces the SPM augmentation technique and the Confidence-Margin reweighting strategy to advance SFDA. Experimental results demonstrate that the proposed approach outperforms state-of-the-art methods across three benchmark datasets. Notably, the method is especially effective on smaller datasets like PACS, where limited target samples increase the risk of overfitting and label noise. The SPM augmentation technique enhances generalization by generating diverse training samples, while the reweighting strategy prioritizes reliable pseudo-labels, mitigating the impact of noisy predictions. Future work could explore extending SPM to broader domain adaptation paradigms, including self-supervised and semi-supervised learning.

7. REFERENCES

- [1] Joaquin Quionero-Candela, Masashi Sugiyama, Anton Schwaighofer, and Neil Lawrence, *Dataset Shift in Machine Learning*, The MIT Press, 2009.
- [2] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell, “Adversarial discriminative domain adaptation,” in *CVPR*, July 2017.
- [3] Yaroslav Ganin and Victor Lempitsky, “Unsupervised domain adaptation by backpropagation,” in *ICML*. PMLR, 2015, pp. 1180–1189.
- [4] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan, “Conditional adversarial domain adaptation,” *NeurIPS*, vol. 31, 2018.
- [5] Prasanna Reddy Pulakurthi, Sohail A Dianat, Majid Rabbani, Suyu You, and Raghuveer M Rao, “Unsupervised domain adaptation using feature aligned maximum classifier discrepancy,” in *Applications of Machine Learning 2022*. SPIE, 2022, vol. 12227, pp. 37–45.
- [6] Dian Chen, Dequan Wang, Trevor Darrell, and Sayna Ebrahimi, “Contrastive test-time adaptation,” in *CVPR*, 2022, pp. 295–305.
- [7] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael I Jordan, “Learning transferable features with deep adaptation networks,” in *ICML*. PMLR, 2015, pp. 97–105.
- [8] Ying Jin, Ximei Wang, Mingsheng Long, and Jianmin Wang, “Minimum class confusion for versatile domain adaptation,” in *ECCV*. Springer, 2020, pp. 464–480.
- [9] Yuchen Zhang, Tianle Liu, Mingsheng Long, and Michael Jordan, “Bridging theory and algorithm for domain adaptation,” in *ICML*. PMLR, 2019, pp. 7404–7413.
- [10] Werner Zellinger, Thomas Grubinger, Edwin Lughofer, Thomas Natschlager, and Susanne Saminger-Platz, “Central moment discrepancy for domain-invariant representation learning,” in *ICLR*, 2017.
- [11] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei Efros, and Trevor Darrell, “CyCADA: Cycle-consistent adversarial domain adaptation,” in *ICML*, 2018.
- [12] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell, “Tent: Fully test-time adaptation by entropy minimization,” in *ICLR*, 2021.
- [13] Shiqi Yang, Yaxing Wang, Luis Herranz, Shangling Jui, and Joost van de Weijer, “Casting a bait for offline and online source-free domain adaptation,” *Comput. Vis. Image Underst.*, vol. 234, no. C, 2023.
- [14] Jogendra Nath Kundu, Naveen Venkat, M. V. Rahul, and R. Venkatesh Babu, “Universal source-free domain adaptation,” in *CVPR*, 2020, pp. 4543–4552.
- [15] Jian Liang, Dapeng Hu, and Jiashi Feng, “Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation,” in *ICML*. PMLR, 2020, pp. 6028–6039.
- [16] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li, “Fixmatch: Simplifying semi-supervised learning with consistency and confidence,” *NeurIPS*, vol. 33, pp. 596–608, 2020.
- [17] Yidong Wang, Hao Chen, Qiang Heng, Wenxin Hou, Yue Fan, Zhen Wu, Jindong Wang, Marios Savvides, Takahiro Shinozaki, Bhiksha Raj, Bernt Schiele, and Xing Xie, “Freematch: Self-adaptive thresholding for semi-supervised learning,” in *ICLR*, 2023.
- [18] Mengye Ren, Wenyuan Zeng, Bin Yang, and Raquel Urtasun, “Learning to reweight examples for robust deep learning,” in *ICML*. PMLR, 2018, pp. 4334–4343.
- [19] Waqar Ahmed, Pietro Morerio, and Vittorio Murino, “Cleaning noisy labels by negative ensemble learning for source-free unsupervised domain adaptation,” in *WACV*, 2022, pp. 356–365.
- [20] Xi Chen, Haosen Yang, Huicong Zhang, Hongxun Yao, and Xiatian Zhu, “Uncertainty-aware pseudo-label filtering for source-free unsupervised domain adaptation,” *Neurocomputing*, vol. 575, 2024.
- [21] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz, “mixup: Beyond empirical risk minimization,” *ICLR*, 2018.
- [22] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo, “Cutmix: Regularization strategy to train strong classifiers with localizable features,” in *ICCV*, 2019.
- [23] Vikas Verma, Alex Lamb, Christopher Beckham, Amir Najafi, Ioannis Mitliagkas, David Lopez-Paz, and Yoshua Bengio, “Manifold mixup: Better representations by interpolating hidden states,” in *ICML*. PMLR, 2019, pp. 6438–6447.
- [24] Jihao Liu, Boxiao Liu, Hang Zhou, Hongsheng Li, and Yu Liu, “Tokenmix: Rethinking image mixing for data augmentation in vision transformers,” in *ECCV*. Springer, 2022, pp. 455–471.
- [25] Jie-Neng Chen, Shuyang Sun, Ju He, Philip HS Torr, Alan Yuille, and Song Bai, “Transmix: Attend to mix for vision transformers,” in *CVPR*, 2022, pp. 12135–12144.
- [26] Jaemin Na, Heechul Jung, Hyung Jin Chang, and Wonjun Hwang, “Fixbi: Bridging domain spaces for unsupervised domain adaptation,” in *CVPR*, 2021, pp. 1094–1103.
- [27] Jinjing Zhu, Haotian Bai, and Lin Wang, “Patch-mix transformer for unsupervised domain adaptation: A game perspective,” in *CVPR*, 2023.
- [28] Minghao Xu, Jian Zhang, Bingbing Ni, Teng Li, Chengjie Wang, Qi Tian, and Wenjun Zhang, “Adversarial domain adaptation with domain mixup,” in *AAAI*, 2020, vol. 34, pp. 6502–6509.
- [29] Yuhe Ding, Lijun Sheng, Jian Liang, Aihua Zheng, and Ran He, “Proxymix: Proxy-based mixup training with label refinery for source-free domain adaptation,” *Neural Networks*, vol. 167, pp. 92–103, 2023.
- [30] Yu Mitsuzumi, Akisato Kimura, and Hisashi Kashima, “Understanding and improving source-free domain adaptation from a theoretical perspective,” in *CVPR*, 2024, pp. 28515–28524.
- [31] Uiwon Hwang, Jonghyun Lee, Juhyeon Shin, and Sungroh Yoon, “Sf(da)2: Source-free domain adaptation through the lens of data augmentation,” in *ICLR*, 2024.
- [32] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He, “Improved baselines with momentum contrastive learning,” *arXiv preprint arXiv:2003.04297*, 2020.
- [33] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales, “Deeper, broader and artier domain generalization,” in *ICCV*, 2017.
- [34] 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.
- [35] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang, “Moment matching for multi-source domain adaptation,” in *CVPR*, 2019, pp. 1406–1415.
- [36] Konstantinos Bousmalis, George Trigeorgis, Nathan Silberman, Dilip Krishnan, and Dumitru Erhan, “Domain separation networks,” in *NeurIPS*, 2016, vol. 29, p. 343–351.
- [37] Behnam Gholami, Pritish Sahu, Ognjen Rudovic, Konstantinos Bousmalis, and Vladimir Pavlovic, “Unsupervised multi-target domain adaptation: An information theoretic approach,” *IEEE Transactions on Image Processing*, vol. 29, pp. 3993–4002, 2020.
- [38] Le Thanh Nguyen-Meidine, Atif Belal, Madhu Kiran, Jose Dolz, Louis-Antoine Blais-Morin, and Eric Granger, “Knowledge distillation methods for efficient unsupervised adaptation across multiple domains,” *Image and Vision Computing*, vol. 108, pp. 104096, 2021.
- [39] Chen-Yu Lee, Tanmay Batra, Mohammad Haris Baig, and Daniel Ulbricht, “Sliced wasserstein discrepancy for unsupervised domain adaptation,” in *CVPR*, 2019, pp. 10285–10295.
- [40] Guoliang Kang, Lu Jiang, Yi Yang, and Alexander G Hauptmann, “Contrastive adaptation network for unsupervised domain adaptation,” in *CVPR*, 2019, pp. 4893–4902.
- [41] Rui Li, Qianfen Jiao, Wenming Cao, Hau-San Wong, and Si Wu, “Model adaptation: Unsupervised domain adaptation without source data,” in *CVPR*, 2020, pp. 9641–9650.