Confidence-Guided Self-Training for Gradual Domain Adaptation

Akram Heidarizadeh[1] **Akram Awad**[1] **HanQin Cai**[2,3] **George Atia**[1,3]
[1]Department of Electrical and Computer Engineering, University of Central Florida, Orlando, FL 32816, USA
[2]School of Data, Mathematical, and Statistical Sciences, University of Central Florida, Orlando, FL 32816, USA
[3]Department of Computer Science, University of Central Florida, Orlando, FL 32816, USA

Abstract

Domain adaptation addresses the challenge of distributional shift between a labeled source domain and an unlabeled target domain. In gradual domain adaptation (GDA), the shift is assumed to occur through a sequence of intermediate domains, enabling smoother adaptation. A popular approach in this setting is self-training, where a model iteratively generates pseudo-labels for unlabeled data. However, pseudo-labeling errors can accumulate across rounds, especially under large shift, undermining generalization.

We develop a theoretical framework for self-training under gradual domain shift that explicitly quantifies and controls the pseudo-labeling error incurred at each round. Our first result is a modular generalization bound that decomposes the excess target risk into *coverage*, *pseudo-label error* (ε_k) on the accepted set, domain shift, sample complexity, and regularization. Unlike prior bounds, our analysis separates the coverage penalty (due to rejecting inputs) from the pseudo-label error (controlled by confidence calibration or margin filtering, including Tsybakov-type noise via margin decay or calibration assumptions). We also provide the first theoretical justification for percentile (quantile) thresholding schemes used in practice: such schedules directly control coverage while tightening ε_k, yielding a principled coverage–noise tradeoff. Under mild conditions, both terms accumulate only logarithmically, leading to improved generalization. We validate these insights across multiple GDA benchmarks, using both observed and OT-generated intermediate domains.

Proceedings of the 29[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2026, Tangier, Morocco. PMLR: Volume 300.

1 INTRODUCTION

Domain adaptation (DA) addresses the problem of learning from a labeled source domain while generalizing to an unlabeled target domain with a different distribution. A key challenge arises when the shift between the source and target domains is large (e.g., due to changes in pose, lighting, style, or temporal drift). In such cases, direct adaptation can fail due to distribution mismatch and poor generalization.

Gradual Domain Adaptation (GDA) mitigates this by introducing a sequence of intermediate domains that interpolate between the source and target (Zou et al., 2018; Kumar et al., 2020; He et al., 2024). By adapting incrementally, GDA allows the model to traverse the domain shift in smaller, more manageable steps, often leading to improved generalization.

Self-training is a popular approach in GDA and semi-supervised learning more broadly, where the model generates pseudo-labels on unlabeled data to refine itself over time. However, pseudo-labeling is highly sensitive to label noise, particularly under domain shift, leading to error accumulation that can compound over rounds. Despite the use of heuristics like confidence filtering (Sohn et al., 2020; Zhang et al., 2021), there is limited theoretical understanding of how such filtering affects generalization in iterative adaptation.

This paper provides a theoretical framework to understand and control pseudo-labeling error in GDA. We analyze iterative self-training through an accepted-set decomposition that, at each round, separates the *rejection rate* from the *pseudo-label substitution error* on the accepted set, alongside distribution shift, sample complexity, and regularization. The framework is filter-agnostic – compatible with either confidence or margin scores – and makes the per-round tradeoff explicit: stricter thresholds admit fewer points, in-

creasing the rejection rate yet lowering the pseudo-label error on those retained. We show that a simple percentile schedule that keeps roughly the top $1 - \frac{c}{k}$ fraction of the most reliable points lets the threshold loosen as the model improves; under standard calibration or margin-decay conditions, this choice drives *both* the rejected mass and the accepted-set error down at rate $O(1/k)$, so their round-wise contributions accumulate only logarithmically in K.

Contributions. Our main contributions are summarized as:

- **Generalization bound.** We establish a modular generalization bound for iterative self-training that is explicit in the rejection rate $(1 - \rho_k)$, the accepted-set pseudo-label error (ε_k), and additional distribution shift, sampling, and regularization terms.
- **Mechanisms to control each term.** We show how percentile thresholds directly set coverage ρ_k, while confidence calibration or margin-decay conditions (including Tsybakov-type decay) bound the accepted-set error ε_k. Under mild schedules, the cumulative contribution $\sum_k \big((1 - \rho_k) + 2\varepsilon_k\big)$ is $O(\log K)$.
- **Practical guidance and validation.** The analysis quantifies the coverage–noise trade-off and justifies adaptive thresholding used in practice. We validate the theory across multiple GDA benchmarks, including settings with observed and OT-generated intermediate domains.

2 RELATED WORK

Domain adaptation. Unsupervised domain adaptation (UDA) aims to transfer knowledge from a labeled source domain to an unlabeled target domain under a distribution shift (Mansour et al., 2009; Huang et al., 2006; Courty et al., 2017b; Ganin and Lempitsky, 2015; Long et al., 2015; Tzeng et al., 2017; Damodaran et al., 2018; Saito et al., 2018). Foundational theoretical work in UDA bounds the target risk using source risk and a measure of domain discrepancy, such as $\mathcal{H}$-divergence (Ben-David et al., 2010) or Wasserstein distance (Redko et al., 2017). While these frameworks provide valuable insights, they focus on one-shot adaptation and do not model the iterative nature of self-training or gradual shift across domains.

Gradual domain adaptation (GDA). GDA assumes access to a sequence of intermediate domains interpolating between source and target, enabling more stable adaptation than one-shot transfer. Early work by Gopalan et al. (2014) generated such intermediates via subspace interpolation, while He et al. (2024) proposed a self-training approach that traverses optimal transport (OT)-based barycenters between domains.

On the theoretical side, Kumar et al. (2020) provided the first generalization bound for gradual self-training, though it scales exponentially with the number of adaptation steps K. Wang et al. (2022) improved this by deriving a tighter bound with linear dependence on K, highlighting a trade-off between accumulated domain shift and statistical variance, and establishing the existence of an optimal number of steps. He et al. (2024) independently derived a similar trade-off using a stability-based analysis. However, none of these works isolate pseudo-labeling error or provide theoretical guarantees for filtering strategies commonly used in self-training. To the best of our knowledge, our work is the first to establish modular excess risk bounds for gradual self-training that explicitly track the per-round pseudo-labeling error ε_k and provide provable control via percentile-based thresholding.

Self-training and pseudo-label filtering. Modern self-training methods, such as FixMatch (Sohn et al., 2020) and FlexMatch (Zhang et al., 2021), use confidence-based filtering to reduce pseudo-label noise. While FixMatch applies a fixed global threshold, FlexMatch introduces curriculum pseudo-labeling with adaptive, class-wise thresholds that evolve during training. While effective empirically, these heuristics are typically not grounded in theory and can be sensitive to confidence thresholds or implicit curriculum dynamics, particularly under domain shift. Theoretical work on selective classification (El-Yaniv and Wiener, 2010; Jiang et al., 2018) and the Tsybakov margin condition (Tsybakov, 2004) offer partial tools, but do not address iterative adaptation or the accumulation of pseudo-labeling error across rounds. Our framework bridges this gap by formally connecting pseudo-labeling error to confidence calibration, margin decay, and thresholding schedules.

Intermediate domain generation via OT. When intermediate domains are not observed, several works generate them synthetically to enable gradual adaptation. Gopalan et al. (2014) proposed generating intermediate feature subspaces, while GOAT (He et al., 2024) uses OT-based barycenters to interpolate between distributions. Related OT-based methods (Courty et al., 2017a; Perrot et al., 2016) estimate transport plans or mappings to align source and target domains. Our experiments adopt a similar OT-based interpolation strategy, but apply it within a filtering-aware self-training framework, grounded in generalization bounds that account for pseudo-labeling error.

Compared to these prior works, our contribution lies

in providing a unified theoretical framework that (i) tracks and controls pseudo-labeling error under gradual shift, and (ii) justifies confidence- and margin-based filtering strategies that are used in practice but previously untheorized.

3 SETUP AND NOTATION

We study pseudo-labeling in the context of GDA, where a model is trained across a sequence of domains exhibiting incremental distributional shift. Let $\mathcal{X} \subseteq \mathbb{R}^d$ denote the input space and $[C] := \{1, \dots, C\}$ the set of class labels. A classifier is a function $h : \mathcal{X} \to \Delta_C$, where Δ_C is the probability simplex over C classes. Each $h(x)$ is a vector of soft class probabilities, and the predicted label is

$$\hat{y}(x) := \arg\max_{c \in [C]} h_c(x).$$

We seek to learn a high-accuracy classifier on the target domain μ_K using labeled data only from the source domain μ_0. To bridge the distribution gap we adopt GDA, that is, the learner proceeds though a sequence of intermediate domains $\mu_0,\ \mu_1,\ \dots,\ \mu_K$, each representing a small shift. At round k ($1 \le k \le K$), it receives an unlabeled batch $S_k = \{x_i\}_{i=1}^{M_k} \sim \mu_k$, annotates each x_i with the prediction of the previous model h_{k-1}, and filters the pseudo-labels by a confidence or margin threshold θ_k. Accepted points $\tilde{S}_k$ form the training set for the next classifier h_k, which is obtained by regularized empirical risk minimization (ERM) on $\tilde{S}_k$. The threshold schedule (Sec. 5.3) controls the accepted coverage ρ_k and, through standard calibration or margin-decay conditions, the quality of the pseudo-labels on that set.

We assume a bounded classification loss $\ell(h(x), y) \in [0, 1]$, which is L-Lipschitz in x for fixed h.

Definition 1 (p-Wasserstein distance)**.** *Let μ and ν be two probability measures over the space $\mathcal{X}$ and let $\mathsf{d} : \mathcal{X} \times \mathcal{X} \to \mathbb{R}_{\ge 0}$ be a metric. The* p-Wasserstein distance *between μ and ν is defined as*

$$W_p(\mu, \nu) := \left(\inf_{\pi \in \Pi(\mu,\nu)} \int_{\mathcal{X}\times\mathcal{X}} \mathsf{d}(x, x')^p \, d\pi(x, x') \right)^{1/p},$$

where $\Pi(\mu, \nu)$ denotes the set of all joint distributions over $\mathcal{X} \times \mathcal{X}$ whose marginals are μ and ν.

Definition 2 (Confidence and margin)**.** *Let $f : \mathcal{X} \to \mathbb{R}^C$ be a real-valued function that assigns a score to each class. In the case of softmax-based classifiers, $h(x) = \mathrm{softmax}(f(x))$, where $h(x) \in \Delta_C$ denotes the vector of predicted class probabilities. The confidence of prediction $h(x)$ is defined as*

$$c(x) := \max_{c \in [C]} h_c(x), \tag{1}$$

and the margin of prediction $f(x) \in \mathbb{R}^C$ is

$$\mathrm{margin}(x) := f_{\hat{y}(x)}(x) - \max_{j \ne \hat{y}(x)} f_j(x). \tag{2}$$

Filtering rule and acceptance. At round k, a sample x is accepted for pseudo-labeling according to the acceptance indicator

$$\mathsf{A}_k(x) := \mathbf{1}\{x \text{ is retained}\}, \qquad \rho_k := \mathbb{E}_{x\sim\mu_k}[\mathsf{A}_k(x)],$$

where ρ_k denotes the coverage, i.e., the expected fraction of accepted samples. We consider two instances:

1. **Confidence filtering:** $\mathsf{A}_k(x) = \mathbf{1}\{c_{k-1}(x) \ge \tau_k\}$.
2. **Margin filtering:** $\mathsf{A}_k(x) = \mathbf{1}\{\mathrm{margin}_{k-1}(x) \ge m_k\}$.

We use $\tilde{y}_k(x)$ to denote the pseudo-label under the chosen filtering strategy. We assign the top class of h_{k-1}, i.e.,

$$\tilde{y}_k(x) := \arg\max_c h_{k-1,c}(x), \tag{3}$$

and keep it only when $\mathsf{A}_k(x) = 1$. The corresponding retained pseudo-labeled dataset at round k is

$$\tilde{S}_k := \{(x, \tilde{y}_k(x)) :\ \mathsf{A}_k(x) = 1\}, \tag{4}$$

where $n_k := |\tilde{S}_k|$ denote its size and define $n_{\min} := \min_k n_k$.

We now formalize the notions of risk and pseudo-risk used in our analysis.

Definition 3 (Risks)**.** *The following risk quantities are used throughout the paper.*

True risk.

$$\mathcal{E}_{\mu_k}(h) := \mathbb{E}_{(x,y)\sim\mu_k}\big[\ell(h(x), y)\big]. \tag{5}$$

Expected pseudo-risk (under filtering). *Let $\mathsf{A}_k(x)$ be the acceptance indicator at round k and $\tilde{y}_k(x)$ the pseudo-label from h_{k-1}. Define*

$$\widehat{\mathcal{E}}_k(h) := \mathbb{E}_{x\sim\mu_k}\big[\mathsf{A}_k(x)\,\ell\big(h(x), \tilde{y}_k(x)\big)\big]. \tag{6}$$

Masked (accepted-set) true risk.

$$\overline{\mathcal{E}}_k(h) := \mathbb{E}_{(x,y)\sim\mu_k}\big[\mathsf{A}_k(x)\,\ell\big(h(x), y\big)\big]. \tag{7}$$

Empirical pseudo-risk. *For the retained set $\tilde{S}_k = \{(x_i, \tilde{y}_i)\}_{i=1}^{n_k}$,*

$$\widehat{R}_k(h) := \frac{1}{n_k} \sum_{(x_i,\tilde{y}_i)\in\tilde{S}_k} \ell\big(h(x_i), \tilde{y}_i\big). \tag{8}$$

We quantify pseudo-label noise *on the accepted set* (i.e., where $\mathsf{A}_k(x) = 1$) as the worst-case absolute deviation between the masked true risk $\overline{\mathcal{E}}_k$ and the masked pseudo-risk $\widehat{\mathcal{E}}_k$; loss on rejected points will be handled separately via the coverage term $1 - \rho_k$ in the main bound.

Definition 4 (Pseudo-labeling error)**.** *The pseudo-labeling error at round k is defined as*

$$\varepsilon_k \;:=\; \max_{h\in\{h_k,h_{k-1}\}} \big|\, \overline{\mathcal{E}}_k(h) - \widehat{\mathcal{E}}_k(h) \,\big|. \tag{9}$$

The maximization over $h \in \{h_k, h_{k-1}\}$ reflects that pseudo-labels are generated by h_{k-1} while generalization at round k concerns h_k; taking the max yields a model-agnostic bound that uniformly controls either role.

Training and complexity. Each classifier h_k is trained by minimizing the regularized empirical pseudo-risk:

$$h_k := \arg\min_{h\in\mathcal{H}} \Big\{ \widehat{R}_k(h) + \lambda\|h\|^2 \Big\}. \tag{10}$$

Definition 5 (Rademacher complexity)**.** *Let $\mathcal{F} := \{x \mapsto \ell(h(x), \tilde{y}(x)) \mid h \in \mathcal{H}\}$. Given a sample $S = \{x_1, \ldots, x_n\}$, the empirical Rademacher complexity is*

$$\widehat{\mathfrak{R}}_S(\mathcal{F}) := \mathbb{E}_\sigma \left[\sup_{f\in\mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} \sigma_i f(x_i) \right], \tag{11}$$

where $\sigma_i \sim \mathrm{Uniform}\{-1, +1\}$ are i.i.d. Rademacher variables.

4 GENERALIZATION BOUND VIA PSEUDO-LABELING ERROR

We begin by presenting a generalization bound that explicitly characterizes the excess risk of the final classifier h_K in terms of pseudo-labeling error, domain shift, and estimation error over filtered pseudo-labeled data. This bound holds under confidence or margin-based filtering and does not rely on algorithmic stability assumptions. It provides a foundation for the subsequent sections, where we develop concrete mechanisms based on confidence calibration and margin noise conditions to control the pseudo-labeling error and justify adaptive thresholding strategies.

Theorem 1 (Generalization bound with pseudo-labeling error)**.** *Let $\mu_0, \mu_1, \ldots, \mu_K$ be a sequence of distributions over $\mathcal{X} \times \mathcal{Y}$, and let $h_0, h_1, \ldots, h_K$ be classifiers where each h_k is obtained by minimizing regularized empirical risk over a pseudo-labeled dataset $\tilde{S}_k = \{(x_i, \tilde{y}_k(x_i))\}_{i=1}^{n_k}$, with pseudo-labels generated by h_{k-1}.*

Let ℓ be a loss taking values in $[0, 1]$, and assume that for any $h \in \mathcal{H}$ and $y \in \mathcal{Y}$ the map $x \mapsto \ell(h(x), y)$ is L-Lipschitz in x, and assume that each classifier satisfies $\|h_k\| \le B$, with the norm used in the regularization term in (10). Further, assume the empirical Rademacher complexity satisfies $\widehat{\mathfrak{R}}_{\tilde{S}_k}(\ell \circ \mathcal{H}) \le \frac{C_r}{\sqrt{n_k}}$, where the composed function class $\ell \circ \mathcal{H} := \{x \mapsto \ell(h(x), \tilde{y}_k(x)) \mid h \in \mathcal{H}\}$ [1]. For each round k, let $\rho_k := \mathbb{E}_{x\sim\mu_k}[\mathsf{A}_k(x)]$ be the coverage and let $\overline{\mathcal{E}}_k$ and $\widehat{\mathcal{E}}_k$ be the masked true risk (7) and expected pseudo-risk (6), respectively. Define ε_k as in (9). Then, for any $\delta \in (0, 1)$, with probability at least $1 - \delta$ over the pseudo-labeled datasets $\tilde{S}_1, \ldots, \tilde{S}_K$, the excess risk satisfies

$$\begin{aligned} &\mathcal{E}_{\mu_K}(h_K) - \mathcal{E}_{\mu_0}(h_0) \\ &\le \sum_{k=1}^{K} \Big((1-\rho_k) + 2\varepsilon_k + \frac{C_\delta}{\sqrt{n_k}} + LW_1(\mu_k, \mu_{k-1}) + \lambda B^2 \Big), \end{aligned}$$

where

$$C_\delta := 4C_r + 6\sqrt{\frac{\log(2K/\delta)}{2}}.$$

Theorem 1 gives a fine-grained generalization bound for iterative self-training along a sequence of domains. Its key feature is that it isolates, at each round k, the accepted-set pseudo-label substitution error ε_k. The bound decomposes the change in risk into interpretable terms: the rejection rate (missing coverage) $(1 - \rho_k)$ and pseudo-label error (ε_k), a sample-complexity term controlled by empirical Rademacher complexity, a distribution-shift term measured by the Wasserstein-1 distance between successive domains, and a regularization term. This modular form lets calibration- or margin-based assumptions directly bound ε_k and informs the design of filtering schedules.

In contrast, He et al. (2024) develop a stability-based analysis of GDA whose bounds rely on smoothness and optimization-stability and effectively control cumulative domain shift. Their framework does not explicitly model pseudo-label substitution or the effect of confidence/margin filtering. Our analysis makes this error term explicit, enabling provable guidance for filtering schedules and pseudo-label quality.

Remark 1 (coverage)**.** *The term $1 - \rho_k$ upper bounds the loss on the unaccepted set at round k, and thus is the unavoidable price of selectivity. Under percentile schedules (see Appendix 5.3) with $\rho_k \ge 1 - c/k$, this penalty accumulates only logarithmically in K.*

[1] This assumption holds for many standard hypothesis classes with norm constraints (Bartlett and Mendelson, 2002; Mohri et al., 2018).

5 FILTERING-BASED CONTROL OF PSEUDO-LABELING ERROR

Recall from Theorem 1 that the total excess risk is expressed as a sum of round-wise contributions. Among these, two terms are directly affected by filtering at round k: the *rejection rate* (i.e., the fraction of inputs not accepted by the filter) and the *pseudo-label substitution error* on the accepted set. This section develops confidence- and margin-based strategies to control both quantities. We introduce round-dependent calibration and margin functions that bound the accepted-set substitution error, and we show how percentile or adaptive thresholds can be used to set coverage. Together, these tools yield a principled coverage–noise trade-off and, under mild schedules, logarithmic accumulation of their combined effect across rounds.

5.1 Confidence-Based Filtering

We now control the accepted-set pseudo-labeling error ε_k via confidence thresholding. At round k, pseudo-labels are produced by h_{k-1} and retained when $c_{k-1}(x) \geq \tau_k$, which induces coverage $\rho_k := \Pr_{x\sim\mu_k}[c_{k-1}(x) \geq \tau_k]$. This ties the round-wise decomposition from Theorem 1 to practical controls: the threshold sets coverage and, through calibration, controls substitution error on the accepted set.

Definition 6 (Confidence-conditioned error)**.** *For round k, define*

$$\phi_k(\tau) \;:=\; \Pr_{(x,y)\sim\mu_k}\left[\hat{y}_{k-1}(x) \neq y \,\middle|\, c_{k-1}(x) \geq \tau\right],\; \tau \in [0,1],$$

where $c_{k-1}(x) = \max_c h_{k-1,c}(x)$ and $\hat{y}_{k-1}(x) = \arg\max_c h_{k-1,c}(x)$.

The function $\phi_k(\tau)$ is the conditional error on the accepted set at threshold τ, aligning with the risk–coverage viewpoint in selective classification (El-Yaniv and Wiener, 2010; Jiang et al., 2018), where higher-confidence predictions typically incur lower conditional error.

Lemma 1 (Confidence control of accepted-set error)**.** *Let $\ell \in [0,1]$. For any threshold τ_k,*

$$\varepsilon_k \;\leq\; \rho_k\,\phi_k(\tau_k) \;\leq\; \phi_k(\tau_k).$$

Corollary 1 (Controlled decay under calibrated thresholds)**.** *If the thresholds $\tau_k \uparrow 1$ are chosen so that $\phi_k(\tau_k) \leq C_a/k$, for some constant C_a, then*

$$\sum_{k=1}^{K}\varepsilon_k \;\leq\; \sum_{k=1}^{K}\rho_k\,\phi_k(\tau_k) \;\leq\; C_a\,(1+\log K).$$

If, in addition, a percentile schedule enforces $1-\rho_k \leq c/k$, then

$$\sum_{k=1}^{K}\big((1-\rho_k)+2\varepsilon_k\big) \;\leq\; (c+2C_a)\,(1+\log K).$$

Remark 2. *These bounds make the coverage–noise trade-off explicit: raising τ_k typically reduces the conditional error $\phi_k(\tau_k)$ on accepted points but also decreases coverage ρ_k. Percentile (quantile) thresholds set coverage directly, while calibration improves the conditional error. Together, they enable schedules under which the combined contribution $\sum_k \big((1-\rho_k)+2\varepsilon_k\big)$ grows only logarithmically, tightening the overall generalization bound (See Appendix 5.3).*

5.2 Margin-Based Filtering

When probabilistic confidence is poorly calibrated, it is often preferable to filter by the *logit margin*, which measures the separation between the top predicted class and its nearest competitor. At round k, we retain a pseudo-label from h_{k-1} only when the previous-round margin exceeds a threshold m_k, i.e., when $\mathrm{margin}_{k-1}(x) \geq m_k$, inducing coverage $\rho_k := \Pr_{x\sim\mu_k}[\mathrm{margin}_{k-1}(x) \geq m_k]$. This subsection develops accepted-set bounds for the pseudo-label substitution error under margin filtering, paralleling the confidence case.

Definition 7 (Round-dependent margin-conditioned error)**.** *For round k, define the margin-conditioned error curve of h_{k-1} on μ_k by*

$$\zeta_k(m) \;:=\; \Pr_{(x,y)\sim\mu_k}\left[\hat{y}_{k-1}(x) \neq y \,\middle|\, \mathrm{margin}_{k-1}(x) \geq m\right],$$

where $m \geq 0$, $\hat{y}_{k-1}(x) = \arg\max_c h_{k-1,c}(x)$, and $\mathrm{margin}_{k-1}(x) = f_{k-1,\hat{y}_{k-1}(x)}(x) - \max_{j\neq\hat{y}_{k-1}(x)} f_{k-1,j}(x)$ as in (2).

Lemma 2 (Margin control of accepted-set error)**.** *Let $\ell \in [0,1]$. For any threshold $m_k \geq 0$,*

$$\varepsilon_k \;\leq\; \rho_k\,\zeta_k(m_k) \;\leq\; \zeta_k(m_k).$$

Corollary 2 (Controlled decay under margin thresholds)**.** *If the thresholds m_k are chosen so that $\zeta_k(m_k) \leq C_m/k$, for some constant C_m, then*

$$\sum_{k=1}^{K}\varepsilon_k \;\leq\; \sum_{k=1}^{K}\rho_k\,\zeta_k(m_k) \;\leq\; C_m\,(1+\log K).$$

If, in addition, a percentile rule on the margin enforces $1-\rho_k \leq c/k$, then

$$\sum_{k=1}^{K}\big((1-\rho_k)+2\varepsilon_k\big) \;\leq\; (c+2C_m)\,(1+\log K).$$

Remark 3 (Relation to Tsybakov conditions)**.** *The curve $\zeta_k(m)$ quantifies the conditional error among high-margin examples. Under smoothness and class-separation assumptions, one typically has $\zeta_k(m) \to 0$ as $m \to \infty$, and often a power-law decay*

$$\zeta_k(m) \;\le\; C\, m^{-\alpha} \quad (\alpha > 0),$$

analogous to rates derived from the Tsybakov noise condition in statistical learning (Tsybakov, 2004; Bartlett et al., 2006; Koltchinskii and Panchenko, 2002; Mohri et al., 2018). In binary classification, the Tsybakov condition bounds the mass near the decision boundary, $\Pr(|\eta(x) - \frac{1}{2}| \le t) \le Ct^{\alpha}$, where $\eta(x) := \mathbb{P}(y = 1 \mid x)$ denotes the Bayes posterior in the binary setting, and multiclass analogues based on top-two margin yield similar decay. Such decay implies schedules with $\zeta_k(m_k) = O(1/k)$, e.g., $m_k \propto k^{1/\alpha}$.

Margin-based filtering requires no probabilistic and directly reflects geometric separation: increasing m_k typically lowers the accepted-set conditional error $\zeta_k(m_k)$ but also reduces coverage ρ_k. Together with percentile or adaptive thresholding on margins, Lemma 2 and Corollary 2 provide practical controls for the two filtering-dependent terms in the round-wise decomposition, yielding logarithmic accumulation under mild schedules.

5.3 Adaptive Thresholding and Algorithm Design

Building on the analysis of Sections 5.1–5.2, we propose a *percentile* thresholding strategy that sets coverage directly from unlabeled data. Under a mild score-tail risk–coverage assumption (Assumption 1), the accepted-set conditional error is controlled so that the round-wise contributions accumulate at most logarithmically in K. Each round, the filter controls the rejection rate $(1-\rho_k)$ and the accepted-set substitution error ε_k via a confidence or margin threshold; the percentile schedule fixes ρ_k, while the assumption translates a small rejected tail into a small accepted-set error.

Let $\mathrm{score}_{k-1}(x) \in \{c_{k-1}(x), \mathrm{margin}_{k-1}(x)\}$, and define

$$\rho_k(\theta) := \Pr_{x\sim\mu_k}\big[\mathrm{score}_{k-1}(x) \ge \theta\big],$$
$$\psi_k(\theta) := \Pr_{(x,y)\sim\mu_k}\big[\hat{y}_{k-1}(x) \ne y \;\big|\; \mathrm{score}_{k-1}(x) \ge \theta\big].$$

Given a target coverage $q_k \in (0,1)$, choose θ_k as the empirical $(1-q_k)$-quantile and retain

$$\tilde{S}_k = \{(x, \tilde{y}_k(x)) :\, \mathrm{score}_{k-1}(x) \ge \theta_k\},\;\; \tilde{y}_k(x) = \hat{y}_{k-1}(x).$$

Percentile schedules (coverage from unlabeled data). We use a nondecreasing schedule $q_k \ge 1 - \frac{c}{k}$, which enforces (at the population level) a shrinking rejected tail $1 - \rho_k(\theta_k) \le \frac{c}{k}$.

Lemma 3 (Quantile accuracy)**.** *Let M_k unlabeled scores be drawn i.i.d. from μ_k, and let θ_k be the empirical $(1-q_k)$-quantile. Then, with probability at least $1-\delta$ uniformly over $k = 1, \dots, K$,*

$$\big|\rho_k(\theta_k) - q_k\big| \;\le\; \sqrt{\frac{2\,\log(2K/\delta)}{M_k}}. \tag{12}$$

By Lemmas 1 and 2,

$$\varepsilon_k \;\le\; \rho_k(\theta_k)\,\psi_k(\theta_k) \;\le\; \psi_k(\theta_k).$$

The percentile control translates into decay of $\psi_k(\theta_k)$ under the following assumption.

Assumption 1 (Score-tail error decay)**.** *There exist constants $C_t, \alpha > 0$ and θ_0 such that for all $\theta \ge \theta_0$,*

$$\psi_k(\theta) \;\le\; C_t\,\big(1 - \rho_k(\theta)\big)^{\alpha}.$$

According to Assumption 1 – which is satisfied, for instance, when the score tail obeys a (C_t, α) power-law decay (cf. Tsybakov-type noise conditions) – whenever the rejected tail $1 - \rho_k(\theta)$ is small (i.e., retention is high), the accepted-set conditional error $\psi_k(\theta)$ is controlled and scales polynomially with the tail size (see Remark 3).

Proposition 1 (Pseudo-label error under percentile control)**.** *Let $q_k \ge 1 - \frac{c}{k}$ and choose θ_k as above. If Assumption 1 holds and $\theta_k \ge \theta_0$, then with probability at least $1-\delta$,*

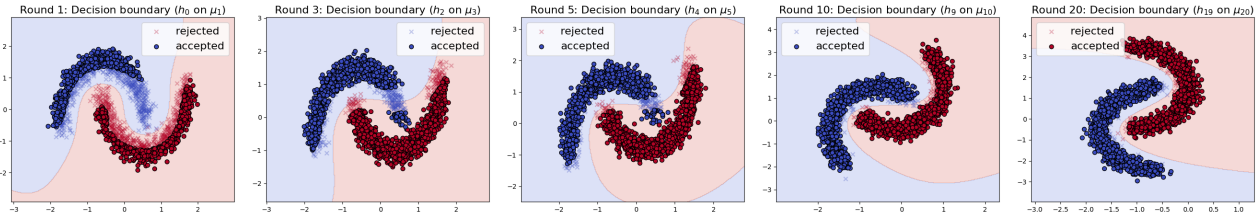
$$\varepsilon_k \;\le\; \rho_k(\theta_k)\,\psi_k(\theta_k) \;\le\; C_t\Big(\tfrac{c}{k} + \sqrt{\tfrac{2\log(2K/\delta)}{M_k}}\Big)^{\alpha}. \tag{13}$$

Hence, if $M_k = \omega\big(k^2 \log(K/\delta)\big)$ so $\sqrt{\log(2K/\delta)/M_k} = o(1/k)$, then $\psi_k(\theta_k), \varepsilon_k = O(k^{-\alpha})$ and $\sum_{k=1}^{K} \varepsilon_k = O(\log K)$ when $\alpha = 1$ (and $O(1)$ when $\alpha > 1$).

Therefore, percentile thresholding fixes the rejection rate from unlabeled data, and, under Assumption 1, this coverage control implies the decay of the accepted-set conditional error and thus of ε_k, yielding the $O(\log K)$ accumulation predicted by the round-wise bound.

5.4 Methodology

We describe our filtered self-training procedure over a gradually shifting sequence of domains $\mu_0, \mu_1, \dots, \mu_K$ (Sec. 5.1–5.2). The design follows the accepted-set viewpoint in which each round sets a threshold to realize target coverage and forms a pseudo-labeled set on which h_k is trained.



$1-\rho_k = 0.5$ $1-\rho_k = 0.17$ $1-\rho_k = 0.1$ $1-\rho_k = 0.05$ $1-\rho_k = 0.02$

Figure 1: Step-by-step visualization of pseudo-label filtering across selected training rounds (Rounds 1, 3, 5, 10, 20). Each panel shows the desicion boundary of h_{k-1} on μ_k, with the rejection rate $1-\rho_k$ shown below.

Filtered self-training over intermediate domains. At round k, model h_{k-1} produces scores on S_k – either confidence $c_{k-1}(x)$ or margin $\text{margin}_{k-1}(x)$. A threshold θ_k induces acceptance $\mathsf{A}_k(x) = \mathbf{1}\{\text{score}_{k-1}(x) \geq \theta_k\}$ with coverage $\rho_k = \mathbb{E}[\mathsf{A}_k(x)]$. We use a *percentile* schedule to target coverage q_k (e.g., $q_k = 1 - \frac{c}{k}$), setting θ_k to the empirical $(1-q_k)$-quantile of scores on S_k. Accepted points receive pseudo-labels $\tilde{y}_k(x) = \hat{y}_{k-1}(x)$ to form $\tilde{S}_k$, and h_k is trained by regularized ERM on $\tilde{S}_k$. For details, see Algorithms 1 and 2 in Appendix D.

6 EXPERIMENTS

Our experiments evaluate confidence-guided pseudo-label filtering in GDA. We compare our Confidence-Filtered Self-Training (CFSTDA) and Margin-Filtered Self-Training (MFSTDA) against (i) Gradual Self-Training (GST) (Kumar et al., 2020), which self-trains only along the given source–target trajectory, and (ii) GOAT (He et al., 2024), which augments this sequence with intermediate domains via Wasserstein barycentric interpolation (details in Appendix E). We also include one-shot unsupervised DA (UDA) baselines—DANN (Ganin et al., 2016) and DeepCoral (Sun and Saenko, 2016)—that operate only on the source and target and cannot leverage intermediate unlabeled data. We evaluate on synthetic datasets, four real-world gradual-shift benchmarks, and an additional standard UDA benchmark to assess performance beyond native GDA settings.

6.1 Synthetic Datasets

We first visualize the pseudo-labeling strategies on the Two Moons dataset (Pedregosa et al., 2011) with gradual degree shifts. The data undergoes a gradual 90° rotation from source to target over 20 steps, with 1000 samples drawn per domain. At round k, we admit the top q_k fraction of an unlabeled batch according to the chosen score, following the schedule $q_k = 1 - \frac{c}{k}$, $c = 0.5$. Figure 1 shows the CFSTDA trajectory at rounds $k \in \{1, 3, 5, 10, 20\}$. Each panel displays the decision boundary of h_{k-1} evaluated on the current domain μ_k, with filled points indicating the accepted pseudo-labels and crosses indicating the rejected ones. The rejection rate $1-\rho_k$ is shown below each plot. As training progresses, the coverage ρ_k increases and more points are accepted as the model improves. This behavior supports our theoretical premise: pseudo-label quality can be controlled via percentile filtering, gradually expanding the trusted region as the learner adapts. An additional example using a 2D Gaussian mixture distribution is provided in Appendix G. Our code is available online at `https://github.com/aheidarizadeh/CFST-GDA`.

Empirical validation. We simulate a binary task under gradual domain shift for 45 rounds using the Two Moons distribution. At each round k we rotate the source by 2° to obtain μ_k (total 90° from source to target), draw an unlabeled batch $S_k \sim \mu_k$, and update h_k by self–training on its pseudo-labels. We then retain the top $q_k = 1 - \frac{c}{k}$ fraction by confidence (CFSTDA) or by margin (MFSTDA), and record coverage ρ_k, accepted-set loss ψ_k, and substitution error ε_k. Figure 2 (left) plots the conditional errors ψ_k and pseudo-labeling error ε_k. The curves remain bounded and nearly stable across rounds, which is the behavior predicted by Proposition 1. Across all runs we observe the bound $\varepsilon_k \leq \psi_k$, consistent with Lemmas 1 and 2. We regress $\log \psi_k$ on $\log(1-\rho_k)$ across rounds k and take the ordinary-least-squares (OLS) slope as $\hat{\alpha}$. On Two Moons, the log–log relation is approximately linear with positive slope $\hat{\alpha} \approx 2.39$ (*center-right* panel), and the envelope $\psi_k/(1-\rho_k)^{\hat{\alpha}}$ remains bounded over k (*center-left* panel), providing direct evidence that Assumption 1 holds in this setting. The *right* panel shows the coverage curves ρ_k, demonstrating that the empirical acceptance closely follows the schedule q_k, so that the rejection rate $1-\rho_k$ decreases with k.

Ablation on quantile scheduling. To evaluate the robustness of our percentile-thresholding design, we vary the constant c in the quantile schedule $q_k = 1 - \frac{c}{k}$. Figure 3 reports the resulting pseudo-labeling error ε_k and coverage ρ_k across different $c \in \{0.0, 0.1, 0.25, 0.5, 0.75, 1.0\}$. For MFSTDA, larger c values consistently reduce the pseudo-labeling

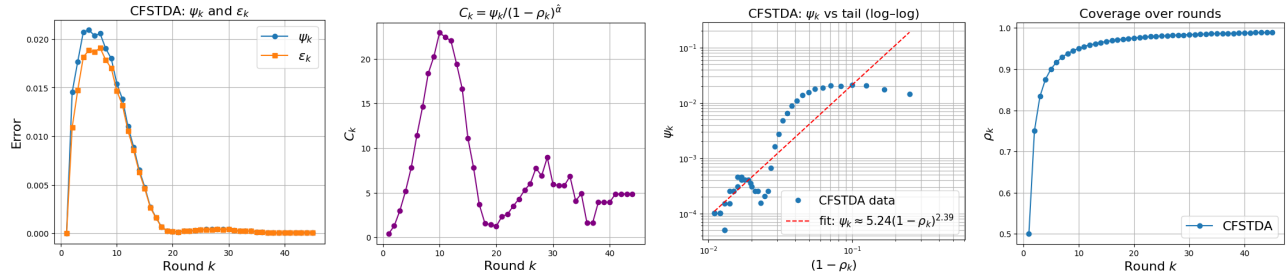


Figure 2: Empirical validation of pseudo-labeling error bounds under gradual domain adaptation. **Left:** Conditional error ψ_k and pseudo-labeling error ε_k. **Center-left:** Ratio $\psi_k/(1-\rho_k)^{\hat{\alpha}}$ (proportional to the scaled errors $k^{\hat{\alpha}}\psi_k$ and $k^{\hat{\alpha}}\varepsilon_k$). **Center-right:** Log–log plot of ψ_k against $(1-\rho_k)$ with fitted slope $\hat{\alpha} \approx 2.39$, **Right:** Coverage ρ_k over rounds, indicating the fraction of pseudo-labeled points retained after filtering.

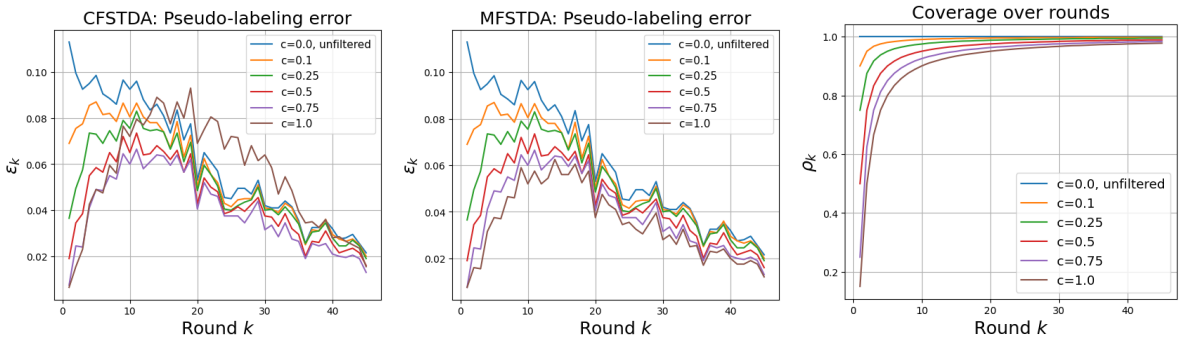


Figure 3: Ablation on percentile parameter c in the quantile schedule $q_k = 1 - \frac{c}{k}$. Coverage ρ_k (right) and Pseudo-labeling error ε_k are plotted for both CFSTDA (left) and MFSTDA (center), with varying $c \in \{0.0, 0.1, 0.25, 0.5, 0.75, 1.0\}$.

error, indicating more stable adaptation when more pseudo-labeled samples are retained. CFSTDA exhibits a non-monotonic pattern: very large c values initially produce higher error due to low-confidence early pseudo-labels but improve steadily over rounds as the model stabilizes. Overall, moderate to large c values provide the best trade-off between early precision and long-term accuracy.

6.2 Real Datasets

We conduct experiments on four popular gradual shift benchmarks: Rotated MNIST (LeCun et al., 1998), Color-Shift MNIST (He et al., 2024), Portraits (Ginosar et al., 2015), and Cover Type (Blackard and Dean, 1999). These benchmarks provide either natural or standard gradual source-to-target trajectories and therefore directly match the GDA setting studied in our theory. We also include additional experiments on Office-Home (Venkateswara et al., 2017), a standard UDA benchmark that does not provide a native gradual trajectory. A full description of the datasets and the network architectures is given in Appendix F.

Training protocol with filtering. At each self-training round k, we compute confidence or margin scores for all unlabeled samples and retain only the top q_k fraction per class based on these scores. To preserve class diversity, a minimum of 10% of samples per *predicted* class is always retained. Both CFSTDA and MFSTDA use a gradually increasing schedule with $q_k = 1 - \frac{1}{k}$, accepting more data over time. This implementation aligns with our theoretical schedule and enables dynamic control of pseudo-label quality. The complete training configuration and hyperparameter choices are provided in Appendix F. We further analyze the effect of class imbalance in Appendix G.

Comparison with GDA methods. We evaluate our CFSTDA and MFSTDA on the *target* domain and compare against four baselines: (i) Baseline (no adaptation) – a classifier trained only on the source; (ii) UDA methods DANN (Ganin et al., 2016) and Deep-CORAL (Sun and Saenko, 2016); (iii) GST; and (iv) GOAT. A "given" domain is a true intermediate domain provided by the dataset's gradual-shift sequence; a "gen" domain is an additional intermediate domain generated by Wasserstein barycentric interpolation between two consecutive given domains. We sweep the grid given $\in \{0, 1, 2, 3\} \times$ gen $\in \{0, 1, 2\}$; see Appendix F for full details. Table 1 reports target accuracy averaged over five runs (95% CIs) for the representative setting given $= 2$, gen $= 2$. MFSTDA is top on every benchmark–except Rotated MNIST, where CFSTDA slightly outperforms it–with CFSTDA otherwise consistently second. For instance, on Color-Shift MNIST,

Table 1: Comparison of direct UDA and GDA methods.

Method	Rot. MNIST	Color-Shift	Portraits	Cover Type
Baseline (no adaptation)	44.3±1.7	35.5±11.1	75.6±1.2	61.2±4.9
DANN (Ganin et al., 2016)	48.0±6.4	37.88±22.6	76.8±3.2	66.0±2.6
DeepCoral (Sun and Saenko, 2016)	51.6±3.1	50.2±24.9	74.7±0.8	65.9±2.4
GST (Kumar et al., 2020)(2 given)	59.3±5.2	53.2±14.0	76.3 ±1.7	67.8±4.6
GOAT (He et al., 2024)(2 given, 2 gen)	64.3±3.5	82.8±11.6	81.1±2.6	70.4±1.7
CFSTDA (2 given, 2 gen)	**67.7±5.4**	88.8±5.5	79.8±3.1	71.7±2.9
MFSTDA (2 given, 2 gen)	65.5±4.7	**91.1±3.4**	**84.2±0.6**	**72.5±2.3**

Table 2: Evaluation on Office-Home under direct and generated gradual adaptation settings.

Method	Office-Home Real → Product				Office-Home Art → Product			
Baseline (no adaptation)	74.0±0.7				62.8±1.2			
DANN (Ganin et al., 2016)	73.8±0.7				62.9±0.9			
DeepCoral (Sun and Saenko, 2016)	74.0±0.4				63.2±1.1			
	0 gen	1 gen	2 gen	3 gen	0 gen	1 gen	2 gen	3 gen
GOAT (He et al., 2024)	74.2±1.0	74.7±1.0	73.1±0.6	72.2±0.4	63.5±0.9	62.8±0.9	61.0±1.3	57.9±1.2
CFSTDA	74.7±0.5	76.3±0.8	75.8±0.9	75.5±0.4	65.5±0.8	**67.6±1.2**	66.3±1.0	64.2±1.7
MFSTDA	75.0±0.8	**76.8±1.4**	76.1±1.7	75.8±0.7	65.9±1.0	67.2±1.2	67.0±1.2	64.1±1.1

MFSTDA reaches 91.1%, compared to 88.8% for CFSTDA and 82.8% for GOAT–a 8.3-point boost over the strongest non-filtered baseline. The results also clearly demonstrate the advantage of GDA methods over traditional UDA approaches. Detailed accuracy results for all combinations of observed ("given") and OT-generated ("gen") intermediate domains are reported in Appendix G, where we also provide pseudo-labeling error and coverage over rounds to illustrate their behavior. Across these settings, both CFSTDA and MFSTDA consistently outperform GST and GOAT, confirming that filtered self-training provides large and reliable gains under gradual domain shift.

Additional evaluation on standard UDA benchmarks. To evaluate beyond native gradual-shift datasets, we also consider Office-Home (Venkateswara et al., 2017), which does not provide an inherent ordered sequence of intermediate domains. We therefore report results in two settings. First, we study the standard direct adaptation setting ($K = 1$), which enables fair comparison to conventional UDA baselines on the source→target task (0 generated intermediate domains). Second, we construct gradual trajectories by inserting OT-generated intermediate domains between the source and target, specifically, $1, 2$, and 3 generated domains. This allows us to test whether the proposed filtering mechanism remains beneficial when gradual structure is induced rather than given by the dataset. Table 2 shows that CFSTDA and MFSTDA remain competitive in the direct setting and improve further when a small number of intermediate domains is introduced; most notably, MFSTDA achieves 76.8% on Real→Product and CFSTDA achieves 67.6% on Art→Product, both at 1 generated domain. These results support our claim that the proposed filtering rule and percentile schedule are not tied to a specific native GDA benchmark, but provide a complementary mechanism that continues to improve self-training on standard domain adaptation datasets as well.

7 CONCLUSION

We presented and analyzed a filtered self-training framework for GDA that treats each round through an *accepted-set* lens. Our decomposition isolates two controllable terms–the coverage penalty $(1 - \rho_k)$ and the accepted-set pseudo-label error (ε_k)–from shift, sampling and regularization effects. Percentile schedules fix coverage directly from unlabeled scores, while calibration or margin-decay (Tsybakov-type) assumptions bound ε_k, so their joint contribution grows only $O(\log K)$. Experiments with both observed and OT-generated intermediates match these predictions and give concrete guidelines for threshold design.

Limitations and directions. Theorem 1 does not rely on the score-tail decay in Assumption 1; however, when that decay fails (e.g., under very large shifts), our percentile schedule is no longer guaranteed to deliver the $O(\log K)$ accumulation. Two extensions could restore tight control: (i) *Adaptive thresholds*: use a small set of labeled probes each round to adjust the threshold until the observed accepted-set error meets a target level, thereby bounding ε_k directly; (ii) *Localized complexity*: replace the global sampling term with complexity measures restricted to the accepted set (e.g., local Rademacher quantities), which remain small even when coverage is low.

Acknowledgements

This work was supported by DARPA under Agreement No. HR0011-24-9-0427 and NSF under Awards CCF-2106339 and DMS-2304489.

References

Peter L. Bartlett and Shahar Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3:463–482, 2002.

Peter L Bartlett, Michael I Jordan, and Jon D McAuliffe. Convexity, classification, and risk bounds. *Journal of the American Statistical Association*, 101(473):138–156, 2006.

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine Learning*, 79(1):151–175, 2010. doi: 10.1007/s10994-009-5152-4.

Jock A Blackard and Denis J Dean. Comparative accuracies of artificial neural networks and discriminant analysis in predicting forest cover types from cartographic variables. *Computers and electronics in agriculture*, 24(3):131–151, 1999.

Nicolas Courty, Rémi Flamary, Amaury Habrard, and Alain Rakotomamonjy. Joint distribution optimal transportation for domain adaptation. In *Proceedings of the 31st International Conference on Neural Information Processing Systems (NeurIPS)*, pages 3733–3742, Red Hook, NY, USA, 2017a. Curran Associates Inc. ISBN 9781510860964.

Nicolas Courty, Rémi Flamary, Devis Tuia, and Alain Rakotomamonjy. Optimal transport for domain adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(9):1853–1865, 2017b.

Bharath Bhushan Damodaran, Benjamin Kellenberger, Rémi Flamary, Devis Tuia, and Nicolas Courty. "deepjdot": Deep joint distribution optimal transport for unsupervised domain adaptation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 447–463, 2018.

Ran El-Yaniv and Yair Wiener. On the foundations of noise-free selective classification. *Journal of Machine Learning Research (JMLR)*, 11:1605–1641, 2010.

Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, page 1180–1189. JMLR.org, 2015.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario March, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of machine learning research*, 17(59):1–35, 2016.

Shiry Ginosar, Kate Rakelly, Sarah Sachs, Brian Yin, and Alexei A Efros. A century of portraits: A visual historical record of american high school yearbooks. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 1–7, 2015.

Raghuraman Gopalan, Ruonan Li, and Rama Chellappa. Unsupervised adaptation across domain shifts by generating intermediate data representations. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 36(11):2288–2302, 2014.

Rui He, Chao Wang, Jiangchao Li, and Boqing Gong. Gradual domain adaptation: Theory and algorithms. *Journal of Machine Learning Research*, 25: 1–40, 2024.

Jiayuan Huang, Alexander J. Smola, Arthur Gretton, Karsten M. Borgwardt, and Bernhard Scholkopf. Correcting sample selection bias by unlabeled data. In *Proceedings of the 20th International Conference on Neural Information Processing Systems*, page 601–608, Cambridge, MA, USA, 2006. MIT Press.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015.

Heinrich Jiang, Been Kim, Melody Y Guan, and Maya Gupta. To trust or not to trust a classifier. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 31, 2018.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

Vladimir Koltchinskii and Dmitriy Panchenko. Empirical margin distributions and bounding the generalization error of combined classifiers. *Annals of Statistics*, 30(1):1–50, 2002.

Ananya Kumar, Tengyu Ma, and Percy Liang. Understanding self-training for gradual domain adaptation. In *International conference on machine learning (ICML)*, pages 5468–5479. PMLR, 2020.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86 (11):2278–2324, 1998.

Mingsheng Long, Yue Cao, Jianmin Wang, and Michael Jordan. Learning transferable features with deep adaptation networks. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 97–105, Lille, France, 07–09 Jul 2015. PMLR.

Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. Domain adaptation: Learning bounds and algorithms. In *The 22nd Conference on Learning Theory (COLT), Montreal, Canada*, 2009.

Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of Machine Learning*. MIT Press, 2 edition, 2018.

Adam Paszke, Sam Gross, Francisco Massa, et al. PyTorch: An imperative style, high-performance deep learning library. *NeurIPS*, 2019.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. Scikit-learn: Machine learning in python. *Journal of machine learning research*, 12(Oct):2825–2830, 2011.

Michaël Perrot, Nicolas Courty, Rémi Flamary, and Amaury Habrard. Mapping estimation for discrete optimal transport. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems (NeurIPS)*, volume 29. Curran Associates, Inc., 2016.

Ievgen Redko, Amaury Habrard, and Marc Sebban. Theoretical analysis of domain adaptation with optimal transport. In *Machine Learning and Knowledge Discovery in Databases*, pages 737–753. Springer International Publishing, 2017.

Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada. Maximum classifier discrepancy for unsupervised domain adaptation. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3723–3732, 2018. doi: 10.1109/CVPR.2018.00392.

Kihyuk Sohn, David Berthelot, Chun-Liang Li, Zizhao Zhang, Nicholas Carlini, Ekin D. Cubuk, Alex Kurakin, Han Zhang, and Colin Raffel. Fixmatch: simplifying semi-supervised learning with consistency and confidence. In *Proceedings of the 34th International Conference on Neural Information Processing Systems (NeurIPS)*, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, et al. Dropout: A simple way to prevent neural networks from overfitting. *JMLR*, 15(56):1929–1958, 2014.

Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *European conference on computer vision*, pages 443–450. Springer, 2016.

Alexander B Tsybakov. Optimal aggregation of classifiers in statistical learning. *Annals of Statistics*, 32 (1):135–166, 2004.

Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2962–2971, 2017. doi: 10.1109/CVPR.2017.316.

Aad W. van der Vaart and Jon A. Wellner. *Weak Convergence and Empirical Processes: With Applications to Statistics*. Springer, 1996.

Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5018–5027, 2017.

Cédric Villani. *Optimal Transport: Old and New*, volume 338. Springer, 2008.

Haoxiang Wang, Bo Li, and Han Zhao. Understanding gradual domain adaptation: Improved analysis, optimal path and beyond. *Proceedings of Machine Learning Research*, 162:22784–22801, 2022. ISSN 2640-3498.

Bowen Zhang, Yidong Wang, Wenxin Hou, Hao Wu, Jindong Wang, Manabu Okumura, and Takahiro Shinozaki. Flexmatch: boosting semi-supervised learning with curriculum pseudo labeling. In *Proceedings of the 35th International Conference on Neural Information Processing Systems (NeurIPS)*, Red Hook, NY, USA, 2021. Curran Associates Inc. ISBN 9781713845393.

Yang Zou, Zhiding Yu, B. V. K. Vijaya Kumar, and Jinsong Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *ECCV: 15th European Conference, Proceedings, Part III*, pages 297–313, Berlin, Heidelberg, 2018. Springer-Verlag.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [No]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes]

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes]
   (b) Complete proofs of all theoretical results. [Yes]
   (c) Clear explanations of any assumptions. [Yes]

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes]
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes]
   (b) The license information of the assets, if applicable. [Not Applicable]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]
   (d) Information about consent from data providers/curators. [Not Applicable]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

Supplementary Materials for Confidence-Guided Self-Training for Gradual Domain Adaptation

A Proof of Generalization Bound

Recall Theorem 1: Let $\mu_0, \mu_1, \ldots, \mu_K$ be a sequence of distributions over $\mathcal{X} \times \mathcal{Y}$, and let $h_0, h_1, \ldots, h_K$ be classifiers where each h_k is obtained by minimizing regularized empirical risk over a pseudo-labeled dataset $\tilde{S}_k = \{(x_i, \tilde{y}_k(x_i))\}_{i=1}^{n_k}$, with pseudo-labels generated by h_{k-1}. Further, assume the empirical Rademacher complexity satisfies

$$\widehat{\mathfrak{R}}_k := \widehat{\mathfrak{R}}_{\tilde{S}_k}(\ell \circ \mathcal{H}) \leq \frac{C_r}{\sqrt{n_k}}.$$

Then, for any $\delta \in (0,1)$, with probability at least $1-\delta$ over the pseudo-labeled datasets $\tilde{S}_1, \ldots, \tilde{S}_K$, the excess risk satisfies

$$\mathcal{E}_{\mu_K}(h_K) - \mathcal{E}_{\mu_0}(h_0) \leq \sum_{k=1}^{K} \left((1-\rho_k) + 2\varepsilon_k + \frac{C_\delta}{\sqrt{n_k}} + LW_1(\mu_k, \mu_{k-1}) + \lambda B^2 \right),$$

where

$$C_\delta := 4C_r + 6\sqrt{\frac{\log(2K/\delta)}{2}}.$$

Proof. We use a telescoping decomposition:

$$\mathcal{E}_{\mu_K}(h_K) - \mathcal{E}_{\mu_0}(h_0) = \sum_{k=1}^{K} \left(\mathcal{E}_{\mu_k}(h_k) - \mathcal{E}_{\mu_{k-1}}(h_{k-1}) \right).$$

Each term is split as

$$\mathcal{E}_{\mu_k}(h_k) - \mathcal{E}_{\mu_{k-1}}(h_{k-1}) = \left[\mathcal{E}_{\mu_k}(h_k) - \mathcal{E}_{\mu_k}(h_{k-1}) \right] + \left[\mathcal{E}_{\mu_k}(h_{k-1}) - \mathcal{E}_{\mu_{k-1}}(h_{k-1}) \right].$$

By Kantorovich–Rubinstein duality, since for any fixed h and y the function $x \mapsto \ell(h(x), y)$ is L-Lipschitz in x, the second term is bounded as

$$\left| \mathcal{E}_{\mu_k}(h_{k-1}) - \mathcal{E}_{\mu_{k-1}}(h_{k-1}) \right| \leq LW_1(\mu_k, \mu_{k-1})$$

Now consider the difference

$$\mathcal{E}_{\mu_k}(h_k) - \mathcal{E}_{\mu_k}(h_{k-1}) = T_1 + T_2 + T_3 + T_4 + T_5,$$

where

$$\begin{aligned}
T_1 &= \mathcal{E}_{\mu_k}(h_k) - \widehat{\mathcal{E}}_k(h_k), \\
T_2 &= \widehat{\mathcal{E}}_k(h_k) - \widehat{R}_k(h_k), \\
T_3 &= \widehat{R}_k(h_k) - \widehat{R}_k(h_{k-1}), \\
T_4 &= \widehat{R}_k(h_{k-1}) - \widehat{\mathcal{E}}_k(h_{k-1}), \\
T_5 &= \widehat{\mathcal{E}}_k(h_{k-1}) - \mathcal{E}_{\mu_k}(h_{k-1}).
\end{aligned}$$

By standard Rademacher generalization bounds and Hoeffding-type concentration (e.g., (Mohri et al., 2018, Theorem 3.5)), we have for each k with probability at least $1-\delta/K$,

$$\left| \widehat{\mathcal{E}}_k(h) - \widehat{R}_k(h) \right| \leq 2\widehat{\mathfrak{R}}_k + 3\sqrt{\frac{\log(2K/\delta)}{2n_k}} \leq \frac{2C_r}{\sqrt{n_k}} + 3\sqrt{\frac{\log(2K/\delta)}{2n_k}}.$$

Applying a union bound over all $k \in [K]$, we conclude that with probability at least $1-\delta$, for all k, both T_2 and T_4 are bounded by this quantity. Define

$$C_\delta := 4C_r + 6\sqrt{\frac{\log(2K/\delta)}{2}},$$

so

$$T_2 + T_4 \le \frac{C_\delta}{\sqrt{n_k}}.$$

The regularized ERM step ensures $T_3 \le \lambda B^2$.

For T_1 and T_5, observe for any h that

$$\mathcal{E}_{\mu_k}(h) - \widehat{\mathcal{E}}_k(h) = \underbrace{\big(\mathcal{E}_{\mu_k}(h) - \overline{\mathcal{E}}_k(h)\big)}_{\text{rejection (missing coverage)}} + \underbrace{\big(\overline{\mathcal{E}}_k(h) - \widehat{\mathcal{E}}_k(h)\big)}_{\text{label substitution on accepted set}}.$$

The gap between full and masked true risk,

$$\mathcal{E}_{\mu_k}(h) - \overline{\mathcal{E}}_k(h) = \mathbb{E}[(1 - \mathsf{A}_k(x))\,\ell(h(x), y)] \le 1 - \rho_k,$$

is the *rejection (missing coverage)* penalty, i.e., loss on points the filter does not accept at round k. The gap between masked true risk and masked pseudo-risk,

$$\overline{\mathcal{E}}_k(h) - \widehat{\mathcal{E}}_k(h) = \mathbb{E}\big[\mathsf{A}_k(x)\,\big(\ell(h(x), y) - \ell(h(x), \tilde{y}_k(x))\big)\big],$$

is the *label substitution* error on the accepted set. By the absolute-deviation definition $\varepsilon_k := \max_{h \in \{h_k, h_{k-1}\}} \big|\overline{\mathcal{E}}_k(h) - \widehat{\mathcal{E}}_k(h)\big|$, both h_k and h_{k-1} satisfy $\big|\overline{\mathcal{E}}_k(h) - \widehat{\mathcal{E}}_k(h)\big| \le \varepsilon_k$. Therefore, $T_1 \le (1-\rho_k) + \varepsilon_k$ and $T_5 \le \varepsilon_k$ (because $\overline{\mathcal{E}}_k(h_{k-1}) - \mathcal{E}_{\mu_k}(h_{k-1}) \le 0$).

Combining all five terms, we get

$$\mathcal{E}_{\mu_k}(h_k) - \mathcal{E}_{\mu_k}(h_{k-1}) \le 2\varepsilon_k + \frac{C_\delta}{\sqrt{n_k}} + \lambda B^2.$$

Therefore,

$$\mathcal{E}_{\mu_K}(h_K) - \mathcal{E}_{\mu_0}(h_0) \le \sum_{k=1}^{K} \left((1-\rho_k) + 2\varepsilon_k + \frac{C_\delta}{\sqrt{n_k}} + LW_1(\mu_k, \mu_{k-1}) + \lambda B^2 \right),$$

with probability at least $1-\delta$. □

B Proofs of Filtering Lemmas

Recall Lemma 1: Let ℓ be a bounded loss with range in $[0,1]$. Suppose h_{k-1} is confidence-calibrated with calibration function $\phi(\cdot)$. Then, the pseudo-labeling error at step k satisfies

$$\varepsilon_k \;\le\; \rho_k\,\phi(\tau_k) \;\le\; \phi(\tau_k).$$

Proof. For any $h \in \{h_k, h_{k-1}\}$, by $\ell \in [0,1]$,

$$\begin{aligned}
\big|\overline{\mathcal{E}}_k(h) - \widehat{\mathcal{E}}_k(h)\big| &= \big|\,\mathbb{E}\big[\mathsf{A}_k(x)\,\big(\ell(h(x), y) - \ell(h(x), \tilde{y}_k(x))\big)\big]\,\big| \\
&\le \mathbb{E}\big[\mathsf{A}_k(x)\,\big|\,\ell(h(x), y) - \ell(h(x), \tilde{y}_k(x))\,\big|\big] \\
&\le \mathbb{P}[\mathsf{A}_k(x) = 1,\ \tilde{y}_k(x) \ne y] \\
&= \rho_k\,\mathbb{P}[\tilde{y}_k(x) \ne y \mid \mathsf{A}_k(x) = 1] \;\le\; \rho_k\,\phi(\tau_k).
\end{aligned}$$

Taking the max over $h \in \{h_k, h_{k-1}\}$ gives $\varepsilon_k \le \rho_k\,\phi(\tau_k) \le \phi(\tau_k)$. □

Recall Lemma 2: Let h_{k-1} be a classifier with logits f_{k-1}, and let $\tilde{y}_k(x) := \hat{y}_{k-1}(x)$ be the pseudo-label assigned when $\text{margin}_{k-1}(x) \geq m_k$. If the margin misclassification function satisfies $\zeta(m_k)$, then

$$\varepsilon_k \;\leq\; \rho_k\, \zeta(m_k) \;\leq\; \zeta(m_k).$$

Proof. The result follows by the same argument as in Lemma 1, replacing confidence thresholding with margin thresholding. On the accepted set,

$$\left|\overline{\mathcal{E}}_k(h) - \widehat{\mathcal{E}}_k(h)\right| \;\leq\; \mathbb{P}[\mathsf{A}_k(x) = 1,\; \tilde{y}_k(x) \neq y] \;\leq\; \rho_k\, \zeta(m_k).$$

This finishes the proof. □

C Proofs for Section 5.3

Lemma 4 (Quantile accuracy; Restatement of Lemma 3)**.** *Let M_k unlabeled scores be drawn i.i.d. from μ_k, and let θ_k be the empirical $(1-q_k)$-quantile obtained by sorting the scores in descending order and taking the element at index $\lceil q_k M_k \rceil$. Then, with probability at least $1-\delta$ uniformly over $k = 1, \ldots, K$,*

$$\left|\rho_k(\theta_k) - q_k\right| \;\leq\; \sqrt{\frac{2\,\log(2K/\delta)}{M_k}}.$$

Proof. Fix a round k and let $X_1, \ldots, X_{M_k}$ be the i.i.d. scores $\text{score}_{k-1}(x)$ for $x \sim \mu_k$. Let $F(t) = \Pr[X \leq t]$ be the population CDF and $F_{M_k}(t) = \frac{1}{M_k}\sum_{i=1}^{M_k} \mathbf{1}\{X_i \leq t\}$ its empirical CDF; define the corresponding functions $S(t) = 1 - F(t^-) = \Pr[X \geq t]$ and $S_{M_k}(t) = 1 - F_{M_k}(t^-) = \frac{1}{M_k}\sum_{i=1}^{M_k} \mathbf{1}\{X_i \geq t\}$. Note that $S(t) = \rho_k(t)$ and $S_{M_k}(t)$ is the empirical coverage at threshold t.

According to the Dvoretzky–Kiefer–Wolfowitz (DKW) inequality (van der Vaart and Wellner, 1996), for any $\eta > 0$,

$$\Pr\Big(\sup_t |F_{M_k}(t) - F(t)| > \eta\Big) \;\leq\; 2\,e^{-2M_k\eta^2}.$$

Since $S_{M_k}(t) - S(t) = -(F_{M_k}(t^-) - F(t^-))$, the same bound holds for the survival functions, i.e.,

$$\Pr\Big(\sup_t |S_{M_k}(t) - S(t)| > \eta\Big) \;\leq\; 2\,e^{-2M_k\eta^2}.$$

Choose $\eta_k = \sqrt{\frac{\log(2K/\delta)}{2\,M_k}}$. By a union bound over $k = 1, \ldots, K$, with probability at least $1-\delta$ we have

$$\sup_t |S_{M_k}(t) - S(t)| \;\leq\; \eta_k \qquad \text{simultaneously for all } k.$$

By construction of the empirical $(1-q_k)$-quantile θ_k (descending order with index $\lceil q_k M_k \rceil$), we have

$$S_{M_k}(\theta_k) \;\geq\; q_k \qquad \text{and} \qquad S_{M_k}(\theta_k) \;\leq\; q_k + \tfrac{1}{M_k},$$

because the empirical function is a step function with jumps of size $1/M_k$ at observed scores. Therefore, on the DKW event,

$$q_k - \eta_k \;\leq\; S(\theta_k) \;\leq\; q_k + \tfrac{1}{M_k} + \eta_k.$$

Equivalently,

$$\left|\,\rho_k(\theta_k) - q_k\,\right| \;\leq\; \eta_k + \tfrac{1}{M_k}.$$

Since $\frac{1}{M_k} \leq \eta_k := \sqrt{\frac{\log(2K/\delta)}{2\,M_k}}$ whenever $M_k \geq \frac{2}{\log(2K/\delta)}$, we obtain

$$\left|\,\rho_k(\theta_k) - q_k\,\right| \;\leq\; 2\eta_k = \sqrt{\frac{2\log(2K/\delta)}{M_k}}.$$

This finishes the proof. □

Proposition 2 (Pseudo-label error under percentile control; Restatement of Prop. 1)**.** *Let $q_k \geq 1 - \frac{c}{k}$ and choose θ_k as the empirical $(1-q_k)$-quantile. Suppose Assumption 1 holds and $\theta_k \geq \theta_0$. Then, with probability at least $1-\delta$ (uniformly in k),*

$$\psi_k(\theta_k) \;\leq\; C_t\Big(1-\rho_k(\theta_k)\Big)^{\alpha} \;\leq\; C_t\left(\tfrac{c}{k}+\sqrt{\tfrac{2\log(2K/\delta)}{M_k}}\right)^{\alpha}, \qquad \varepsilon_k \;\leq\; \rho_k(\theta_k)\,\psi_k(\theta_k) \;\leq\; C_t\left(\tfrac{c}{k}+\sqrt{\tfrac{2\log(2K/\delta)}{M_k}}\right)^{\alpha}.$$

In particular, if $M_k = \omega\big(k^2\log(K/\delta)\big)$ so that $\sqrt{\log(2K/\delta)/M_k} = o(1/k)$, then $\psi_k(\theta_k), \varepsilon_k = O(k^{-\alpha})$ and $\sum_{k=1}^{K}\varepsilon_k = O(\log K)$ when $\alpha = 1$ (and $O(1)$ when $\alpha > 1$).

Proof. From the percentile schedule, $1 - q_k \leq \frac{c}{k}$. By Lemma 3, with probability at least $1-\delta$,

$$\big|\rho_k(\theta_k) - q_k\big| \;\leq\; \sqrt{\tfrac{2\log(2K/\delta)}{M_k}},$$

hence,

$$1-\rho_k(\theta_k) \;\leq\; (1-q_k) + \sqrt{\tfrac{2\log(2K/\delta)}{M_k}} \;\leq\; \tfrac{c}{k} + \sqrt{\tfrac{2\log(2K/\delta)}{M_k}}.$$

Applying Assumption 1 at θ_k gives

$$\psi_k(\theta_k) \;\leq\; C_t\Big(1-\rho_k(\theta_k)\Big)^{\alpha} \;\leq\; C_t\left(\tfrac{c}{k}+\sqrt{\tfrac{2\log(2K/\delta)}{M_k}}\right)^{\alpha}.$$

Finally, by Lemmas 1 and 2, $\varepsilon_k \leq \rho_k(\theta_k)\,\psi_k(\theta_k) \leq \psi_k(\theta_k)$, yielding the stated bound. The summability claim follows from comparing $\sum_{k=1}^{K} k^{-\alpha}$ to a harmonic or convergent p-series and noting the DKW term is $o(1/k)$ by assumption on M_k. □

D Pseudo-code for Algorithms

Algorithm 1: Confidence- or Margin-Filtered Self-Training

Input: Initial labeled data $S_0 \sim \mu_0$; unlabeled sets $\{S_k\}_{k=1}^{K}$; filtering method (CONFIDENCE or MARGIN); coverage targets $\{q_k\}_{k=1}^{K}$; regularization λ.
Output: Final classifier h_K.

$h_0 \leftarrow \textsc{Train}(S_0, \lambda)$; // e.g., $\arg\min_h \widehat{R}(h) + \lambda\|h\|^2$
for $k = 1$ **to** K **do**
  $\theta_k \leftarrow \textsc{PercentileThreshold}(S_k, h_{k-1}, q_k, \textsc{method})$; // empirical $(1-q_k)$-quantile
  $\tilde{S}_k \leftarrow \emptyset$;
  foreach $x \in S_k$ **do**
    if METHOD=CONFIDENCE **then**
      $\text{score}(x) \leftarrow \max_c h_{k-1}(x)_c, \quad \tilde{y}_k(x) \leftarrow \arg\max_c h_{k-1}(x)_c$;
    else if METHOD=MARGIN **then**
      $f(x) \leftarrow$ logits from $h_{k-1}, \quad \hat{y}(x) \leftarrow \arg\max_i f_i(x)$;
      $\text{score}(x) \leftarrow f_{\hat{y}(x)}(x) - \max_{j \neq \hat{y}(x)} f_j(x), \quad \tilde{y}_k(x) \leftarrow \hat{y}(x)$;
    if *score*$(x) \geq \theta_k$ **then**
      add $(x, \tilde{y}_k(x))$ to $\tilde{S}_k$
  $h_k \leftarrow \textsc{Train}(\tilde{S}_k, \lambda)$
return h_K

Algorithm 2: PercentileThreshold

Input: Unlabeled set S; model h; quantile level q; filtering method.
Output: Threshold θ.

$\mathcal{S} \leftarrow [\,]$;
foreach $x \in S$ **do**
  if METHOD=CONFIDENCE **then**
    score$(x) \leftarrow \max_c h(x)_c$
  else if METHOD=MARGIN **then**
    $f(x) \leftarrow$ logits from h, $\hat{y}(x) \leftarrow \arg\max_i f_i(x)$;
    score$(x) \leftarrow f_{\hat{y}(x)}(x) - \max_{j \neq \hat{y}(x)} f_j(x)$
  append score(x) to $\mathcal{S}$
sort $\mathcal{S}$ in descending order ;
$\theta \leftarrow \mathcal{S}[\lceil q \cdot |S| \rceil]$; // empirical q-quantile (retain top-q scores)
return θ

E Intermediate Domains via OT Geodesics.

Let μ_0 and μ_K denote the source and target domains, respectively. We generate intermediate domains along the 2-Wasserstein geodesic between μ_0 and μ_K. Write $W_2^2(\cdot,\cdot)$ for the squared 2-Wasserstein distance and define the geodesic interpolation (Villani, 2008):

$$\mu^\lambda \;:=\; \operatorname*{arg\,min}_{\mu \in \mathcal{P}(\mathcal{X})} (1-\lambda)\, W_2^2(\mu,\mu_0) + \lambda\, W_2^2(\mu,\mu_K),\;\; \lambda \in [0,1]. \tag{14}$$

For empirical measures $\mu_0 = \frac{1}{n_s}\sum_{i=1}^{n_s} \delta_{x_i^{(s)}}$ and $\mu_K = \frac{1}{n_t}\sum_{j=1}^{n_t} \delta_{x_j^{(t)}}$, let $\gamma_0 \in \mathbb{R}_{\geq 0}^{n_s \times n_t}$ be an optimal Kantorovich coupling solving

$$\begin{aligned} \gamma_0 \in \operatorname*{arg\,min}_{\gamma \geq 0} & \sum_{i=1}^{n_s}\sum_{j=1}^{n_t} \gamma_{ij}\, \|x_i^{(s)} - x_j^{(t)}\|_2^2 \\ \text{s.t.} \quad & \gamma\, \mathbf{1}_{n_t} = \tfrac{1}{n_s}\mathbf{1}_{n_s}, \;\; \gamma^\top \mathbf{1}_{n_s} = \tfrac{1}{n_t}\mathbf{1}_{n_t}. \end{aligned} \tag{15}$$

Then, the discrete geodesic at λ is the pushforward of γ_0 by $(x^{(s)}, x^{(t)}) \mapsto (1-\lambda)x^{(s)} + \lambda x^{(t)}$:

$$\widehat{\mu}^\lambda \;=\; \sum_{i=1}^{n_s}\sum_{j=1}^{n_t} \gamma_0(i,j)\; \delta_{(1-\lambda)x_i^{(s)} + \lambda x_j^{(t)}}.$$

We set the round-k domain to $\mu_k := \mu^{\lambda_k}$ with $\lambda_k = \frac{k}{K}$, and obtain an unlabeled sample $S_k \sim \mu_k$ (e.g., by sampling support points proportional to $\gamma_0(i,j)$).

F Dataset Description and Training Protocol

Dataset description. Here, we describe the benchmark datasets that we used to conduct our main experiments.

Rotated MNIST. MNIST digit images (LeCun et al., 1998) are rotated to induce a gradual geometric shift. The source domain contains digits at 0°, and the target domain at 45°. Eight evenly spaced rotations between these angles form the intermediate domains.

Color-shift MNIST. A synthetic variant of MNIST digits (He et al., 2024) with a gradual appearance shift. Pixel values shift linearly from $[0,1]$ in the source domain to $[1,2]$ in the target domain, with ten evenly spaced steps forming the intermediate domains.

Portraits. A dataset of 18,000 grayscale portraits of U.S. high-school seniors, spanning the years 1905–2013 (Ginosar et al., 2015). Images are sorted chronologically to create a temporal shift, with one source domain, seven intermediate groups, and one target domain.

Cover type. A non-vision, tabular benchmark from the UCI forest-cover dataset (Blackard and Dean, 1999), with 54 cartographic input features. Gradual shift is imposed by sorting samples by distance to water, forming one source, ten intermediates, and one target domain.

Office-Home. A visual domain adaptation benchmark comprising 65 object categories across four domains: Art, Clipart, Product, and Real World (Venkateswara et al., 2017). Unlike the other benchmarks, Office-Home does not provide an inherent ordered sequence of intermediate domains; gradual trajectories are therefore constructed by inserting OT-generated intermediate domains between source and target, as described in Section 6.2 and Appendix E.

Implementation details and protocol. All experiments are implemented in PyTorch (Paszke et al., 2019) and executed on NVIDIA RTX A2000 GPUs. We replicate the model architectures used in (He et al., 2024): a 4-layer CNN followed by 3 fully connected layers for Rotated MNIST, Color-Shift MNIST, and Portraits; and a 3-layer MLP with 256 hidden units for Cover Type. We use the Adam optimizer (Kingma and Ba, 2015), BatchNorm (Ioffe and Szegedy, 2015), and Dropout (Srivastava et al., 2014) for training stability.

Training hyperparameters. We use the same training configuration across GST, GOAT, CFSTDA, and MFSTDA. Each self-training step is trained for 10 epochs using a batch size of 128, learning rate of 0.0001, and 2 data loader workers. Filtering in CFSTDA and MFSTDA follows a class-wise thresholding procedure with a minimum 10% per-class retention rate based on confidence or margin scores.

Intermediate domain configuration. We evaluate our filtering-based self-training methods under two DA configurations, consistent with prior work (He et al., 2024). In the first setting, we use only the given intermediate domains that are labeled and available as part of the dataset. In the second setting, we augment the sequence by inserting generated intermediate domains between consecutive given domains.

Generated domains are constructed following the optimal-transport (OT) geodesic formulation described in Appendix E, which computes new feature distributions by interpolating between pairs of consecutive labeled domains. This process produces smoothly transitioning domains that bridge the distributional gaps between given domains. In our setup, we vary the number of given intermediate domains from 0 to 3. For each interval between given domains, we insert 0, 1, or 2 generated domains. For example, with 2 given domains (forming 3 intervals along the source-to-target path), inserting 2 generated domains per interval results in a total of 10 domains: source, 2 given, target, and 6 generated.

G Additional Results

This section presents additional analyses and experiments that complement the main results. We provide both qualitative and quantitative findings to further support the trends observed in the paper. The first part includes an extended synthetic data experiment using a Gaussian mixture distribution, investigating the effect of class imbalance and quantile scheduling on pseudo-labeling dynamics. The second part reports extended results on real benchmarks, confirming the consistency and robustness of our confidence- and margin-based filtering strategies across different domain-shift settings.

Additional synthetic data. To complement the experiments on the synthetic dataset, we provide an additional example using a 2D Gaussian mixture distribution with gradual domain shifts. The setup and evaluation protocol follow exactly the same procedure described in Section 6.1, except that the data now consists of two Gaussian clusters that rotate by 2° per step, totaling a 90° rotation between the source and target domains over 45 rounds. At each round k, we draw an unlabeled batch, train the model h_k on the top $q_k = 1 - \frac{c}{k}$ fraction of samples according to either the confidence (CFSTDA) or margin (MFSTDA) criterion. Figure 4 illustrates the step–by–step refinement of the decision boundary and the progressive expansion of the accepted region, while Figure 5 presents the corresponding pseudo–labeling errors ε_k, accepted–set losses ψ_k, and coverage ρ_k over rounds. The observed decay of ψ_k and ε_k follows the behavior predicted by Proposition 1, and the fitted positive slope $\hat{\alpha}$ in the log–log plot confirms that the tail–decay assumption (Assumption 1) continues to hold for this Gaussian setting.

Effect of class imbalance. We analyze the impact of class imbalance using the two-moons synthetic dataset under the same gradual shift setup, but with a class prior of 30% vs. 70%. A known challenge in pseudo-labeling is class imbalance: without constraints, percentile filtering tends to favor majority classes, causing minority classes

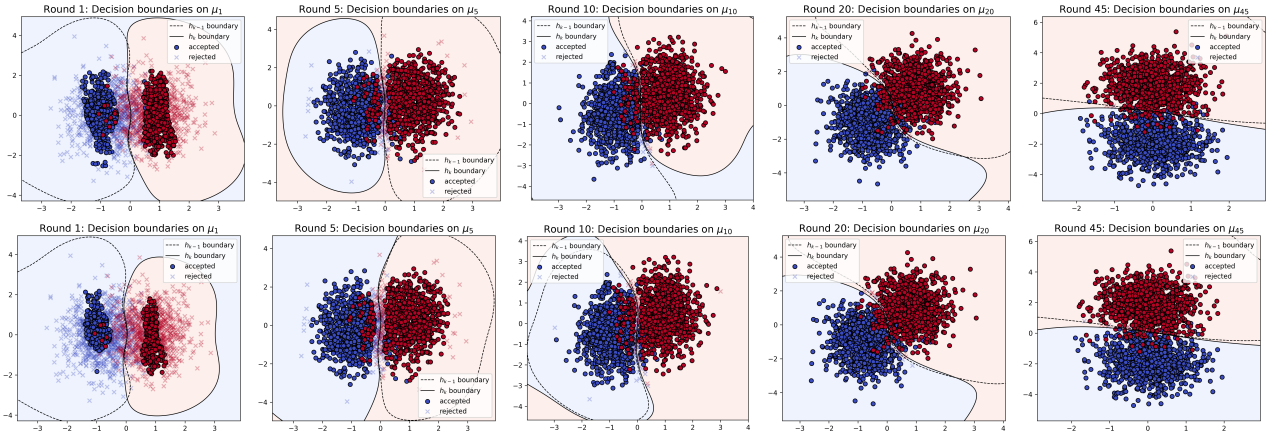


Figure 4: Step-by-step visualization of pseudo-label filtering and decision-boundary refinement across selected rounds ($k \in 1, 5, 10, 20, 45$). The top row shows confidence-based filtering (CFSTDA), and the bottom row shows margin-based filtering (MFSTDA).

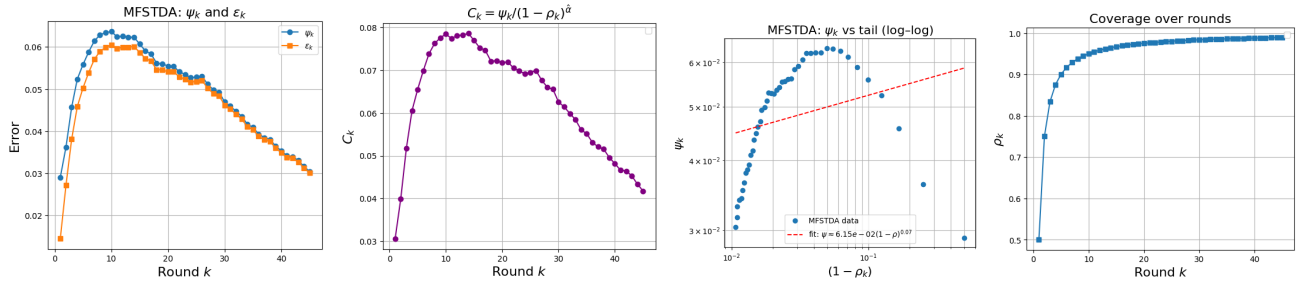


Figure 5: Empirical validation of pseudo-labeling error bounds under gradual domain adaptation. **Left:** Conditional error ψ_k and pseudo-labeling error ε_k. **Center-left:** Ratio $\psi_k/(1-\rho_k)^{\hat{\alpha}}$ (proportional to the scaled errors $k^{\hat{\alpha}}\psi_k$ and $k^{\hat{\alpha}}\varepsilon_k$). **Center-right:** Log–log plot of ψ_k against $(1-\rho_k)$ with fitted slope $\hat{\alpha} \approx 0.07$, **Right:** Coverage ρ_k over rounds.

to be underrepresented or even completely dropped in early rounds. To mitigate this issue, our implementation enforces a per-class minimum retention of 10-20% at each round (based on predicted labels). Figure 6 compares the evolution of per-class coverage and decision boundaries with (bottom row) and without (top row) balancing, illustrating that per-class retention prevents minority collapse and stabilizes the learning dynamics. Figure 7 further shows the quantitative impact of balancing: the (left) panel reports the pseudo-labeling error ε_k, which remains low when per-class retention is applied; the (center) panel tracks target-domain accuracy, showing steady improvement with balancing; and the (right) panel shows higher overall coverage ρ_k with balancing, since per-class retention compensates for classes whose acceptance drops below the threshold.

Comparison with baseline self-training. We now validate our theoretical insights on real data (Color-shift MNIST) by comparing our filtered self-training methods (CFSTDA and MFSTDA) with GST, using only given intermediate domains (no generated ones). Filtering is applied using percentile schedules $q_k = 1 - c/k$, with $c = 0.5$ for CFSTDA and $c = 0.7$ for MFSTDA. Figure 8 reports target domain accuracy (Left), pseudo-labeling error ε_k (center), and accepted-set coverage ρ_k (Right) over 10 self-training rounds. CFSTDA and MFSTDA both exhibit faster and more stable improvement in accuracy, ultimately converging to higher target performance. In the middle panel, we observe that the accepted-set pseudo-labeling errors ε_k remain significantly lower for CFSTDA and MFSTDA across all rounds, while GST shows a continuous increase in error. This illustrates that our percentile-thresholding schemes not only control the pseudo-label quality but also enable more effective model training at each step. The right-hand panel confirms that coverage grows smoothly under the scheduled thresholds, whereas GST accepts all points ($\rho_k = 1$).

Comparison with GOAT. Figure 9 visualizes the evolution of target accuracy for our filtered self-training methods (CFSTDA and MFSTDA) compared to GOAT, under a 10-domain adaptation path constructed with 2 given and 2 generated intermediate domains. We include $k = 0$ to show the performance of the initial source-trained model h_0 evaluated directly on the target domain. This highlights that all methods start training from the same source model. Both CFSTDA and MFSTDA use an increasing quantile schedule ($q_k = 1 -$

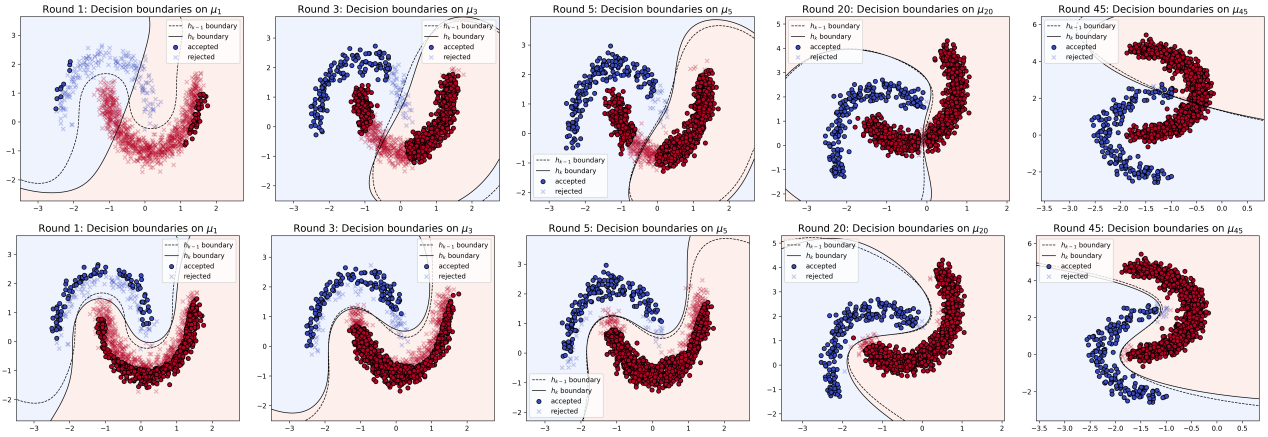


Figure 6: Effect of class imbalance and per-class retention during gradual self-training. Top row: decision boundaries and accepted samples *without* per-class balancing. Bottom row: applying a per-class minimum retention preserves both classes and yields smoother adaptation.

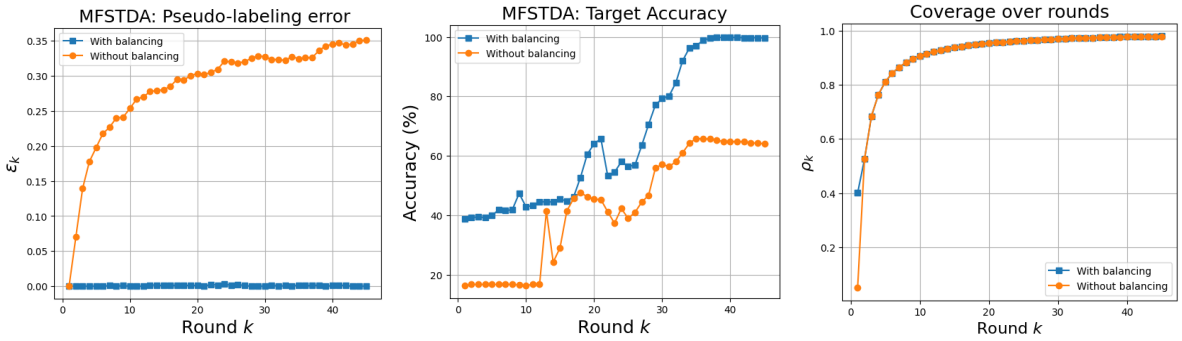


Figure 7: Effect of per-class balancing on pseudo-labeling performance under class imbalance. **Left:** pseudo-labeling error ε_k; **Center:** target-domain accuracy; **Right:** coverage ρ_k.

c/k) to progressively expand the accepted pseudo-labeled set. On Rotated MNIST (Left), GOAT achieves higher accuracy in the early rounds, while CFSTDA gradually overtakes it as training progresses. MFSTDA performs comparably, though it remains slightly below CFSTDA in later steps. On Color-Shift MNIST (Right), both filtering-based methods clearly outperform GOAT, with MFSTDA showing the strongest overall accuracy throughout adaptation. All accuracy curves are reported with 95% confidence intervals computed over 5 random seeds.

Full accuracy tables and additional Office-Home results. Complete target-accuracy results (%) for the four GDA benchmarks—Rotated MNIST, Color-Shift MNIST, Covtype, and Portraits—are reported in Tables 3–6. Each table reports performance across all combinations of given domains (0–3) and generated domains (gen = 0, 1, 2). The best (boldface) and second-best (underlined) methods in each group are highlighted.

Table 7 reports results for the Art→Real and Real→Art directions on Office-Home, both of which present challenging adaptation scenarios. Results are consistent with the main-text findings: CFSTDA achieves 75.8% on Art→Real at 1 generated domain, outperforming all baselines. In the Real→Art direction, both CFSTDA and MFSTDA achieve 64.5% at 0 generated domains, exceeding DANN and remaining competitive with DeepCoral in the direct adaptation setting.

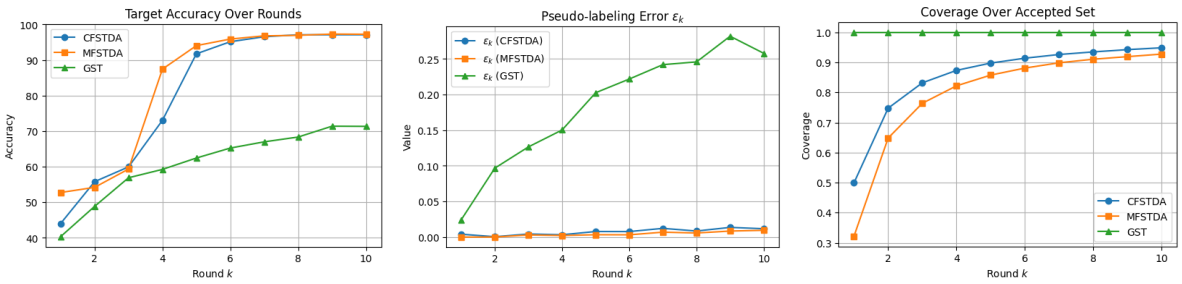


Figure 8: Evaluation of our CFSTDA/MFSTDA approach on Color-shift MNIST dataset and GST approach for different domain rounds k. **Left**: Target accuracy. **Middle**: Pseudo-labeling error ε_k. **Right**: Coverage ρ_k, the fraction of pseudo-labeled points retained after filtering.

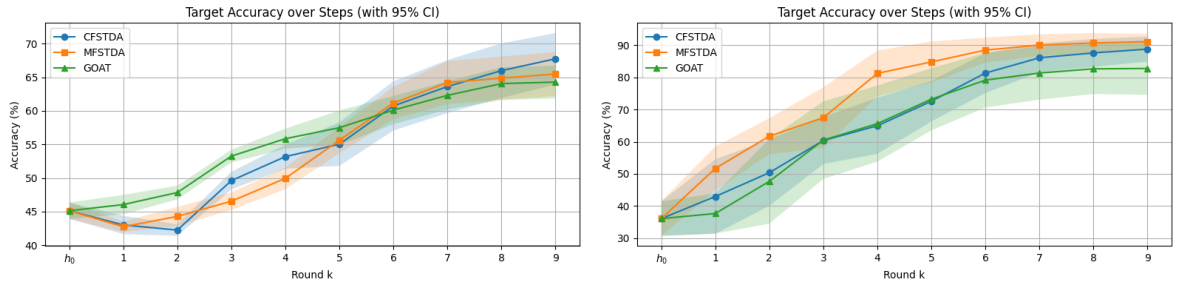


Figure 9: Target accuracy versus adaptation round along the 10 domains (2 given, 2 gen). Curves show GOAT, CFSTDA, and MFSTDA; shaded bands indicate 95% CIs over 5 seeds. **Left**: Rotated MNIST, **Right**: Color-Shift MNIST.

Table 3: Accuracy (%) on Color-shift MNIST dataset.

# Given Domains	GST (gen=0)	CFSTDA (gen=0)	MFSTDA (gen=0)	GOAT (gen=1)	CFSTDA (gen=1)	MFSTDA (gen=1)	GOAT (gen=2)	CFSTDA (gen=2)	MFSTDA (gen=2)
0	41.6±9.2	51.7 ±12.5	**59.7±13.0**	42.6±13.7	60.7±9.9	**87.9±7.8**	68.5±15.8	70.5±11.7	**91.4±2.6**
1	47.9±8.6	62.0±12.9	**82.6±15.1**	64.8±20.3	78.5±9.5	**91.3±3.3**	75.5±6.5	87.9±2.7	**92.2±2.8**
2	53.2±14.0	72.0±8.9	**91.5±5.6**	71.4±20.0	84.6±13.1	**92.0±7.7**	82.8±11.6	88.8±5.5	**91.1±3.4**
3	59.0±19.2	78.8±10.5	**92.6±5.1**	79.2±8.3	87.3±7.3	**92.4±3.2**	85.2±9.1	**89.3±4.1**	87.9±2.2

Table 4: Accuracy (%) on Rotated MNIST dataset.

# Given Domains	GST (gen=0)	CFSTDA (gen=0)	MFSTDA (gen=0)	GOAT (gen=1)	CFSTDA (gen=1)	MFSTDA (gen=1)	GOAT (gen=2)	CFSTDA (gen=2)	MFSTDA (gen=2)
0	44.8±2.3	47.1±2.9	**50.9±2.0**	44.7±3.3	45.4±2.3	**48.7±2.2**	42.7.±2.2	42.0±3.4	**47.1±2.6**
1	53.2±3.7	**57.2±3.0**	56.8±2.3	50.9±3.6	53.4±2.0	**58.5±1.6**	49.0±4.3	52.0±3.3	**62.8 ±3.0**
2	59.3±5.2	67.6±1.7	**69.9±4.0**	63.1±5.6	65.7± 1.9	**66.2±2.9**	64.3±3.5	**67.7±5.4**	65.5±4.7
3	69.1±5.6	**74.1±3.2**	74.0±4.1	78.1±4.2	**78.6±2.0**	73.2±5.8	**82.2±3.3**	79.0±8.8	78.1±7.7

Table 5: Accuracy (%) on Covtype dataset.

# Given Domains	GST (gen=0)	CFSTDA (gen=0)	MFSTDA (gen=0)	GOAT (gen=1)	CFSTDA (gen=1)	MFSTDA (gen=1)	GOAT (gen=2)	CFSTDA (gen=2)	MFSTDA (gen=2)
0	61.0±5.2	62.2±4.5	**62.3±2.0**	62.3±4.6	63.4±5.6	**65.4±1.5**	62.8±6.5	64.9±6.3	**65.7±1.5**
1	63.9±5.7	64.2±6.0	**67.6±2.3**	65.1±5.9	65.8±5.8	**70.3±3.3**	64.7±7.0	67.7±4.4	**68.1±6.6**
2	67.8±4.6	67.7±4.6	**69.7±3.2**	67.7±4.1	68.2±4.3	**71.2±2.6**	70.4±1.7	71.7±2.9	**72.5±2.3**
3	66.5±4.8	67.2±5.6	**69.2±3.7**	66.0±5.8	**71.3±3.5**	69.9±4.1	68.4±3.7	**72.8±2.8**	69.0±3.0

Table 6: Accuracy (%) on Portraits dataset.

# Given Domains	GST (gen=0)	CFSTDA (gen=0)	MFSTDA (gen=0)	GOAT (gen=1)	CFSTDA (gen=1)	MFSTDA (gen=1)	GOAT (gen=2)	CFSTDA (gen=2)	MFSTDA (gen=2)
0	77.4±1.3	76.5±1.6	**82.0±3.0**	77.8±0.8	78.6±1.8	**82.8±2.0**	75.8±1.7	76.4±3.9	**82.9±3.3**
1	76.3±1.7	76.6±1.8	**81.7±1.8**	74.9±3.9	78.6±2.3	**84.5±1.1**	76.9±3.7	81.4±4.1	**84.9±0.5**
2	76.3 ±1.7	77.2±4.8	**84.6±0.7**	79.2±3.8	79.5±3.4	**85.6±0.6**	81.1±2.6	79.8±3.1	**84.2±0.6**
3	83.0±1.9	80.7±2.3	**85.0±1.1**	77.9±7.4	79.8±4.3	**84.3±1.1**	80.4±2.7	79.3±5.7	**82.9±1.7**

Table 7: Extended evaluation on Office-Home under direct and generated gradual adaptation settings.

Method	**Office-Home Art→Real**				**Office-Home Real→Art**			
Baseline (no adaptation)	73.2±0.8				60.9±1.3			
DANN (Ganin et al., 2016)	73.0±0.9				63.9±0.7			
DeepCoral (Sun and Saenko, 2016)	73.1±0.5				64.3±0.9			
	0 gen	1 gen	2 gen	3 gen	0 gen	1 gen	2 gen	3 gen
GOAT (He et al., 2024)	73.4±0.9	73.7±0.8	73.2±0.7	71.6±0.8	61.7±1.0	61.8±1.6	62.1±0.6	60.8±1.2
CFSTDA	74.0±0.6	**75.8±0.7**	74.6±1.1	73.3±1.3	**64.5±0.9**	62.8±1.2	62.1±1.2	62.0±0.8
MFSTDA	73.0±0.4	75.7±0.5	74.9±1.1	73.5±1.0	64.5±0.7	62.3±1.2	62.5±1.4	62.3±1.2