

A Comprehensive Survey on Test-Time Adaptation under Distribution Shifts

Jian Liang, *Member, IEEE*, Ran He, *Senior Member, IEEE*, and Tieniu Tan, *Fellow, IEEE*

Abstract—Machine learning methods strive to acquire a robust model during training that can generalize well to test samples, even under distribution shifts. However, these methods often suffer from a performance drop due to unknown test distributions. Test-time adaptation (TTA), an emerging paradigm, has the potential to adapt a pre-trained model to unlabeled data during testing, before making predictions. Recent progress in this paradigm highlights the significant benefits of utilizing unlabeled data for training self-adapted models prior to inference. In this survey, we divide TTA into several distinct categories, namely, test-time (source-free) domain adaptation, test-time batch adaptation, online test-time adaptation, and test-time prior adaptation. For each category, we provide a comprehensive taxonomy of advanced algorithms, followed by a discussion of different learning scenarios. Furthermore, we analyze relevant applications of TTA and discuss open challenges and promising areas for future research. A comprehensive list of TTA methods can be found at <https://github.com/tim-learn/awesome-test-time-adaptation>.

Index Terms—test-time adaptation, source-free domain adaptation, test-time training, continual test-time adaptation, prior adaptation

1 INTRODUCTION

TRADITIONAL machine learning methods assume that the training and test data are drawn independently and identically (i.i.d.) from the same distribution [1]. However, when the test distribution (target) differs from the training distribution (source), we face the problem of *distribution shifts*. Such a shift poses significant challenges for machine learning systems deployed in the wild, such as images captured by different cameras [2], road scenes of different cities [3], and imaging devices in different hospitals [4]. As a result, the research community has developed a variety of generalization or adaptation techniques to improve model robustness against distribution shifts. For instance, *domain generalization* (DG) [5] aims to learn a model using data from one or multiple source domains that can generalize well to any out-of-distribution target domain. On the other hand, *domain adaptation* (DA) [6] follows the transductive learning principle to leverage knowledge from a labeled source domain to an unlabeled target domain.

This survey primarily focuses on *test-time adaptation* (TTA), which involves adapting a pre-trained model from the source domain to unlabeled data in the target domain before making predictions [7], [8], [9], [10]. While DG operates solely during the training phase, TTA has the advantage of being able to access test data from the target domain during the test phase. This enables TTA to enhance recognition performance through adaptation with the available test data. Additionally, DA typically necessitates access to both labeled data from the source domain and (unlabeled) data from the target domain simultaneously, which can be prohibitive in privacy-sensitive applications such as medical

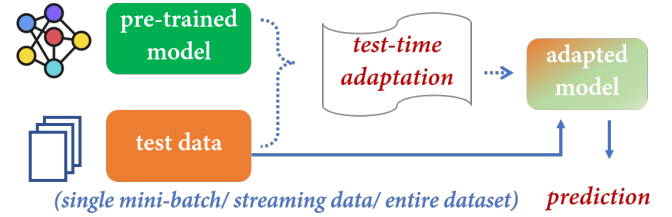


Fig. 1. The test-time adaptation (TTA) paradigm aims to adapt the pre-trained model to various types of unlabeled test data, including single mini-batch, streaming data, or an entire dataset, before making predictions. During the adaptation process, either the model or the input data can be altered to improve performance.

data. In contrast, TTA only requires access to the pre-trained model from the source domain, making it a secure and practical alternative solution.

Based on the characteristics of the test data ¹, TTA methods can be categorized into three distinct cases in Fig. 1: *test-time domain adaptation*, *test-time batch adaptation* (TTBA), and *online test-time adaptation* (OTTA). For better illustration, let us assume that we have m unlabeled mini-batches $\{b_1, \dots, b_m\}$ at test time. Firstly, test-time domain adaptation, also known as source-free domain adaptation (SFDA) [7], [11], [12], utilizes all m test batches for multi-epoch adaptation before generating final predictions. Secondly, TTBA individually adapts the pre-trained model to one ² or a few instances [8], [13], [14]. That is to say, the predictions of each mini-batch are independent of the predictions for the other mini-batches. Thirdly, OTTA adapts the pre-trained model to the target data $\{b_1, \dots, b_m\}$ in an online manner, where each mini-batch can only be observed once. However, knowledge learned from previously seen mini-batches can facilitate adaptation to the current mini-batch. It is worth noting that OTTA methods can be applied

¹ In this survey, we use the terms “test data” and “target data” interchangeably to refer to the data used for adaptation at test time.

² Such a single sample adaptation corresponds to a batch size of 1.

- The authors are with the State Key Laboratory of Multimodal Artificial Intelligence Systems and Center for Research on Intelligent Perception and Computing, Institute of Automation, Chinese Academy of Sciences.
- Ran He is also with the School of Artificial Intelligence, University of Chinese Academy of Sciences.
- Tieniu Tan is also with Nanjing University, China.
- E-mail: liangjian92@gmail.com, {rhe, tnt}@nlpr.ia.ac.cn.

to SFDA with multiple epochs, and TTBA methods can be applied to OTTA with the assumption of knowledge reuse. While most existing TTA methods focus on addressing data shift (e.g., covariate shift [1]), we also investigate the label shift scenario within the TTA framework, referred to as *test-time prior adaptation* (TTPA) [15], [16]. An illustrative example of label shift is medical diagnosis discussed in [17], where disease prevalences can vary significantly at different times of the year. Nonetheless, existing TTA studies have often misused terms due to a lack of comprehensive understanding of the relationships among these sub-problems.

In this survey, we for the first time define the broad concept of *test-time adaptation* and view the four aforementioned topics (i.e., SFDA, TTBA, OTTA, and TTPA) as its special cases. Subsequently, we thoroughly review the advanced algorithms for each of these topics and summarize a variety of applications related to TTA. Our contributions can be summarized into three following aspects.

- 1) To our knowledge, this is the first survey that provides a systematic overview of four distinct topics under the broad test-time adaptation paradigm, including source-free domain adaptation (SFDA, Sec. 3), test-time batch adaptation (TTBA, Sec. 4), online test-time adaptation (OTTA, Sec. 5), and test-time prior adaptation (TTPA, Sec. 6).
- 2) We propose a novel taxonomy of existing methods and provide a clear definition for each topic. We hope this survey will help readers develop a better understanding of the advancements in each topic.
- 3) We analyze various applications related to the TTA paradigm in Sec. 7, and provide an outlook of recent emerging trends and open problems in Sec. 8 to shed light on future research directions.

Comparison with previous surveys. Our survey is related to other surveys on domain adaptation [6], [18], [19], and prior adaptation [20]. However, our specific focus is on test-time learning with distribution shifts. Recently, two surveys have been published on source-free domain adaptation [21], [22], which is only a particular topic of test-time adaptation discussed in our survey. Even for the specific topic, we provide a new taxonomy that covers more related papers. Another survey [23] considers SFDA as a problem of data-free knowledge transfer, which shares some overlap with our survey. However, our survey differs in that it unifies SFDA and other related problems from the perspective of model adaptation under distribution shifts, which is a novel and pivotal contribution to the transfer learning field.

2 RELATED RESEARCH TOPICS

2.1 Domain Adaptation and Domain Generalization

As a special case of transfer learning [24], domain adaptation (DA) [25] leverages labeled data from a source domain to learn a classifier for an unlabeled target domain with a different distribution, in the transductive learning manner [26]. There are two major assumptions of domain shift [1], [27]: *covariate shift* in which the features cause the labels; and *label shift* in which the labels cause the features. We briefly introduce a few popular techniques and refer the reader to the existing literature on DA (e.g., [6], [18], [19]) for further

information. DA methods rely on the existence of source data to bridge the domain gap, and existing techniques can be broadly divided into four categories, i.e., input-level translation [28], [29], feature-level alignment [30], [31], [32], output-level regularization [33], [34], [35], and prior estimation [17], [36], [37]. If it is possible to generate training data from the source model [12], then the SFDA problem can be addressed using standard DA methods.

Specifically, one closely related topic is **one-shot domain adaptation** [38], [39], [40], [41], [42], which involves adapting to only one unlabeled instance while still requiring the source domain during adaptation. Another closely related topic is **online domain adaptation** [43], [44], which involves adapting to an unlabeled target domain with streaming data that is deleted immediately after adaptation. Inspired by federated learning [45], several studies [46], [47], [48] on **federated domain adaptation** offer an alternative solution for test-time domain adaptation that acquires the feedback (e.g., gradients and model parameters) from the target domain to the source domain multiple times. Nonetheless, this solution may not be feasible for certain scenarios due to the limited availability of training data and the high communication costs involved in transferring data across multiple devices, as compared to one-time model adaptation [7], [8], [9].

Domain generalization (DG) [49], [50], [51] aims to learn a model from one or multiple different but related domains that can generalize well on unseen testing domains. Researchers often develop specific training techniques to improve the generalization capability of the pre-trained model, which can be compatible with the studied TTA paradigm. For further information on this topic, we refer the reader to existing literature (e.g., [5], [52]).

2.2 Hypothesis Transfer Learning

Hypothesis transfer learning (HTL) [53] is another special case of transfer learning where pre-trained models (source hypotheses) retain information about previously encountered tasks. Shallow HTL methods [54], [55], [56], [57], [58], [59] generally assume that the optimal target hypothesis is closely related to these source hypotheses. For example, Ahmed *et al.* [59] learn the optimal metric with labeled target data under the assumption that the target metric is a convex combination of source metrics. Other methods [60], [61] extend this approach to a semi-supervised scenario where unlabeled target data are also used for training. Fine-tuning [62] is a typical example of a deep HTL method that may update a partial set of parameters in the source model. Although HTL methods assume no explicit access to the source domain or any knowledge about the relatedness of the source and target distributions, they still require a certain number of labeled data in the target domain.

2.3 Continual Learning and Meta-Learning

Continual learning (CL) [63] aims at learning a model for multiple tasks in a sequence, where knowledge obtained from the preceding tasks is gradually accumulated for future tasks. There are three fundamental scenarios of CL [64]: task-incremental, domain-incremental, and class-incremental learning. To name a few, several recent works [65], [66] consider incremental learning under domain shift.

Existing CL methods [67], [68] fall into three main categories: rehearsal-based [69], [70], parameter-based regularization [71], [72], and generative-based [67], [68], [73]. Even though the latter two cases do not access the training data of previous tasks, CL methods focus more on the anti-forgetting ability after learning a supervised task.

Meta-learning [74] shares a similar assumption with continual learning, but with training data randomly drawn from a task distribution, while test data are tasks with few examples. MAML [75] is a representative approach that learns the initialization of a model’s parameters to achieve optimal fast learning on a new task using a small number of samples and gradient steps. Generally, meta-learning offers a straightforward solution for test-time adaptation without the incorporation of test data in the meta-training stage.

2.4 Data-Free Knowledge Distillation

Knowledge distillation (KD) [76] aims to transfer knowledge from a teacher model to a student model by matching the network outputs or intermediate features. To address privacy and confidentiality concerns, the data-free KD paradigm [23] is proposed without requiring access to the original training data. Current data-free KD methods can be roughly divided into two categories: adversarial training [77], [78], [79], which focuses on generating worst-case synthetic samples for student learning, and data prior matching [80], [81], [82], where synthetic samples are forced to satisfy priors like class prior, activation regularization, and batch normalization statistics. A recent work [83] even considers a challenging scenario, decision-based black-box KD, where the teacher model is not accessible but returns only one-hot predictions. Compared with TTA, data-free KD performs knowledge transfer between models instead of distribution-shifted datasets.

2.5 Self-Supervised and Semi-Supervised Learning

Self-supervised learning [84] is a learning paradigm that focuses on how to learn from unlabeled data by obtaining supervisory signals from the data itself through pretext tasks that leverage its underlying structure. Early pretext tasks in the computer vision field include image colorization [85], image inpainting [86], image rotation [87], context prediction [88], video prediction [89]. Advanced pretext tasks like clustering [90], [91] and contrastive learning [92], [93], [94] have achieved remarkable success, even exceeding the performance of their supervised counterparts. Self-supervised learning is also popular in other fields like natural language processing [95], speech processing [96], and graph-structured data [97]. For TTA tasks, these self-supervised learning techniques can be utilized to help learn discriminative features [98] or act as an auxiliary task [8].

Semi-supervised learning [99] is another learning paradigm concerned with leveraging unlabeled data to reduce the reliance on labeled data. A common objective for semi-supervised learning comprises two terms: a supervised loss over labeled data and an unsupervised loss over unlabeled data. Regarding the latter term, there are three typical cases: self-training [100], [101], which encourages the model to produce confident predictions; consistency regularization under input variations [102], [103], [104] and

model variations [105], [106], which forces networks to output similar predictions when inputs or models are perturbed; and graph-based regularization [107], which seeks local smoothness by maximizing the pairwise similarities between nearby data points. For TTA tasks, these semi-supervised learning techniques could be easily incorporated to unsupervisedly update the pre-trained model.

2.6 Test-Time Augmentation

Data augmentation techniques [108], such as geometric transformations and color space augmentations, can create modified versions of training images that improve the robustness of deep models against unknown perturbations. Typically, data augmentation can also be employed during test time to boost prediction accuracy [109], [110], estimate uncertainty [111], and enhance robustness [112], [113]. Ten-crop testing [110] is a typical example of test-time augmentation, which obtains the final prediction via averaging predictions from ten different scaled versions of a test image. Other aggregation strategies include selective augmentation [114] and learnable aggregation weights [115]. Apart from data variation, Monte Carlo dropout [116] enables dropout within the network during testing and performs multiple forward passes with the same input data, to estimate the model uncertainty. Generally, test-time augmentation techniques do not explicitly consider distribution shifts but can be utilized by TTA methods.

3 SOURCE-FREE DOMAIN ADAPTATION

3.1 Problem Definition

Definition 1 (Domain). A domain \mathcal{D} is a joint distribution $p(x, y)$ defined on the input-output space $\mathcal{X} \times \mathcal{Y}$, where random variables $x \in \mathcal{X}$ and $y \in \mathcal{Y}$ denote the input data and the label (output), respectively.

In a well-studied domain adaptation problem, the domain of interest is called the target domain $p_{\mathcal{T}}(x, y)$ and the domain with labeled data is called the source domain $p_{\mathcal{S}}(x, y)$. The label y can either be discrete (in a classification task) or continuous (in a regression task). Unless otherwise specified, \mathcal{Y} is a C -cardinality label set, and we usually have one labeled source domain $\mathcal{D}_{\mathcal{S}} = \{(x_1, y_1), \dots, (x_{n_s}, y_{n_s})\}$ and one unlabeled target domain $\mathcal{D}_{\mathcal{T}} = \{x_1, \dots, x_{n_t}\}$ under data distribution shifts: $\mathcal{X}_{\mathcal{S}} = \mathcal{X}_{\mathcal{T}}$, $p_{\mathcal{S}}(x) \neq p_{\mathcal{T}}(x)$, including the *covariate shift* [117] assumption ($p_{\mathcal{S}}(y|x) = p_{\mathcal{T}}(y|x)$). Typically, the unsupervised domain adaptation (UDA) paradigm aims to leverage supervised knowledge in $\mathcal{D}_{\mathcal{S}}$ to help infer the label of each target sample in $\mathcal{D}_{\mathcal{T}}$.

Chidlovskii *et al.* [118] for the first time consider performing adaptation with no access to source domain data. Specifically, they propose three scenarios for feature-based domain adaptation with: source classifier with accessible models and parameters, source classifier as a black-box model, and source class means as representatives. This new setting utilizes all the test data to adjust the classifier learned from the training data, which could be considered as a broad test-time adaptation scheme. Several methods [119], [120], [121] follow this learning mechanism and adapt the source classifier to unlabeled target features. To gain benefits from

TABLE 1
A taxonomy on SFDA methods with representative strategies.

Families	Model Rationale	Representative Strategies
pseudo-labeling	centroid-based	SHOT [7], [98], BMD [122]
	neighbor-based	NRC [123], SSNLL [124]
	complementary labels	LD [125], ATP [126]
	optimization-based	ASL [127], KUDA [128]
consistency	data variations	G-SFDA [129], APA [130]
	model variations	SFDA-UR [131], FMML [132]
	both variations	AdaContrast [133], MAPS [134]
clustering	entropy minimization	ASFA [135], 3C-GAN [12]
	mutual information	SHOT [7], [98], UMAD [136]
	explicit clustering	ISFDA [137], SDA-FAS [138]
source estimation	data generation	3C-GAN [12], DI [139]
	data translation	SFDA-IT [140], ProSFDA [141]
	data selection	SHOT++ [98], DaC [142]
	feature estimation	VDM-DA [143], CPGA [144]
self-supervision	auxiliary tasks	SHOT++ [98], StickerDA [145]

end-to-end representation learning, researchers are more interested in generalization with deep models. Such a setting without access to source data during adaptation is termed as source data-absent (free) domain adaptation [7], [98], model adaptation [12], and source-free domain adaptation [11], respectively. For the sake of simplicity, we utilize the term *source-free domain adaptation* and give a unified definition.

Definition 2 (Source-free Domain Adaptation, SFDA). Given a well-trained classifier $f_S : \mathcal{X}_S \rightarrow \mathcal{Y}_S$ on the source domain \mathcal{D}_S and an unlabeled target domain \mathcal{D}_T , *source-free domain adaptation* aims to leverage the labeled knowledge implied in f_S to infer labels of all the samples in \mathcal{D}_T , in a transductive learning [26] manner. Note that, all test data (target data) are required to be seen during adaptation.

So far as we know, the term *source-free domain adaptation* is first proposed by Nelakurthi *et al.* [61], where they try to leverage the noisy predictions of an off-the-shelf classifier and a few labeled examples from the target domain, in order to obtain better predictions for all the unlabeled target samples. The definition here covers [61] as a special case, where the classifier f_S is not accessible but provides the predictions of target data $\{f_S(x)|x \in \mathcal{D}_T\}$.

3.2 Taxonomy on SFDA Algorithms

3.2.1 Pseudo-labeling

To adapt a pre-trained model to an unlabeled target domain, a majority of SFDA methods take inspiration from the semi-supervised learning (SSL) field [146] and employ various prevalent SSL techniques tailored for unlabeled data during adaptation. A simple yet effective technique, pseudo-labeling [101], aims to assign a class label $\hat{y} \in \mathbb{R}^C$ for each unlabeled sample x in \mathcal{X}_t and optimize the following supervised learning objective to guide the learning process,

$$\min_{\theta} \mathbb{E}_{\{x, \hat{y}\} \in \mathcal{D}_t} w_{pl}(x) \cdot d_{pl}(\hat{y}, p(y|x; \theta)), \quad (1)$$

where $w_{pl}(x) \in \mathbb{R}$ denotes the weight associated with each pseudo-labeled sample $\{x, \hat{y}\}$, and $d_{pl}(\cdot)$ denotes the divergence between the predicted label probability distribution and the pseudo label probability \hat{y} , e.g., $-\sum_c \hat{y}_c \log[p(y|x; \theta)]_c$ if using the cross entropy as the

divergence measure. Since the pseudo labels of target data are inevitably inaccurate under domain shift, there exist three different solutions: (1) improving the quality of pseudo labels via denoising; (2) filtering out inaccurate pseudo labels with $w_{pl}(\cdot)$; (3) developing a robust divergence measure $d_{pl}(\cdot, \cdot)$ for pseudo-labeling.

Following the classic pseudo-labeling work [101], many SFDA methods directly obtain the pseudo label \hat{y}_t as the one-hot encoding of the class that has the maximum predicted probability [147], [148], [149], [150], [151]. To reduce the effects of noisy pseudo labels based on the argmax operation, most of these methods develop various filtering mechanisms to consider only reliable pseudo labels during pseudo-labeling, e.g., maximum prediction probability [151], [152], [153], [154], [155], self-entropy [156], and consistency score [150], [153], [157]. In the following, we review not only different types of denoised pseudo labels but also different forms of the pseudo-labeling objective.

Centroid-based pseudo labels. Inspired by a classic self-supervised approach, DeepCluster [90], SHOT [7] and SHOT++ [98] resort to target-specific clustering for denoising the pseudo labels. The key idea is to obtain target-specific class centroids based on the network predictions and the target features and then derive the unbiased pseudo labels via the nearest centroid classifier. Formally, the class centroids and pseudo labels are updated as follows,

$$\begin{cases} m_c = \sum_x [p_{\theta}(y_c|x) \cdot g(x)] / \sum_x p_{\theta}(y_c|x), c = 1, \dots, C, \\ \hat{y} = \arg \min_c d(g(x), m_c), \forall x \in \mathcal{D}_t, \end{cases} \quad (2)$$

where $p_{\theta}(y_c|x) = [p(y|x; \theta)]_c$ denotes the probability associated with the c -th class, and $g(x)$ denotes the feature of input x . m_c denotes the c -th class centroid, and $d(\cdot, \cdot)$ denotes the cosine distance function. Note that, $p_{\theta}(y_c|x)$ is initialized by the soft network prediction and further updated by the one-hot encoding of the new pseudo-label \hat{y} . Besides, CDCL [158] directly performs K-means clustering in the target feature space with the cluster initialized by the source class prototypes. As class centroids always contain robust discriminative information and meanwhile weaken the category imbalance problem, this label refinery is prevalent in follow-up SFDA studies [142], [144], [159], [160], [161], [162], [163], [164], [165], [166], [167], [168], [169], [170], [171], [172].

To construct more accurate centroids, some methods (e.g., PPDA [173], PDA-RoG [174], and Twofer [175]) propose to identify confident samples to obtain the class centroids, and SCLM [176] further integrates the enhanced centroids with the coarse centroids. By leveraging more discriminative representations, TransDA [160] maintains a mean teacher model [106] as guidance, while Co-learn [177] exploits an additional pre-trained feature extractor. Besides, BMD [122] and PCSR [178] conjecture that a coarse centroid cannot effectively represent ambiguous data, and instead utilize K-means clustering to discover multiple prototypes for each class. In addition, ADV-M [179] directly employs the Mahalanobis distance to measure the similarity between samples and centroids, while CoWA-JMDS [180] and PDA-RoG [174] perform Gaussian Mixture Modeling (GMM) in the target feature space to obtain the log-likelihood

and pseudo label of each sample. Except for hard pseudo labels, some recent works [160], [181], [182] explore soft pseudo labels based on the class centroids, e.g., $\hat{y}_c = \frac{\exp(-d(g(x), m_c)/\tau)}{\sum_c \exp(-d(g(x), m_c)/\tau)}$, where τ denotes the temperature. In contrast to offline clustering in Eq. (2), BMD [122] and DMAPL [183] develop a dynamic clustering strategy that uses an exponential moving average (EMA) to accumulate the class centroids in mini-batches.

Neighbor-based pseudo labels. Based on the assumption of local smoothness among neighbors, another popular label denoising strategy generates the pseudo-label by incorporating the predictions of its neighbors [124], [133], [184], [185], [186], [187]. In particular, SSNLL [124] conducts K-means clustering in the target domain and then aggregates predictions of its neighbors within the same cluster, and SCLM [176] fuses the predictions of its two nearest neighbors and itself. DIPE [184] diminishes label ambiguity by correcting the pseudo label to the label of the majority of its neighbors. Moreover, one line of work [147], [185], [188] constructs an anchor set comprising only highly confident target samples, and a greedy chain-search strategy is proposed to find its nearest neighbor in the anchor set [185], [188]. While SFDA-APM [147] employs a point-to-set distance function to generate the pseudo labels, CAiDA [188] interpolates its nearest anchor to the target feature and uses the prediction of the synthetic feature instead.

Inspired by neighborhood aggregation [189], one line of work [133], [156], [186], [187], [190], [191] maintains a memory bank storing both features and predictions of the target data $\{g(x_i), q_i\}_{i=1}^{n_t}$, allowing online refinement of pseudo labels. Typically, the refined pseudo label is obtained through

$$\hat{p}_i = \frac{1}{m} \sum_{j \in \mathcal{N}_i} q_j, \quad (3)$$

where \mathcal{N}_i denotes the indices of m nearest neighbors of $g(x_i)$ in the memory bank. Specifically, ProxyMix [187] sharpens the network output \bar{p} with the class frequency to avoid class imbalance and ambiguity, while RS2L [156] and NRC [123] devise different weighting schemes for neighbors during aggregation, respectively. Instead of the soft pseudo label \hat{p} , AdaContrast [133] and DePT [190] utilize the hard pseudo label with the argmax operation.

Complementary pseudo labels. Motivated by the idea of negative learning [192], PR-SFDA [193] randomly chooses a label from the set $\{1, \dots, C\} \setminus \{\hat{y}_i\}$ as the complementary label \bar{y}_i and thus optimizes the following loss function,

$$\min_{\theta} - \sum_{i=1}^{n_t} \sum_{c=1}^C \mathbb{1}(\bar{y}_i = c) \log(1 - p_{\theta}(y_c | x_i)), \quad (4)$$

where \hat{y}_i denotes the inferred hard pseudo label. We term \bar{y} as a negative pseudo label that indicates the given input does not belong to this label. Note that, the probability of correctness is $\frac{C-1}{C}$ for the complementary label \bar{y}_i , providing correct information even from wrong labels \hat{y}_i . LD [125] further develops a heuristic strategy to randomly select an informative complementary label with medium prediction scores, followed by CST [194]. Besides, NEL [195] and PLUE [191] randomly pick up multiple complementary labels except for the inferred pseudo label and optimizes the multi-class variant of Eq. (4). Further, some alternative methods [126], [196], [197] generate multiple complementary labels

according to a pre-defined threshold on the prediction scores.

Optimization-based pseudo labels. By leveraging the prior knowledge of the target label distribution like class balance [198], a line of SFDA methods [125], [131], [199], [200] varies the threshold for each class so that a certain proportion of points per class are selected. Such a strategy would avoid the ‘winner-takes-all’ dilemma where pseudo labels come from several major categories and deteriorate the following training process. In addition, ASL [127] directly imposes the equi-partition constraint on the pseudo labels \hat{p}_i and solves the following optimization problem,

$$\begin{aligned} \min_{\hat{p}_i} & - \sum_i \sum_c \hat{p}_{ic} \log p_{\theta}(y_c | x_i) + \lambda \sum_i \sum_c \hat{p}_{ic} \log \hat{p}_{ic}, \\ \text{s.t. } & \forall i, c: \hat{p}_{ic} \in [0, 1], \sum_c \hat{p}_{ic} = 1, \sum_i \hat{p}_{ic} = \frac{n_t}{C}. \end{aligned} \quad (5)$$

Similarly, IterNLL [201] provides a closed-form solution of $\{\hat{p}\}$ under the uniform prior assumption, and NOTELA [202] exploits the Laplacian regularizer to promote similar pseudo-labels for nearby points in the feature space. Later, KUDA [128] comes up with the hard constraint $\hat{p}_{ic} \in \{0, 1\}$ and solves the zero-one programming problem. Moreover, U-D4R [203] estimates a joint distribution matrix between the observed and latent labels to correct the pseudo labels.

Ensemble-based pseudo labels. Rather than rely on one single noisy pseudo label, ADV-M [179] and ISFDA [137] generate a secondary pseudo label to aid the primary one. Besides, ASL [127] and C-SFDA [204] adopt a weighted average of predictions under multiple random data augmentation, while ST3D [205] and ELR [206] ensemble historical predictions from previous training epochs. NEL [195] further aggregates the logits under different data augmentation and trained models at the same time. Inspired by a classic semi-supervised learning method [105], a few SFDA methods [207], [208] maintain an EMA of predictions at different time steps as pseudo labels. By contrast, some methods [204], [209], [210] maintain a mean teacher model [106] that generates pseudo labels for the current student network. Additionally, other methods try to generate the pseudo label based on predictions from different models, e.g., multiple source models [207], [211], [212], a multi-head classifier [213], [214], and models from both source and target domains [157]. In particular, SFDA-VS [215] follows the idea of Monte Carlo (MC) dropout [116] and obtains the final prediction through multiple stochastic forward passes. Co-learn [177] further trains an auxiliary target model in another backbone and produces pseudo labels based on the consistency of dual models and confidence filtering.

Another line of ensemble-based SFDA methods [186], [216], [217] tries to integrate predictions from different labeling criteria using a weighted average. For example, e-SHOT-CE [186] utilizes both centroid-based and neighbor-based pseudo labels, and PLA-DAP [216] combines centroid-based pseudo labels and the original network outputs. CSFA [217] further develops a switching strategy to choose between centroid-based pseudo labels or neighbor-based pseudo labels. Further, OnDA [208] integrates the centroid-based pseudo labels and the temporal ensemble of network predictions using a dot product operation. Apart from the weighting scheme, other approaches [144], [183], [184], [188], [218]

explore different labeling criteria in a cascade manner. For instance, several methods (e.g., CAiDA [188], DIPE [184], and Twofer [175]) employ the neighbor-based labeling criterion based on the centroid-based pseudo labels. CPGA [144] and DMAPL [183] also adopt the centroid-based pseudo labels but employ a self-ensembling technique to obtain the final predictions over historical predictions. Besides, CoNMix [218] further calculates a cluster consensus matrix and utilizes it in the temporal ensemble over centroid-based pseudo labels.

Learning with pseudo labels. Different robust divergence measures d_{pl} have been employed in existing pseudo-labeling-based SFDA methods. Generally, most of them employed the standard cross-entropy loss for all the target samples with hard pseudo labels [7], [98], [127], [149], [154], [158], [176], [178], [216], [218], [219] or soft pseudo labels [165], [220]. Note that, several methods [156], [187], [221] convert hard pseudo labels into soft pseudo labels using the label smoothing trick [222]. As pseudo labels are noisy, many SFDA methods incorporate an instance-specific weighting scheme into the standard cross-entropy loss, including hard weights [125], [131], [147], [150], [155], [157], [173], [223], and soft weights [180], [199], [214], [215], [224], [225], [226]. Besides, AUGCO [153] considers the class-specific weight in the cross-entropy loss to mitigate the label imbalance. In addition to the cross-entropy loss, alternative choices include the Kullback–Leibler divergence [156], the generalized cross entropy [227], the inner product distance between the pseudo label and the prediction [123], [144], and a new discrepancy measure $\log(1 - \hat{y}^T p(y|x; \theta))$ [206]. Inspired by a classic noisy label learning approach [228], BMD [122] and OnDA [208] employ the symmetric cross-entropy loss to guide the self-labeling process. SFAD [168] further develops a normalized symmetric cross-entropy variant by incorporating the normalized cross-entropy loss [229], and CATTAn [230] exploits the negative log-likelihood ratio between correct and competing classes [231].

3.2.2 Consistency Training

As a prevailing strategy in recent semi-supervised learning literature [99], [232], consistency regularization is primarily built on the smoothness assumption or the manifold assumption, which aims to enforce consistent network predictions or features under variations in the input data space or the model parameter space. Besides, another line of consistency training methods tries to match the statistics of different domains even without the source data. In the following, we review different consistency regularizations under data and model variations together with other consistency-based distribution matching methods.

Consistency under data variations. Benefiting from advanced data augmentation techniques (e.g., Cutout [233], RandAugment [234]), several prominent semi-supervised learning methods [104], [235] unleash the power of consistency regularization over unlabeled data that can be effortlessly adopted in SFDA approaches. Formally, an exemplar of consistency regularization [104] is expressed as:

$$\mathcal{L}_{fm}^{con} = \frac{1}{n_t} \sum_{i=1}^{n_t} \text{CE}(p_{\hat{\theta}}(y|x_i), p_{\theta}(y|\hat{x}_i)), \quad (6)$$

where $p_{\theta}(y|x_i) = p(y|x_i; \theta)$, and $\text{CE}(\cdot, \cdot)$ refers to cross-entropy between two distributions. Besides, \hat{x}_i represents the variant of x_i under another augmentation transformation, and $\hat{\theta}$ is a fixed copy of current network parameters θ . Another representative consistency regularization is virtual adversarial training (VAT) [102] that devises a smoothness constraint as follows,

$$\mathcal{L}_{vat}^{con} = \frac{1}{n_t} \sum_{i=1}^{n_t} \max_{\|\Delta_i\| \leq \epsilon} [\text{KL}(p_{\hat{\theta}}(y|x_i) || p_{\theta}(y|x_i + \Delta_i))], \quad (7)$$

where Δ_i is a perturbation that disperses the prediction most within an intensity range of ϵ for the target data x_i , and KL denotes the Kullback–Leibler divergence.

ATP [126] and ALT [236] directly employ the same consistency regularization in Eq. (6), while other SFDA methods [133], [142], [181], [190], [218], [237] replace $p_{\hat{\theta}}(y|x_i)$ with hard pseudo labels for target data under weak augmentation, followed by a cross-entropy loss for target data under strong augmentation. Note that, many of these hard labels are obtained using label denoising techniques mentioned earlier. Apart from strong augmentations, SFDA-ST [223] and RuST [219] seek geometric consistency by switching the order of the model and spatial transforms (e.g., image mirroring and rotation). FMML [132] and FTA-FDA [149] resort to Fourier-based data augmentation [238], while ProSFDA [141] and SFDA-FSM [239] need to learn the domain translation module at first. In addition to the output-level consistency, FAUST [181] and ProSFDA [141] seek feature-level consistency under different augmentations, and TeST [240] even introduce a flexible mapping network to match features under two different augmentations. On the contrary, OSHT [241] maximizes the mutual information between predictions of two different transformed inputs to retain the semantic information as much as possible.

Following the objective in Eq. (7), another line of SFDA methods [12], [127], [242] attempts to encourage consistency between target samples with their data-level neighbors, and APA [130] learns the neighbors in the feature space. Instead of generating the most divergent neighbor $x_i + \Delta_i$ according to the predictions, JN [162] devises a Jacobian norm regularization to control the smoothness in the neighborhood of the target sample. Further, G-SFDA [129] discovers multiple neighbors from a memory bank and minimizes their inner product distances over the predictions. Moreover, Mixup [243] performs linear interpolations on two inputs and their corresponding labels, which can be treated as seeking consistency under data variation [163], [178], [180], [207], [218].

Consistency under model variations. Reducing model uncertainty [116] is also beneficial for learning robust features for SFDA tasks, on top of uncertainty measured with input change. Following a classic uncertainty estimation framework, MC dropout [116], FAUST [181] activates dropout in the model and performs multiple stochastic forward passes to estimate the epistemic uncertainty. SFDA-UR [131] and RuST [219] append multiple extra dropout layers behind the feature encoder and minimize the mean squared error between predictions as uncertainty. Further, several methods [135], [172], [219] add different perturbations to the intermediate features to promote predictive consistency. In addition, FMML [132] offers another type of model variation

by network slimming and sought predictive consistency across different networks.

Another model-based consistency regularization requires the existence of the source and target models and thus minimizes the difference across different models, *e.g.*, feature-level discrepancy [244] and output-level discrepancy [207], [215], [240], [245], [246], [247]. Furthermore, the mean teacher framework [106] is also utilized to form a strong teacher model and a learnable student model, where the teacher and the student share the same architecture, and the weights of the teacher model θ_{tea} is gradually updated by,

$$\theta_{tea} = (1 - \eta)\theta_{tea} + \eta\theta \quad (8)$$

where θ denotes the weights of the student model, and η is the momentum coefficient. Therefore, the mean teacher model is regarded as a temporal ensemble of student models with more accurate predictions. Anat-SFDA [248] minimizes the L_1 distance between predictions of the teacher and student models, and TransDA [160] exploits the features extracted by the teacher model to obtain better pseudo labels for supervising the student model. Moreover, UAMT [209] even incorporates MC dropout into the mean teacher model for reliable pseudo labels along with the sample variances. Similarly, MoTE [249] obtains multiple mean teacher models individually and seeks predictive consistency for each teacher-student pair. In reality, a small number of SFDA methods [214], [250], [251] also consider the multi-head classifier and encourage predictions by different heads to be consistent for robust features. TAN [252] introduces an extra module to align feature encoders across domains with the source classifier fixed.

Consistency under data & model variations. In reality, data variation and model variation could be integrated into a unified framework. For example, the mean teacher framework [106] is enhanced by blending strong data augmentation techniques, and the discrepancy between predictions of the student and teacher models is minimized as follows,

$$\mathcal{L}_{mt}^{con} = \mathbb{E}_{x \in \mathcal{D}_t} d_{mt}(p(y|x, \theta), p(y|\tau(x), \theta_{tea})), \quad (9)$$

where $\tau(\cdot)$ denotes the strong data augmentation, and d_{mt} denotes the divergence measure, *e.g.*, the mean squared error [253], [254], the Kullback-Leibler divergence [138], [157], and the cross-entropy loss [124], [190], [224], [255], [256], [257], [258], [259]. SFDA-DML [170] proposes a mutual learning framework that involves two target networks and optimizes one target network with pseudo labels from the mean teacher model of the other network under different data variations. Besides, several methods [4], [199], [260], [261], [262], [263] attempt to extract useful information from the teacher and employ task-specific loss functions to seek consistency. Apart from the output-level consistency, TT-SFDA [260] matches the features extracted by different models with the MSE distance, while AdaContrast [133] and PLUE [191] learn semantically consistent features like [92].

Instead of strong data augmentations, LODS [262] and SFIT [157] use the style transferred image instead, MAPS [134] considers spatial transforms, and SMT [253] elaborates the domain-specific perturbation by averaging the target images. In addition to model variations of the mean teacher scheme, OnTA [264] distills knowledge from the source model to the target model, while GarDA [265] and HCL

[199] promote consistency among the current model and historical model in feature space and prediction space, respectively. To selectively distill knowledge from the teacher, MAPS [134] also develops a sample filtering strategy to only consider confident samples during consistency training.

Miscellaneous consistency regularizations. To prevent excessive deviation from the original source model, a flexible strategy is adopted by a few methods [12], [127], [216], [242], [266], [267] by establishing a parameter-based regularization term $\|\theta_s - \theta\|_2^2$, where θ_s is the fixed source weight. Further, DTAC [197] utilizes a robust L_1 distance instead. Another line of research focuses on matching the batch normalization (BN) statistics (*i.e.*, the mean and the variance) across models with different measures, *e.g.*, the Kullback-Leibler divergence [268], and the MSE error [197], [253], [269], [270], whereas OSUDA [271] encourages the learned scaling and shifting parameters in BN layers to be consistent. Similarly, an explicit feature-level regularization [272], [273] is further developed that matches the first and second-order moments of features of different domains.

Remarks. As for the network architecture in the target domain, a unique design termed dual-classifier structure is utilized to seek robust domain-invariant representations. For example, BAIT [274] introduces an extra C -dimensional classifier to the source model, forming a dual-classifier model with a shared feature encoder. During adaptation in the target domain, the shared feature encoder and the new classifier are trained with the classifier from the source domain head fixed. Such a training scheme has also been utilized by many SFDA methods [131], [135], [159], [184], [217], [221], [267], [275], [276] through modeling the consistency between different classifiers. Besides, SFDA-APM [147] develops a self-training framework that optimizes the shared feature encoder and two classification heads with different pseudo-labeling losses, respectively. By contrast, CdKD-TSML [164] puts forwards two feature encoders with the shared classifier and performs mutual knowledge distillation across models to enhance the generalization.

3.2.3 Clustering-based Training

Except for the pseudo-labeling paradigm, one assumption explicitly or implicitly made in virtually all semi-supervised learning algorithms is the cluster assumption [146], [277], [278], which states that the decision boundary should not cross high-density regions, but instead lie in low-density regions. Motivated by this, another popular family of SFDA approaches favors low-density separation through reducing the uncertainty of the target network predictions [7], [12], [279] or encouraging clustering over the target features [137], [144].

Entropy minimization. To encourage confident predictions for unlabeled target data, ASFA [135] borrows robust measures from information theory and minimizes the following α -Tsallis entropy [280],

$$\mathcal{L}_{tsa} = \frac{1}{n_t} \sum_{i=1}^{n_t} \frac{1}{\alpha - 1} \left[1 - \sum_{c=1}^C p_{\theta}(y_c|x_i)^{\alpha} \right], \quad (10)$$

where $\alpha > 0$ is called the entropic index. Note that, when α approaches 1, the Tsallis entropy exactly recovers the standard Shannon entropy in $\mathcal{H}(p_{\theta}(y|x_i)) =$

$\sum_c p_\theta(y_c|x_i) \log p_\theta(y_c|x_i)$. In practice, the conditional Shannon entropy $\mathcal{H}(p_\theta(y|x))$ has been widely used in SFDA methods [12], [125], [127], [131], [163], [177], [181], [213], [225], [240], [242], [246], [271], [281], [282], [283]. Besides, there exist many variants of standard entropy minimization. For example, SS-SFDA [154] ranks the target images first and minimizes the entropy from confident data to less-confident data, SFDA-VS [215] develops a nonlinear weighted entropy minimization loss that weighed more on the low-entropy samples. Moreover, TT-SFDA [260] focuses on the entropy of the ensemble predictions under multiple augmentations, and SFDA-GEM [267] accommodates the Gumbel-softmax trick by modifying the predictions with additive Gumbel noise to improve the robustness.

When α equals 2, the Tsallis entropy in Eq. (10) corresponds to the maximum squares loss [33], $\sum_c p_\theta(y_c|x_i)^2$. Compared with the Shannon entropy, the gradient of the maximum squares loss increases linearly, preventing easy samples from dominating the training process in the high probability area. Also, another line of SFDA methods [193], [211], [212], [218], [284] has employed the maximum squares loss. Built on this, Batch Nuclear-norm Maximization (BNM) [34] approximates the prediction diversity using the matrix rank, which is further adopted in other SFDA methods [168], [237], [275]. Additionally, another line of SFDA methods [279], [285] pays attention to the class confusion matrix, which minimizes the inter-class confusion to ensure no samples are ambiguously classified into two classes at the same time. RNA++ [285] also discusses the usage of complement entropy that flattens the predicted probabilities of classes except for the pseudo label.

Mutual information maximization. Another favorable clustering-based regularization is mutual information maximization, which tries to maximize the mutual information [286], [287] between inputs and the discrete labels as follows,

$$\begin{aligned} \max_{\theta} \mathcal{I}(\mathcal{X}_t, \hat{\mathcal{Y}}_t) &= \mathcal{H}(\hat{\mathcal{Y}}_t) - \mathcal{H}(\hat{\mathcal{Y}}_t|\mathcal{X}_t) \\ &= -\sum_{c=1}^C \bar{p}_\theta(y_c) \log \bar{p}_\theta(y_c) + \frac{1}{n_t} \sum_{i=1}^{n_t} \sum_{c=1}^C p_\theta(y_c|x_i) \log p_\theta(y_c|x_i), \end{aligned} \quad (11)$$

where $\bar{p}_\theta(y_c) = \frac{1}{n_t} \sum_i p_\theta(y_c|x_i)$ denotes the c -th element in the estimated class label distribution. Intuitively, increasing the additional diversity term $\mathcal{H}(\hat{\mathcal{Y}}_t)$ encourages the target labels to be uniformly distributed, circumventing the degenerate solution where every sample is assigned to the same class. Such a regularization is first introduced in SHOT [7] and SHOT++ [98] for image classification and then employed in plenty of SFDA methods [149], [160], [161], [162], [164], [166], [167], [170], [171], [178], [182], [184], [191], [226], [250], [264], [268], [288], [289], [290]. Beyond visual classification tasks, it has also been applied in other source-free applications [169], [186], [269], [291], [292], [293], [294], [295]. Instead of using the network prediction $p_\theta(y|x)$, a few methods [165], [176], [185] utilize the ensemble prediction based on its neighbors for mutual information maximization. DaC [142] and U-SFAN [296] introduce a balancing parameter between two terms in Eq. (11) to increase the flexibility. In particular, U-SFAN [296] develops an uncertainty-guided entropy minimization loss by focusing more on the

low-entropy predictions, whereas ATP [126] encompasses the instance-wise uncertainty in both terms of Eq. (11). VMP [297] further provides a probabilistic framework based on Bayesian neural networks and integrated mutual information into the likelihood function.

Note that the diversity term can be rewritten as $\mathcal{H}(\hat{\mathcal{Y}}_t) = -\text{KL}(\bar{p}_\theta(y)|\mathcal{U}) + \log C$, where $\bar{p}_\theta(y)$ is the average label distribution in the target domain, and \mathcal{U} denotes a C -dimensional uniform vector. This term alone has also been used in many SFDA methods [123], [133], [145], [157], [159], [208], [208], [221], [230] to avoid class collapse. To better guide the learning process, previous works [298], [299] modify the mutual information regularization by substituting a reference class-ratio distribution in place of \mathcal{U} . Different from AdaMI [282] that leverages the target class ratio as a prior, UMAD [136] utilizes the flattened label distribution within a mini-batch instead to mitigate the class imbalance problem, and AUGCO [153] maintains the moving average of the predictions as the reference distribution.

Explicit clustering. On top of these clustering-promoting output-level regularizations, other clustering-based methods [137], [138], [161] explicitly perform feature-level clustering to encourage more discriminative decision boundaries. Specifically, both ISFDA [137] and LCCL [161] rely on the pseudo labels and pull closer features from the same class while pushing farther those from different classes. SDA-FAS [138] leverages the prototypical contrastive alignment scheme to match target features and features of the source prototypes. Inspired by neighborhood clustering [300], CPGA [144] and StickerDA [145] calculate the normalized instance-to-instance similarity with a memory bank $\{\bar{g}(x_i)\}_{i=1}^{n_t}$ and thus minimize the following entropy,

$$\mathcal{L}_{nc} = -\frac{1}{n_t} \sum_{i=1}^{n_t} \sum_{j \neq i} s_{ij} \log s_{ij}, \quad (12)$$

where $s_{ij} = \frac{\exp(g(x_i)^T \bar{g}(x_j)/\tau)}{\sum_{k \neq i} \exp(g(x_i)^T \bar{g}(x_k)/\tau)}$ represents the normalized similarity between the i -th target feature and the j -th feature stored in the memory bank, and τ denotes the temperature parameter. Later, A²Net [276] and CSFA [217] integrate the pseudo labels with the normalized similarity to seek semantic clustering. Several remaining methods [236], [301], [302] develop a simplified variant to learn class-discriminative features with local clustering in the target dataset, respectively. In addition to the feature-level clustering strategies, DECDA [303] takes inspiration from the classic deep clustering method [304] and learns feature representations and cluster assignments simultaneously.

3.2.4 Source Distribution Estimation

Another favored family of SFDA approaches compensates for the absence of source data by inferring data from the pre-trained model, which turns the challenging SFDA problem into a well-studied DA problem. Existing distribution estimation methods could be divided into three categories: data generation from random noises [12], [305], [306], data translation from target samples [157], [288], [307], and data selection from the target domain [98], [187], [274].

Data generation. To produce valid target-style training samples, 3C-GAN [12] introduces a data generator $G(\cdot; \theta_G)$

conditioned on randomly sampled labels along with a binary discriminator $D(\cdot; \theta_D)$ and aims to optimize a similar objective to conditional GAN [308] in the following,

$$\min_{\theta_G} \max_{\theta_D} \mathbb{E}_{x_t \in \mathcal{X}_t} [\log D(x_t)] + \mathbb{E}_{y_t, z} [\log(1 - D(G(y_t, z)))] - \lambda_s \mathbb{E}_{y_t, z} \sum_c \mathbb{1}(y_t = c) \log p(y_c | G(y_t, z), \theta), \quad (13)$$

where z is a random noise vector, y_t is a pre-defined label, $\lambda_s > 0$ is a balancing parameter, and θ denotes the parameters of the pre-trained prediction model. By alternately optimizing θ_G and θ_D , the resulting class conditional generator G is able to produce multiple surrogate labeled source instances for the following domain alignment step, *i.e.*, $\mathcal{D}_g = \{x_i, y_i\}_{i=1}^{n_g}$, where $x_i = G(y_i, z)$ and n_g is the number of generated samples. Different from SDG-CMT [242] that employs the same objective for medical data, PLR [305] ignores the last term in Eq. (13) to infer diverse target-like samples. SDDA [306] additionally maximizes the log-likelihood of generated data x_g and employs two different domain discriminators, *i.e.*, a data-level GAN discriminator and a feature-level domain discriminator. SFDA-ADT [266] integrates multiple discriminators together to match the data-label joint distribution between \mathcal{D}_g and \mathcal{D}_t .

In addition to adversarial training, DI [139] performs Dirichlet modeling with the source class similarity matrix and directly optimizes the noisy input to match its network output with the sampled softmax vector q as follows,

$$x_g = \arg \min_x \text{CE}(q, p_\theta(y|x)) \quad (14)$$

is also known as data impression of the source domain. SDG-MA [309] employs the relationship preserving loss [310] to ensure style consistency during learning the label-conditional data generator. Moreover, SPGM [311] first estimates the target distribution using Gaussian Mixture Models (GMM) and then constrains the generated data to be derived from the target distribution.

Motivated by recent advances in data-free knowledge distillation [23], [82], SFDA-KTMA [284] exploits the moving average statistics of activations stored in BN layers of the pre-trained source model and imposes the following BN matching constraint on the generator,

$$\mathcal{L}_{bn} = \sum_l \sum_i \|\mu_{g,l}^{(i)} - \mu_{s,l}^{(i)}\|_2 + \|\delta_{g,l}^{(i)2} - \delta_{s,l}^{(i)2}\|_2, \quad (15)$$

where B is the size of a mini-batch, $\mu_{s,l}^{(i)}$ and $\delta_{s,l}^{(i)2}$ represent the corresponding running mean and variance stored in the source model, and $\mu_{g,l}^{(i)} = \frac{1}{B} \sum_z f_l^{(i)}(x_g)$ and $\delta_{g,l}^{(i)2} = \frac{1}{B} \sum_z (f_l^{(i)}(x_g) - \mu_{g,l}^{(i)})^2$ denote the batch-wise mean and variance estimates of the i -th feature channel at the l -th layer for synthetic data from the generator, respectively. As indicated in [312], matching the BN statistics facilitates forcing the generated data equipped with the source style. SFDA-FSM [239] further minimizes the L_2 -norm difference between intermediate features (*a.k.a.*, the content loss [310]) to preserve the content knowledge of the target domain.

Data translation. Under the assumption that the style information is embedded in low-frequency components, FTA-FDA [149] relies on spectrum mixup [238] across domains to synthesize samples through inverse Fourier Transform.

SSFT-SSD [288] initializes x_g as $x_t \in \mathcal{X}_t$ and directly performs optimization on the input space with the gradient of the L_2 -norm regularized cross-entropy loss being zero. Note that these two methods do not include any additional modules in the translation process. On the contrary, SFDA-TN [313] optimizes a learnable data transformation network that maps target data to the source domain so that the maximum class probability is maximized. Inspired by the success of prompt learning in computer vision [314], ProSFDA [141] adds a learnable image perturbation to all target data so that the BN statistics can be aligned with those stored in the source model. Hereafter, the style-transferred image is generated using spectrum mixup [238] between the target image and its perturbed version.

Another line of data translation methods [140], [157], [307] explicitly introduces an additional module \mathcal{A} to transfer target data to source-like style. In particular, SFDA-IT [140] optimizes the translator with the style matching loss in Eq. (15) as well as the feature-level content loss, with the source model frozen. In addition, SFDA-IT [140] further employs entropy minimization over the fixed source model to promote semantic consistency. To improve the performance of style transfer, SFIT [157] further develops a variant of the style reconstruction loss [310] as follows,

$$\mathcal{L}_{style} = \|g(x)g(x)^T - g(\mathcal{A}(x))g(\mathcal{A}(x))^T\|_2, \quad (16)$$

where $g(x) \in \mathcal{R}^{n_c \times HW}$ denotes the reshaped feature map, and H, W and n_c represent the feature map height, width, and the number of channels, respectively. The channel-wise self correlations $g(x)g(x)^T$ are also known as the Gram matrix. Meanwhile, SFIT [157] maintains the relationship of outputs between different networks. GDA [307] also relies on BN-based style matching and entropy minimization but further enforces the phase consistency and the feature-level consistency between the original image and the stylized image to preserve the semantic content. By contrast, AOS [315] replaces the first convolution block in the source model with a learnable encoder to ensure that BN statistics at the remaining layers match those stored statistics.

Data selection. Besides the synthesized source samples with data generation or data translation, another family of SFDA methods [98], [124], [126], [187], [274], [316] selects source-like samples from the target domain as surrogate source data, greatly reducing computational costs. Typically, the whole target domain is divided into two splits, *i.e.*, a labeled subset $\hat{\mathcal{X}}_{tl}$ and an unlabeled subset $\hat{\mathcal{X}}_{tu}$, and the labeled subset acts as the inaccessible source domain. Based on network outputs of the adapted model in the target domain, SHOT++ [98] makes the first attempt towards data selection by selecting low-entropy samples in each class for an extra intra-domain alignment step. Such an adapt-and-divide strategy has been adopted in later works [126], [215], [284] where the ratio or the number of selected samples per class is always kept same to prevent severe class imbalance. Besides the entropy criterion, DaC [142] utilizes the maximum softmax probability, and A2SFOD [257] exploits the predictive variance based on MC dropout. Furthermore, BETA [316] constructs a two-component GMM over all the target features to separate the confident subset $\hat{\mathcal{X}}_{tl}$ from the less confident subset $\hat{\mathcal{X}}_{tu}$.

Instead of the adapted target model, a few approaches [159], [183], [187], [292], [317], [318], [319] utilize the source model to partition the target domain before the following intra-domain adaptation step. For each class separately, PS [159] and MTRAN [292] select the low-entropy sample, ProxyMix [187] leverages the distance from target features to source class prototypes, and DMAPL [183] and SAB [317] adopt the maximum prediction probability. With multiple pre-trained source models being available, LSFT [318] computes the variance of different target predictions, while MSFDA-SPL [319] obtains the confidence from a weighted average prediction. To simulate the source domain more accurately, PS [159] and MTRAN [292] further apply the mixup augmentation technique after the dataset partition step. On the contrary, other existing methods [124], [177], [215], [217], [221], [236], [267], [274], [275], [276], [320], [321], [322] do not fix the domain partition but alternately update the domain partition and learn the target model in the adaptation step. For example, SSNLL [124] follows the small loss trick for noisy label learning and assigns samples with small loss to the labeled subset at the beginning of each epoch. Except for the global division, BAIT [274] and SFDA-KTMA [284] split each mini-batch into two sets according to the criterion of entropy ranking. Besides, D-MCD [322] and RCHC [166] employ the classifier determinacy disparity and the agreement of different self-labeling strategies for dividing the current mini-batch, respectively.

Feature estimation. In contrast to source data synthesis, previous works [143], [144], [323] provide a cost-effective alternative by simulating the source features. Intuitively, MAS³ [324] and its following works [325], [326] require learning a GMM over the source features before model adaptation, which may not hold in real-world scenarios. Instead, VDM-DA [143] constructs a proxy source domain by randomly sampling features from the following GMM,

$$p_v(z) = \sum_{c=1}^C \pi_c \mathcal{N}(z|\mu_c, \Sigma_c), \quad (17)$$

where z denotes the virtual domain feature, and $p_v(z)$ is the distribution of the virtual domain in the feature space. For each Gaussian component, $\pi_c \geq 0$ represents the mixing coefficient satisfying $\sum_c \pi_c = 1$, and μ_c, Σ_c represent the mean and the covariance matrix, respectively. Specifically, μ_c is approximated by the L_2 -normalized class prototype [327] that corresponds to the c -th row of weights in the source classifier, and a class-agnostic covariance matrix is heuristically determined by pairwise distances among different class prototypes. To incorporate relevant knowledge in the target domain, SFDA-DE [323] further selects confident pseudo-labeled target samples and re-estimates the mean and covariance over these source-like samples as an alternative. In contrast, CPGA [144] trains a prototype generator from conditional noises to generate multiple avatar feature prototypes for each class, so that class prototypes are intra-class compact and inter-class separated, which is followed by later works [289], [328].

Virtual domain alignment. Once the source distribution is estimated, it is essential to seek virtual domain alignment between the proxy source domain and the target domain for knowledge transfer. We review a variety of virtual domain alignment techniques as follows. Firstly, SHOT++ [98] and

ProxyMix [187] follow a classic semi-supervised approach, MixMatch [103], to bridge the domain gap. Secondly, SDDA [306] and PS [159] adopt the widely-used domain adversarial alignment technique [31] that is formally written as:

$$\min_{\theta_H} \max_{\theta_D} \mathbb{E}_{x_t \in \mathcal{X}_p} [\log D(H(x_t))] + \mathbb{E}_{x_t \in \mathcal{X}_t} [\log(1 - D(H(x_t)))], \quad (18)$$

where H and D respectively represent the feature encoder and the binary domain discriminator, and \mathcal{X}_p denotes the proxy source domain. Due to its simplicity, the domain adversarial training strategy has also been utilized in the following works [143], [215], [257], [266], [284]. Besides, a certain number of methods [139], [288], [324], [326] further employ advanced domain adversarial training strategies to achieve better adaptation. Thirdly, BAIT [274] leverages the maximum classifier discrepancy [329] between two classifiers' outputs in an adversarial manner to achieve feature alignment, which has been followed by [221], [267], [275], [322]. Fourthly, FTA-FDA [149] applies the maximum mean discrepancy (MMD) [330] to reduce the difference of features across domains, and its improved variants like conditional MMD and contrastive domain discrepancy have also been used in [142], [317], [318], [320], [323]. In addition, features from different domains could be further aligned through contrastive learning between source prototypes and target samples [142], [144], [289], [328]. To further model the instance-level alignment, MTRAN [292] reduces the difference between features from the target data and its corresponding variant in the virtual source domain.

3.2.5 Self-supervised Learning

Self-supervised learning is another learning paradigm tailored to learn feature representation from unlabeled data based on auxiliary prediction tasks (pretext tasks) [87], [90], [91], [92], [93]. As mentioned above, the centroid-based pseudo labels are similar to the learning manner of DeepCluster [90]. Inspired by rotation prediction [87], SHOT++ [98] further comes up with a relative rotation prediction task and introduces a 4-way classification head in addition to the C -dimensional semantic classification head during adaptation, which has been adopted by later methods [166], [171], [217]. Besides, OnTA [264] and CluP [247] exploit the self-supervised learning frameworks [91], [92] for learning discriminative features as initialization, respectively. TTT++ [272] learns an extra self-supervised branch using contrastive learning [93] in the source model, which facilitates the adaptation in the target domain with the same objective. Recently, StickerDA [145] designs three self-supervision objectives (*i.e.*, sticker location, sticker rotation, and sticker classification) and optimizes the sticker intervention-based pretext task with the auxiliary classification head in both the source training and target adaptation phases.

3.2.6 Optimization Strategy

Apart from the learning objectives categorized above, we further review a variety of optimization strategies of existing SFDA methods. Generally, most of the pseudo-labeling based SFDA methods [122], [131], [135], [143], [145], [149], [158], [160], [161], [162], [165], [166], [167], [168], [171], [184], [186], [187], [218], [251], [268], [323], [331], [332], [333] adopt the optimization strategy developed in SHOT [7]

that only updates parameters of the feature encoder with the parameters of the source classifier frozen. In such a manner, the target features are learned to approach source prototypes in the classifier under objectives like pseudo-labeling, clustering-based training, and consistency training. Note that, SHOT++ [98] and StickerDA [145] also optimize the classification head associated with the self-supervised task as well as the feature encoder, whereas TTT++ [272] fixes the additional self-supervised head during adaptation. To generate domain-invariant representations, DIPE [184] divides the parameters of the feature encoder into two groups based on the consistency of the source model and the target model, followed by a group-specific optimization rule. By contrast, SCA [334] proposes to adapt the classifier to the target domain while freezing the feature encoder.

To enable more efficient optimization, VMP [297] further provides an alternative solution by only perturbing the weights of the convolutional and FC layers in the source model. Following a seminal work [335], SFDA-ST [223] and UBNA [336] focus on updating parameters of the BN layers, mean and variance instead of weights of the entire network. Typically, updating the BN statistics is also computationally efficient as it does not require any backward propagation but calculates a weighted average of previous statistics and those in the current batch, which is also utilized in [337], [338]. To bridge the domain gap, several methods (*i.e.*, AUGCO [153], CATTAn [230] and RPL [227]) further optimize the affine parameters in the BN layers through backward propagation. Similarly, SMPT++ [4] learns the affine parameters in the Group Normalization layers.

For a majority of SFDA methods based on source distribution estimation [140], [141], [190], [239], [311], [313], the pre-trained source model during data synthesis is kept frozen to achieve implicit alignment. For instance, SFDA-IT [140] only learns the domain translator in the target domain, and ProSFDA [141] and DePT [190] learn the visual prompts in the input space, respectively. To improve the quality of data estimation, the source model could be replaced by a learnable target model [12], [242], [284], [305].

3.2.7 Beyond Vanilla Source Model

Although most SFDA methods adapt a vanilla source model to the target domain, there still exist a certain number of methods that required a customized source model [145], [250], [281], [339] or extra information about the source or target domain [149], [296], [324], [340] for model adaptation.

Customized source training. To obtain an accurate source model, a popular framework developed in SHOT [7] learns the classification network with the label smoothing trick [222]. PR-SFDA [193] adopts a class-balanced cross-entropy loss to alleviate the class imbalance in the source model, SFDA-NS [283] introduces an additional ring loss over feature vectors for both domains, and CP3Net [171] employs the focal loss [341] to addresses the data imbalance. Intuitively, these methods still work under a standard cross-entropy loss though the performance would degrade. Besides, PR-SFDA [193] employs data augmentation like color perturbation in the source domain. CPSS [200] and SI-SFDA [279] even come up with a specific data augmentation called cross-patch style swap for both domains. SFDA-mixup [339]

leverages task-specific priors to identify domain-invariant augmentation for mixup operations in both domains.

As for the network design, AdaEnt [281] learns an extra prior predictor besides the primary segmentation network in the source domain. In a similar fashion, SFDA-NS [283] introduces a shape auto-encoder to preserve the shape prior knowledge. TAN [252] individually learns the feature encoder and the classifier in the source training stage in which the encoder is optimized based on a reconstruction objective. StickerDA [145] and TTT++ [272] introduce an auxiliary self-supervised classification head in the source model, and GarDA [265] and ADV-M [179] need to learn a domain discriminator from the source domain to help the semantic model adapt to the target domain. Besides, a few methods [214], [250], [251], [322] employ the multi-head classifier learning strategy by learning multiple classification heads with one shared feature encoder in the source domain. On top of the prior-enforcing auto-encoder, SoMAN-cPAE [213] also learns a global classification head and multiple local classification heads for different data augmentations.

Extra supervision. In addition to the source model, MAS³ [324] also provides the estimated GMM of source features for the target domain. BUFR [340] further requires the marginal feature distributions in softly-binned histograms for measurement shifts. U-SFAN [296] utilizes the distribution of the Bayesian source classifier to quantify the uncertainty for more accurate adaptation. Moreover, Prototype-DA [152] and OnDA [208] need the existence of class prototypes calculated in the source domain. FTA-FDA [149] needs to retain some random spectrum maps of source data for the following data translation step, while CATTAn [230] transfers the subspace learned from source features to the target domain. Some approaches [272], [273] achieve feature alignment using different orders of moments for features in the source domain. Besides, SFDA-NS [283] leverages the learned shape auto-encoder from the source domain to match shape prior across domains. GAP [331] and ADV-M [179] perform indirect cross-domain alignment with the help of a domain discriminator learned from the source domain. In addition, CONDA [210] requires the label distribution in the source domain, while KUDA [128] and AdaMI [282] utilize the prior knowledge about label distribution in the target domain.

Remarks. There are several remaining SFDA methods that have not been covered in the previous discussions. PCT [342] considers the weights in the classifier layer as source prototypes and develops an optimal transport-based feature alignment strategy between target features and source prototypes, while FTA-FDA [149] readily reduces the distance between the estimated target prototypes and source prototypes. Besides, target prototypes could also be considered representative labeled data, and such a prototypical augmentation helps correct the classifier with pseudo-labeling [152], [216]. SFISA [289] employs implicit feature-level augmentation, and LA-VAE [343], [344] exploits variational auto-encoder to achieve latent feature alignment. In addition, the meta-learning mechanism is adopted in a few studies [345], [346], [347] for the SFDA problem. A recent work [348] even generates common sense rules and adapts models to the target domain to reduce rule violations.

3.3 Learning Scenarios of SFDA algorithms

Closed-set v.s. Open-set. Most existing SFDA methods focus on a closed-set scenario, *i.e.*, $\mathcal{C}_s = \mathcal{C}_t$, and some SFDA algorithms [7], [199] are also validated in a relaxed partial-set setting [349], *i.e.*, $\mathcal{C}_t \subset \mathcal{C}_s$. Besides, several SFDA works [7], [241], [350] consider the open-set learning scenario where the target label space \mathcal{C}_t subsumes the source label space \mathcal{C}_s . To allow more flexibility, open-partial-set domain adaptation [351] ($\mathcal{C}_s \setminus \mathcal{C}_t \neq \emptyset, \mathcal{C}_t \setminus \mathcal{C}_s \neq \emptyset$) is studied in SFDA methods [11], [220], [309], [333], [352]. Several recent studies [136], [332], [353], [354] even develop a unified framework for both open-set and open-partial-set scenarios.

Single-source v.s. Multi-source. To fully transfer knowledge from multiple source models, prior SFDA methods [7], [98], [339] extend the single-source SFDA algorithms by combining these adapted model together in the target domain. Besides, a couple of works [188], [355] are elaborately designed for adaptation with multiple source models. While each source domain typically shares the same label space with the target domain, UnMSMA-MiFL [211] considers a union-set multi-source scenario where the union set of the source label spaces is the same as the target label space.

Single-target v.s. Multi-target. Several SFDA methods [195], [200], [218] also validate the effectiveness of their proposed methods for multi-target domain adaptations where multiple unlabeled target domains exist at the same time. Different from [195], [218], the domain label of each target data is even unknown [200]. Note that, each target domain may come in a streaming manner, thus the model is successively adapted to different target domains [167], [208], [325]. **Unsupervised v.s. Semi-supervised.** Some SFDA methods [237], [356], [357] adapt the source model to the target domain with only a few labeled target samples and adequate unlabeled target samples. In these semi-supervised learning scenarios, the standard classification loss over the labeled data could be readily incorporated to further enhance the adaptation performance [98], [237].

White-box v.s. Black-box. In fact, sharing a model with all the parameters is not flexible for adjustment if the model turns out to have harmful applications³. Instead, the source model is accessible as a black-box module through the cloud application programming interface (API). At an early time, IterLNL [201] treats this black-box SFDA problem as learning with noisy labels, and DINE [207] develops several structural regularizations within the knowledge distillation framework, which is further extended by several methods [128], [132]. Beyond the deep learning framework, several shallow studies [118], [119] focus on the black-box SFDA problem with the target features and their predictions.

Active SFDA. To improve the limited performance gains, MHPL [358] introduces a new setting, active SFDA, where a few target data can be selected to be labeled by human annotators. This active SFDA setting is also studied by other methods [196], [244], [359], [360], and the key point lies in how to select valuable target samples for labeling.

Imbalanced SFDA. ISFDA [137] and two following methods [130], [361] pay attention to the class-imbalanced SFDA problem setting where the source and target label distributions are different and extremely imbalanced. The main

challenge is that the source model would be severely biased toward majority classes and overlook minority classes.

Miscellaneous SFDA tasks. In addition, researchers also focus on other aspects of SFDA, *e.g.*, the robustness against adversarial attacks [362], the forgetting of source knowledge [129], [175], and the vulnerability to membership inference attack [294] and image-agnostic attacks (*e.g.*, blended backdoor attack) [363]. In addition, one recent work [364] explores source-free unsupervised clustering under domain shift, and NOTELA [202] even introduces the SFDA scheme for multi-label classification tasks.

4 TEST-TIME BATCH ADAPTATION

During the testing phase, it is possible that there may exist a single instance or instances from different distributions. This situation necessitates the development of techniques that can adapt off-the-shelf models to individual instances. To be concise, we refer to this learning scheme as *test-time instance adaptation* (*a.k.a.*, standard test-time training [8] and one-sample generalization [365]), which can be viewed as a special case of source-free domain adaptation ($n_t = 1$).

4.1 Problem Definition

Definition 3 (Test-Time Instance Adaptation, TTIA). Given a classifier f_S learned on the source domain \mathcal{D}_S , and an unlabeled target instance $x_t \in \mathcal{D}_T$ under distribution shift, *test-time instance adaptation* aims to leverage the labeled knowledge implied in f_S to infer the label of x_t adaptively.

To the best of our knowledge, the concept *test-time adaptation* is first introduced by Wegmann *et al.* [366] in 1998, where the speaker-independent acoustic model is adapted to a new speaker with unlabeled data at test time. However, this differs from the definition of *test-time instance adaptation* mentioned earlier, as it involves using a few instances instead of a single instance for personalized adaptation. This scenario is frequently encountered in real-world applications, such as in single-image models that are tested on real-time video data [367], [368], [369]. To avoid ambiguity, we further introduce a generalized learning scheme, *test-time batch adaptation*, and give its definition as follows.

Definition 4 (Test-Time Batch Adaptation, TTBA). Given a classifier f_S learned on the source domain \mathcal{D}_S , and a mini-batch of unlabeled target instances $\{x_t^1, x_t^2, \dots, x_t^B\}$ from \mathcal{D}_T under distribution shift ($B \geq 1$), *test-time batch adaptation* aims to leverage the labeled knowledge implied in f_S to infer the label of each instance at the same time.

It is important to acknowledge that the inference of each instance is not independent, but rather influenced by the other instances in the mini-batch. Test-Time Batch Adaptation (TTBA) can be considered a form of SFDA [7] when the batch size B is sufficiently large. Conversely, when the batch size B is equal to 1, TTBA degrades to TTIA [8]. Typically, these schemes assume no access to the source data or the ground-truth labels of data on the target distribution. In the following, we provide a taxonomy of TTBA (including TTIA) algorithms, as well as the learning scenarios.

3. <https://openai.com/blog/openai-api/>

TABLE 2
A taxonomy on TTBA methods with representative strategies.

Families	Representative Strategies
batch normalization calibration	PredBN [13], [370], InstCal [371]
model optimization	TTT [8], GeOS [365], MEMO [14]
meta-learning	MLSR [372], Full-OSHOT [373]
input adaptation	TPT [374], TTA-DAE [375]
dynamic inference	EMEA [376], DRM [377]

4.2 Taxonomy on TTBA Algorithms

4.2.1 Batch Normalization Calibration

Normalization layers (*e.g.*, batch normalization [378] and layer normalization [379]) are considered essential components of modern neural networks. For example, a batch normalization (BN) layer calculates the mean and variance for each activation over the training data \mathcal{X}_S , and normalizes each incoming sample x_s as follows,

$$\hat{x}_s = \gamma \cdot \frac{x_s - \mathbb{E}[X_S]}{\sqrt{\mathbb{V}[X_S] + \epsilon}} + \beta, \quad (19)$$

where γ and β denote the scale and shift parameters (*a.k.a.* the learnable affine transformation parameters), and ϵ is a small constant introduced for numerical stability. The BN statistics (*i.e.*, the mean $\mathbb{E}[X_S]$ and variance $\mathbb{V}[X_S]$) are typically approximated using exponential moving averages over batch-level estimates $\{\mu_k, \sigma_k^2\}$,

$$\hat{\mu}_{k+1} = (1 - \rho) \cdot \hat{\mu}_k + \rho \cdot \mu_k, \quad \hat{\sigma}_{k+1}^2 = (1 - \rho) \cdot \hat{\sigma}_k^2 + \rho \cdot \sigma_k^2, \quad (20)$$

where ρ is the momentum term, k denotes the training step, and the BN statistics over the k -th mini-batch $\{x_i\}_{i=1}^{B_s}$ are

$$\mu_k = \frac{1}{B_s} \sum_i x_i, \quad \sigma_k^2 = \frac{1}{B_s} \sum_i (x_i - \mu_k)^2, \quad (21)$$

where B_s denotes the batch size at training time. During inference, the BN statistics estimated at training time are frozen for each test sample.

AdaBN [335], a classic work in the domain adaptation literature, posits that the statistics in the BN layers represent domain-specific knowledge. To bridge the domain gap, the authors suggest replacing the training BN statistics with new statistics estimated over the entire target domain. This method has been shown to be effective in addressing changes in low-level features such as style or texture [380]. Similarly, DARE [381] utilizes a domain-specific whitening strategy to ensure zero mean and identity variance. Inspired by these techniques, a pioneering TTBA work, PredBN [370], replaces the training BN statistics with those estimated per test batch. L2GP [382] further introduces a specific training-time method that facilitates the use of PredBN at test time.

PredBN+ [13] adopts the running averaging strategy for BN statistics during training and suggests mixing the BN statistics per batch with the training statistics $\{\mu_s, \sigma_s^2\}$ as,

$$\bar{\mu}_t = (1 - \rho_t) \cdot \mu_s + \rho_t \cdot \mu_t, \quad \bar{\sigma}_t^2 = (1 - \rho_t) \cdot \sigma_s^2 + \rho_t \cdot \sigma_t^2, \quad (22)$$

where the test statistics $\{\mu_t, \sigma_t^2\}$ are estimated via Eq. (21), and the hyperparameter ρ_t controls the trade-off between training and estimated test statistics. This method has been extensively examined in [383], *e.g.*, the impacts of mean and variance, and the location of rectified BN statistics. When

combined with adversarially trained models, PredBN+ has been shown to achieve certified accuracy and corruption robustness while maintaining its state-of-the-art empirical robustness against adversarial attacks [384]. Moreover, TTN [385] presents an alternative solution that calibrates the estimation of the variance as follows,

$$\bar{\sigma}_t^2 = (1 - \rho_t) \cdot \sigma_s^2 + \rho_t \cdot \sigma_t^2 + \rho_t(1 - \rho_t)(\mu_t - \mu_s)^2. \quad (23)$$

Instead of using the same value for different BN layers, TTN optimizes the interpolating weight ρ_t during the post-training phase using labeled source data.

Typically, methods that rectify BN statistics may suffer from limitations when the batch size B is small, particularly when $B = 1$. CBNA [386] and SaN [387] directly attempt to mix instance normalization (IN) [388] statistics estimated per instance with the training BN statistics. Instead of manually specifying a fixed value at test time, InstCal [371] introduces an additional module during training to learn the interpolating weight between IN and BN statistics, allowing the network to dynamically adjust the importance of training statistics for each test instance. By contrast, AugBN [389] expands a single instance to a batch of instances using random augmentation, then estimates the BN statistics using the weighted average over these augmented instances.

4.2.2 Model Optimization

Another family of test-time adaptation methods involves adjusting the parameters of a pre-trained model for each unlabeled test instance (batch). These methods are generally divided into two main categories: (1) training with auxiliary tasks [8], [365], [390], which introduces an additional self-supervised learning task in the primary task during both training and test phases, and (2) fine-tuning with unsupervised objectives [14], [391], [392], which elaborately designs a task-specific objective for updating the pre-trained model. **Training with auxiliary tasks.** Motivated by prior works [51], [393] in which incorporating self-supervision with supervised learning in a unified multi-task framework enhances adaptation and generalization, TTT [8] and OSHOT [390] are two pioneering works that leverage the same self-supervised learning (SSL) task at both training and test phases, to implicitly align features from the training domain and the test instance. Specifically, they adopt a common multi-task architecture, comprising the primary classification head $h_c(\cdot; \theta_c)$, the SSL head $h_s(\cdot; \theta_s)$, and the shared feature encoder $f_e(\cdot; \theta_e)$. The following joint objective of TTT or OSHOT is optimized at the training stage,

$$\theta_e^*, \theta_c^*, \theta_s^* = \arg \min_{\theta_e, \theta_c, \theta_s} \sum_{i=1}^{n_s} \mathcal{L}_{pri}(x_i, y_i; \theta_c, \theta_e) + \mathcal{L}_{ssl}(x_i; \theta_s, \theta_e), \quad (24)$$

where \mathcal{L}_{pri} denotes the primary objective (*e.g.*, cross-entropy for classification tasks), and \mathcal{L}_{ssl} denotes the auxiliary SSL objective (*e.g.*, rotation prediction [87] and solving jigsaw puzzles [51]). For each test instance x_t , TTT [8] first adjusts the feature encoder $f_e(\cdot; \theta_e)$ by optimizing the SSL objective,

$$\theta_e(x_t) = \arg \min_{\theta_e} \mathcal{L}_{ssl}(x_t; \theta_s^*, \theta_e), \quad (25)$$

then obtains the prediction with the adjusted model as $\hat{y} = h_c(f_e(x; \theta_e(x_t)); \theta_s^*)$. By contrast, OSHOT [390] mod-

ifies the parameters of both the feature encoder and the SSL head according to the SSL objective at test time. Generally, many follow-up methods adopt the same auxiliary training strategy by developing various self-supervisions for different applications [394], [395], [396], [397], [398], [399], [400]. Among them, TTT-MAE [399] is a recent extension of TTT that utilizes the transformer backbone and replaces the self-supervision with masked autoencoders [401]. TTFlow [400] employs the unsupervised normalizing flows as an alternative to self-supervision for the auxiliary task.

To increase the dependency between the primary task and the auxiliary task, GeOS [365] further adds the features of the SSL head to the primary classification head. SSDN-TTT [402] develops an input-dependent mechanism that allows a self-supervised network to predict the weights of the primary network through filter-level linear combination. Similarly, SR-TTT [403] does not follow the Y-shaped architecture but instead utilizes an explicit connection between the primary task and the auxiliary task. Specifically, SR-TTT takes the output of the primary task as the input of the auxiliary task. Besides, TTCP [404] adopts the pipeline of TTT, but it leverages a test-time prediction ensemble strategy by identifying augmented samples that the SSL head could correctly classify. TTAPS [405] modifies another popular SSL technique [91] as self-supervision and merely updates the shared feature encoder for an augmented batch of a single test image.

Training-agnostic fine-tuning. To avoid altering training with auxiliary tasks in the source domain, the other methods focus on developing unsupervised objectives only for test-time model optimization. DIEM [391] proposes a selective entropy minimization objective for pixel-level semantic segmentation, while MALL [392] enforces edge consistency prior with a weighted normalized cut loss. Besides, MEMO [14] optimizes the entropy of the averaged prediction over multiple random augmentations of the input sample. TTAS [406] further develops a class-weighted entropy objective, while SUTA [407] additionally incorporates minimum class confusion to reduce the uncertainty. In contrast, TTA-FoE [408] partitions the input into multiple patches and matches the feature distributions through a Field of Experts [409], and ShiftMatch [410] focuses on matching spatial correlations per channel for Bayesian neural networks.

Self-supervised consistency regularization under various input variations is also favorable in customizing the pre-trained model for each test input [411], [412]. In particular, SSL-MOCAP [413] leverages the cycle consistency between 3D and 2D data, while REFINE [414] encourages both silhouette consistency and confidence-based mesh symmetry for the test input. SCIO [415] develops a self-constrained optimization method to learn the coherent spatial structure. While adapting image models to a video input [367], [416], [417], [418], [419], [420], ensuring temporal consistency between adjacent frames is a crucial aspect of the unsupervised learning objective. Many other methods directly update the model with the unlabeled objectives tailored to specific tasks, such as image matching [421], [422], image denoising [423], [424], generative modeling [425], [426], and style transfer [427], [428]. As an example, R&R+ [429] fine-tunes the reconstruction model and an unknown original image using the cost function over a single instance.

In addition, the model could be adapted to each instance by utilizing the generated data [430], [431] at test time. As an illustration, TTL-EQA [430] generates numerous synthetic question-answer pairs and subsequently leverages them to infer answers in the given context. ZSSR [432] trains a super-resolution network using solely down-sampled examples extracted from the test image itself. Instead of leveraging task-related self-supervisions, TTT-UCDR [433] even utilizes several off-the-shelf self-supervisions (*e.g.*, rotation prediction [87]) to adjust the feature representations at test time, showing improved cross-domain retrieval performance.

4.2.3 Meta-Learning

MAML [75], a notable example of meta-learning [74], learns a meta-model that can be quickly adapted to perform well on a new task using a small number of samples and gradient steps. Such a learning paradigm is typically well-suited for test-time adaptation, where we can update the meta-model using an unlabeled objective over a few test data. There exist two distinct categories: backward propagation [372], [373], and forward propagation [434], [435]. The latter category does not alter the trained model but includes the instance-specific information in the dynamical neural network.

Backward propagation. Inspired by the pioneering work [432], MLSR [372] develops a meta-learning method based on MAML for single-image super-resolution. Concretely, the meta-objective *w.r.t.* the network parameter θ is shown as,

$$\theta^* = \arg \min_{\theta} \sum_i \mathcal{L}(\text{LR}_i, \text{HR}_i; \theta - \alpha \nabla_{\theta} \mathcal{L}(\text{LR}_i \downarrow, \text{LR}_i; \theta)), \quad (26)$$

where $\mathcal{L}(A, B; \theta) = \|f_{\theta}(A) - B\|_2^2$ is the loss function, α is the learning rate of gradient descent, and $\text{LR}_i \downarrow$ denotes the down-scaled version of the low-resolution input in the paired trained data $(\text{LR}_i, \text{HR}_i)$. At inference time, MLSR first adapts the meta-learned network to the low-resolution test image and its down-sized image ($\text{LR} \downarrow$) using the parameter θ^* learned in Eq. (26) as initialization,

$$\theta_t \leftarrow \theta^* - \alpha \nabla_{\theta} \mathcal{L}(\text{LR} \downarrow, \text{LR}; \theta^*), \quad (27)$$

then generates the high-resolution (HR) image as $f_{\theta_t}(\text{LR})$. Such a meta-learning mechanism based on self-supervised internal learning has also been utilized by follow-up methods [436], [437], [438], [439]. Among them, MetaVFI [440] further introduces self-supervised cycle consistency for video frame interpolation, and OST [441] requires an extra subset of training data to form the meta-test objective in Eq. (27) for deepfake detection.

As an alternative, Full-OSHOT [373] proposes a meta-auxiliary learning approach that optimizes the shared encoder with an inner auxiliary task, providing a better initialization for the subsequent primary task as follows:

$$\min_{\theta_e, \theta_c} \sum_i \mathcal{L}_{pri}(x_i, y_i; \theta_e - \alpha \nabla_{\theta_e} \mathcal{L}_{ssl}(x_i; \theta_e, \theta_s), \theta_c), \quad (28)$$

and the definitions of variables are the same as OSHOT [390] in Eq. (24). After meta-training, the parameters (θ_e, θ_s) are updated for each test sample according to the auxiliary self-supervised objective. This learning paradigm is also known as meta-tailoring [442], where \mathcal{L}_{ssl} in the inner loop affects the optimization of \mathcal{L}_{pri} in the outer loop. Follow-up methods exploit various self-supervisions in the inner

loop, including contrastive learning [442], [443], rotation recognition [444], and reconstruction [445], [446].

Forward propagation. Apart from the shared encoder $f_e(\theta_e)$ above, several other meta-learning methods exploit the normalization statistics [447], [448] or domain prototypes [434], [435] or source dictionary [449] from the inner loop, allowing backward-free adaptation at inference time. Besides, some works incorporate extra meta-adjusters [450], [451] or learnable prompts [452], [453], by taking the instance embedding as input, to dynamically generate a small subset of parameters in the network, which are optimized at the training phase. DSON [454] proposes to fuse IN with BN statistics by linearly interpolating the means and variances, incorporating the instance-specific information in the trained model. Following another popular meta-learning framework with episodic training [50], SSGen [455] suggests episodically dividing the training data into meta-train and meta-test to learn the meta-model, which is generalized to the entire training data for final test-time inference. The same idea is employed by [456], [457], [458] where multiple source domains are involved during training.

4.2.4 Input Adaptation

In contrast to model-level optimization, which updates pre-trained models for input data, another line of test-time adaptation methods focuses on changing input data or features for pre-trained models [375], [459], [460], [461], [462]. This approach bears resemblance to prompt tuning [463], which involves modifying the input prompt rather than the model itself during fine-tuning. For example, TPT [374] employs entropy minimization as [464] to optimize instance-specific prompts for test-time adaptation. CVP [465] optimizes the convolutional visual prompts in the input under the guidance of self-supervised contrastive learning objective.

TTA-AE [459], [466] additionally learns a set of auto-encoders in each layer of the trained model at training time. It is posited that unseen inputs have larger reconstruction errors than seen inputs, thus a set of domain adaptors is introduced at test time to minimize the reconstruction loss. Similarly, TTA-DAE [375] only learns an image-to-image translator (*a.k.a.*, input adaptor) for each input so that the frozen training-time denoising auto-encoder could well reconstruct the network output. TTA-AE [467] follows the Y-shaped architecture of TTT and optimizes both the shared encoder and the additional input adaptor to minimize reconstruction errors in both heads. Instead of auxiliary auto-encoders, AdvTTT [468] leverages a discriminator that is adversarially trained to distinguish real from predicted network outputs, so that the prediction output for each adapted test input satisfies the adversarial output prior. PINER [469] even aims for constructing a good input for the black-box model by analyzing the output change rate.

Besides, OST [470] proposes mapping the target input onto the source data manifold using Fourier style transfer [238], serving as a pre-processor to the primary network. SBTS [471] shifts the style statistics of the test sample to the nearest source domain using AdaIN [472]. By contrast, TAF-Cal [460] further utilizes the average amplitude feature over the training data to perform Fourier style calibration [238] at both training and test phases, bridging the gap between training and test data. It is noteworthy that imposing a

data manifold constraint [404], [462], [473], [474] can aid in achieving better alignment between the test data and unseen training data. Specifically, ITTP [473] trains a generative model over source features with target features projected onto points in the source feature manifold for final inference. DDA [462] exploits the generative diffusion model for target data, while ESA [474] updates the target feature by energy minimization through Langevin dynamics.

In addition to achieving improved recognition results against domain shifts, a certain number of test-time adaptation methods also explore input adaptation for the purpose of test-time adversarial defense [475], [476], [477], [478], [479], [480]. Among them, Anti-Adv [479] perturbs the test input to maximize the classifier’s prediction confidence, and Hedge [478] alters the test input by maximizing the KL-divergence between the classifier’s predictions and the uniform distribution. Besides, SOAP [475] leverages self-supervisions like rotation prediction at both training and test phases and purifies adversarial test examples based on self-supervision only. SSRA [477] only exploits the self-supervised consistency under different augmentations at test time to remove adversarial noises in the attacked data.

4.2.5 Dynamic Inference

Upon multiple pre-trained models learned from the source data, a few works [376], [377], [481] learn the weights for each model, without making any changes to the models themselves. For example, EMEA [376] employs entropy minimization to update the ensemble coefficients before each model. GDU [481] directly obtains the weight by calculating the similarity between the test embedding and the domain embeddings associated with each model. Besides, GPR [482] is one of the early works for test-time adaptation that only adjusts the network predictions instead of the pre-trained model. In particular, it bootstraps the more difficult faces in an image from the more easily detected faces and adopts Gaussian process regression to encourage smooth predictions for similar patches.

4.3 Learning Scenarios of TTBA Algorithms

Instance v.s. Batch. As defined above, test-time adaptation could be divided into two cases: instance adaptation [8], [14] and batch adaptation [13], [367], according to whether a single instance or a batch of instances exist at test time.

Single v.s. Multiple. In contrast to vanilla test-time adaptation that utilizes the pre-trained model from one single source domain, some works (*e.g.*, [365], [376], [377], [455], [460], [471], [473], [474]) are interested in domain generalization problems where multiple source domains exist.

White-box v.s. Black-box. A majority of test-time adaptation methods focus on adapting white-box pre-trained models to test instances, while some other works (*e.g.*, [377], [417], [482]) do not have access to the parameters of the pre-trained model (black-box) and instead adjust the predictions according to generic structural constraints.

Customized v.s. On-the-fly. Most existing TTA methods require training one or more customized models in the source domain, *e.g.*, TTT [8] employs a Y-shaped architecture with an auxiliary head. However, it may be not allowed to train the source model in a customized manner for some

TABLE 3
A taxonomy on OTTA methods with representative strategies.

Families	Representative Strategies
batch normalization calibration	DUA [485], DELTA [486]
entropy minimization	Tent [9], SAR [487]
pseudo-labeling	T3A [10], TAST [488]
consistency regularization	CFA [489], LAME [490]
anti-forgetting regularization	CoTTA [483], EATA [484]

real-world applications. Other works [14], [479] do not rely on customized training in the source domain but develop flexible techniques for adaptation with on-the-fly models.

5 ONLINE TEST-TIME ADAPTATION

Previously, we have considered various test-time adaptation scenarios where pre-trained source models are adapted to a domain [7], [12], a mini-batch [13], [447], or even a single instance [8], [14] at test time. However, offline test-time adaptation typically requires a certain number of samples to form a mini-batch or a domain, which may be infeasible for streaming data scenarios where data arrives continuously and in a sequential manner. To reuse past knowledge like online learning, TTT [8] employs an online variant that does not optimize the model episodically for each input but instead retains the optimized model for the last input.

5.1 Problem Definition

Definition 5 (Online Test-Time Adaptation, OTTA). Given a well-trained classifier f_S on the source domain \mathcal{D}_S and a sequence of unlabeled mini-batches $\{\mathcal{B}_1, \mathcal{B}_2, \dots\}$, *online test-time adaptation* aims to leverage the labeled knowledge implied in f_S to infer labels of samples in \mathcal{B}_i under distribution shift, in an online manner. In other words, the knowledge learned in previously seen mini-batches could be accumulated for adaptation to the current mini-batch.

The above definition corresponds to the problem addressed in Tent [9], where multiple mini-batches are sampled from a new data distribution that is distinct from the source data distribution. Besides, it also encompasses the online test-time instance adaptation problem, as introduced in TTT-Online [8] when the batch size equals 1. However, samples at test time may come from a variety of different distributions, leading to new challenges such as error accumulation and catastrophic forgetting. To address this issue, CoTTA [483] and EATA [484] investigate the continual test-time adaptation problem that adapts the pre-trained source model to the continually changing test data. Such a non-stationary adaptation problem could be also viewed as a special case of the definition above, when each mini-batch may come from a different distribution.

5.2 Taxonomy on OTTA Algorithms

5.2.1 Batch Normalization Calibration

As noted in the previous section, normalization layers such as batch normalization (BN) [378] are commonly employed in modern neural networks. Typically, BN layers

can encode domain-specific knowledge into normalization statistics [335]. A recent work [487] further investigates the effects of different normalization layers under the test-time adaptation setting. In the following, we mainly focus on the BN layer due to its widespread usage in existing methods.

Tent [9] and RNCR [491] propose replacing the fixed BN statistics (*i.e.*, mean and variance $\{\mu_s, \sigma_s^2\}$) in the pre-trained model with the estimated ones $\{\hat{\mu}_t, \hat{\sigma}_t^2\}$ from the t -th test batch. CD-TTA [492] develops a switchable mechanism that selects the most similar one from multiple BN branches in the pre-trained model using the Bhattacharya distance. Besides, Core [493] calibrates the BN statistics by interpolating between the fixed source statistics and the estimated ones at test time, namely, $\mu_t = \rho \hat{\mu}_t + (1 - \rho) \mu_s$, $\sigma_t = \rho \hat{\sigma}_t + (1 - \rho) \sigma_s$, where $\rho \in [0, 1]$ is a momentum hyper-parameter.

Similar to the running average estimation of BN statistics during training, ONDA [494] and two follow-up methods [495], [496] propose initializing the BN statistics $\{\mu_0, \sigma_0\}$ as $\{\mu_s, \sigma_s\}$ and updating them for the t -th test batch,

$$\begin{aligned} \mu_t &= \rho \hat{\mu}_t + (1 - \rho) \mu_{t-1}, \\ \sigma_t^2 &= \rho \hat{\sigma}_t^2 + (1 - \rho) \frac{n_t}{n_t - 1} \sigma_{t-1}^2, \end{aligned} \quad (29)$$

where n_t denotes the number of samples in the batch, and ρ is a momentum hyperparameter. Instead of a constant value for ρ , MECTA [497] considers a heuristic weight through computing the distance between $\{\mu_{t-1}, \sigma_{t-1}\}$ and $\{\hat{\mu}_t, \hat{\sigma}_t\}$.

To decouple the gradient backpropagation and the selection of BN statistics, GpreBN [498] and DELTA [486] adopt the following reformulation of batch re-normalization [499],

$$\hat{x}_t = \gamma \cdot \frac{\frac{x_t - \hat{\mu}_t}{\hat{\sigma}_t} \cdot sg(\hat{\sigma}_t) + sg(\hat{\mu}_t) - \mu}{\sigma} + \beta, \quad (30)$$

where $sg(\cdot)$ denotes the stop-gradient operation, and $\{\gamma, \beta\}$ are the affine parameters in the BN layer. To obtain stable BN statistics $\{\mu, \sigma\}$, these methods utilize the test-time dataset-level running statistics via the moving average like Eq. (29).

For online adaptation with a single sample, MixNorm [500] mixes the estimated IN statistics with the exponential moving average BN statistics at test time. On the other hand, DUA [485] adopts a decay strategy for the weighting hyper-parameter ρ and forms a small batch from a single image to stabilize the online adaptation process. To obtain more accurate estimates of test-time statistics, NOTE [501] maintains a class-balanced memory bank that is utilized to update the BN statistics using an exponential moving average. Additionally, NOTE proposes a selective mixing strategy that only calibrates the BN statistics for detected out-of-distribution samples. TN-SIB [502] also leverages a memory bank that provides samples with similar styles to the test sample, to estimate more accurate BN statistics.

5.2.2 Entropy Minimization

Entropy minimization is a widely-used technique to handle unlabeled data. A pioneering approach, Tent [9], proposes minimizing the mean entropy over the test batch to update the affine parameters $\{\gamma, \beta\}$ of BN layers in the pre-trained model, followed by various subsequent methods [498], [501], [503]. Notably, VMP [297] reformulates Tent in a probabilistic framework by introducing perturbations into the

model parameters by variational Bayesian inference. Several other methods [495], [504], [505] also focus on minimizing the entropy at test time but utilize different combinations of learnable parameters. BACS [506] incorporates the entropy regularization for unlabeled data in the approximate Bayesian inference algorithm, and samples multiple model parameters to obtain the marginal probability for each sample. In addition, TTA-PR [507] proposes minimizing the average entropy of predictions under different augmentations. FEDTHE+ [508] employs the same adaptation scheme as MEMO [14] that minimizes the entropy of the average prediction over different augmentations, while FTEA [496] fuses the predictions from different modalities. MEMO-CL [509] develops a filtering approach that selects partial augmentations for marginal entropy minimization.

To avoid overfitting to non-reliable and redundant test data, EATA [484] develops a sample-efficient entropy minimization strategy that identifies samples with lower entropy values than the pre-defined threshold for model updates, which is also adopted by follow-up methods [487], [510]. CD-TTA [492] leverages the similarity between feature statistics of the test sample and source running statistics as sample weights, instead of using discrete weights $\{0, 1\}$. Besides, DELTA [486] derives a class-wise re-weighting approach that associates sample weights with corresponding pseudo labels to mitigate bias towards dominant classes.

There exist many alternatives to entropy minimization for adapting models to unlabeled test samples including class confusion minimization [493], batch nuclear-norm maximization [491], maximum squares loss [492], and mutual information maximization [511], [512]. In addition, MuSLA [511] further considers the virtual adversarial training objective that enforces classifier consistency by adding a small perturbation to each sample. SAR [487] encourages the model to lie in a flat area of the entropy loss surface and optimizes the minimax entropy objective below,

$$\min_{\theta} \max_{\|\Delta_{\theta}\|_2 \leq \epsilon} \mathcal{H}(x; \theta + \Delta_{\theta}), \quad (31)$$

where $\mathcal{H}(\cdot)$ denotes the entropy function, and Δ_{θ} denotes the weight perturbation in a Euclidean ball with radius ϵ . Moreover, a few methods [513], [514] even employ entropy maximization for specific tasks, for example, AUTO [514] performs model updating for unknown samples at test time.

5.2.3 Pseudo-labeling

Unlike the unidirectional process of entropy minimization, many OTTA methods [490], [492], [511], [515], [516], [517] adopt pseudo labels generated at test time for model updates. Among them, MM-TTA [518] proposes a selective fusion strategy to ensemble predictions from multiple modalities. TTA-MDE [519] selects pseudo labels with a high consistency score across different models for further labeling. Besides, DLTAA [520] obtains soft pseudo labels by averaging the predictions of its nearest neighbors in a memory bank, and subsequently optimizes the symmetric KL divergence between the model outputs and these pseudo labels. TAST [488] proposes a similar approach that reduces the difference between predictions from a prototype-based classifier and a neighbor-based classifier. Notably, SLR+IT [521] develops a negative log-likelihood ratio loss instead

of the commonly used cross-entropy loss, providing non-vanishing gradients for highly confident predictions.

Conjugate-PL [522] presents a way of designing unsupervised objectives for TTA by leveraging the convex conjugate function. The resulting objective resembles self-training with specific soft labels, referred to as conjugate pseudo labels. A recent work [523] theoretically analyzes the difference between hard and conjugate labels under gradient descent for a binary classification problem. Motivated by the idea of negative learning [192], ECL [524] further considers complementary labels from the least probable categories. Besides, T3A [10] proposes merely adjusting the classifier layer by computing class prototypes using online unlabeled data and classifying each unlabeled sample based on its distance to these prototypes. MSLC-TSD [525] computes class prototypes using a dynamic memory bank, providing accurate pseudo labels for subsequent model updates.

5.2.4 Consistency Regularization

In the classic mean teacher [106] framework, the pseudo labels under weak data augmentation obtained by the teacher network are known to be more stable. Built on this framework, RMT [526] pursues the teacher-student consistency in predictions through a symmetric cross-entropy measure, while OIL [527] only exploits highly confident samples during consistency maximization. VDP [528] and its follow-up method [529] also utilize this framework to update visual domain prompts with the pre-trained model being frozen. Moreover, CoTTA [483] further employs multiple augmentations to refine the pseudo labels from the teacher network, which is also applied in other methods [530], [531]. Inspired by maximum classifier discrepancy [329], AdaODM [464] proposes minimizing the prediction disagreement between two classifiers at test time to update the feature encoder.

Apart from the model variation above, several methods [507], [532], [533], [534], [535], [536] also enforce the consistency of the corresponding predictions among different augmentations. In particular, SWR-NSP [512] introduces an additional nearest source prototype classifier at test time and minimizes the difference between predictions under two different augmentations. FTEA [496] performs knowledge distillation from the fused multi-modality prediction to the prediction of each modality. Besides, many methods [505], [516], [537], [538], [539] leverage the temporal coherence for video data and design a temporal consistency objective at test time. For example, TeCo [505] encourages adjacent frames to have semantically similar features to increase the robustness against corruption at test time. LAME [490] further exploits the neighbor consistency, forcing neighboring points in the feature space to have consistent assignments.

In contrast to the consistency constraints in the prediction space above, GAPGC [540] and FEDTHE+ [508] pursue consistency in the feature space. Several other OTTA methods [525], [526], [535], [541], [542] even pursue consistency between test features and source or target prototypes in the feature space. CFA [489] further proposes matching multiple central moments to achieve feature alignment. Furthermore, ACT-MAD [543] performs feature alignment by minimizing the discrepancy between the pre-computed training statistics and the estimates of test statistics. ViTTA [532] and TTAC [535] calculate the online estimates of feature mean

and variance at test time instead. Additionally, CAFA [544] uses the Mahalanobis distance to achieve low intra-class variance and high inter-class variance for test data.

5.2.5 Anti-forgetting Regularization

Previous studies [483], [484] find that the model optimized by TTA methods suffers from severe performance degradation (named forgetting) on original training samples. To mitigate the forgetting issue, a natural solution is to keep a small subset of training data that is further learned at test time as regularization [516], [526], [538]. PAD [541] comes up with an alternative approach that keeps the relative relationship of irrelevant auxiliary data unchanged after test-time optimization. AUTO [514] maintains a memory bank to store easily recognized samples for replay and prevents overfitting towards unknown samples at test time.

Another anti-forgetting solution lies in using merely a few parameters for test-time model optimization. For example, Tent [9] only optimizes the affine parameters in the BN layers for test-time adaptation, and AUTO [514] updates the last feature block in the pre-trained model. SWR-NSP [512] divides the entire model parameters into shift-agnostic and shift-biased parameters and updates the former less and the latter more. Recently, VDP [528] fixes the pre-trained model but only optimizes the input prompts during adaptation.

Besides, CoTTA [483] proposes a stochastic restoration technique that randomly restores a small number of parameters to the initial weights in the pre-trained model. PETAL [530] further selects parameters with smaller gradient norms in the entire model for restoration. By contrast, EATA [484] introduces an importance-aware Fisher regularizer to prevent excessive changes in model parameters. The importance is estimated from test samples with generated pseudo labels. SAR [487] proposes a sharpness-aware and reliable optimization scheme, which removes samples with large gradients and encourages model weights to lie in a flat minimum. Further, EcoTTA [510] presents a self-distilled regularization by forcing the output of the test model to be close to that of the pre-trained model.

5.2.6 Miscellaneous Methods

In addition, there are several other proposed solutions for the OTTA problem, such as meta-learning [502], [545], [546], Hebbian learning [504], and adversarial data augmentation [531], [540]. Besides, ETLT [547] develops a test-time calibration approach that learns the linear relation between features and detection scores in an online manner.

Remarks. Besides, a recent work [548] studies two different distribution shifts—contextual shifts and semantic shifts—and compares three TTA methods including Tent [9] and CoTTA [483]. Notably, several common pitfalls are identified from prior efforts based on a diverse set of distribution shifts and two comprehensive evaluation protocols [549]. Among them, one important pitfall is choosing appropriate hyperparameters is exceedingly difficult due to online batch dependency during adaptation.

5.3 Learning Scenarios of OTTA Algorithms

Stationary v.s. Dynamic. As mentioned above, there are two categories of OTTA tasks. Vanilla OTTA [9] assumes the test

data comes from a stationary distribution, while continual OTTA [483] assumes a continually changing distribution.

Other differences among OTTA algorithms are the same as those among TTBA algorithms, *i.e.*, **instance v.s. batch**, **customized v.s. on-the-fly**, and **single v.s. multiple**.

6 TEST-TIME PRIOR ADAPTATION

Different from SFDA methods that narrowly focus on adaptation under data distribution change $p_S(x) \neq p_T(x)$, this section reviews another family of test-time adaptation methods under label distribution change, $p_S(y) \neq p_T(y)$.

6.1 Problem Definition

In 2002, Saerens *et al.* [36] proposes a well-known *prior adaptation* framework that adapts an off-the-shelf classifier to a new label distribution with unlabeled data at test time. We reuse the variables defined in the above section and give the definition of *test-time prior adaptation* as follows.

Definition 6 (Test-Time Prior Adaptation, TTPA). Given a classifier f_S learned on the source domain \mathcal{D}_S and an unlabeled target domain \mathcal{D}_T , *test-time prior adaptation* aims to correct the probabilistic predictions $p_S(y|x) = f_S(x)$ for samples in \mathcal{D}_T under the prior shift assumption.

Formally, prior shift [20] (*a.k.a.* label shift [17], [550]), denotes that the marginal label distributions differ but the conditional distributions of input given label remain the same across domains, namely, $p_S(y) \neq p_T(y)$, $p_S(x|y) = p_T(x|y)$. Based on the Bayes theorem and the label shift assumption, we have the following equations,

$$\begin{aligned} p_T(y|x) &= \frac{p_T(x|y)p_T(y)}{p_T(x)} = p_S(x|y) \cdot \frac{p_T(y)}{p_T(x)} \\ &= \frac{p_S(y|x)p_S(x)}{p_S(y)} \cdot \frac{p_T(y)}{p_T(x)} \propto p_S(y|x) \cdot \frac{p_T(y)}{p_S(y)}, \end{aligned} \quad (32)$$

where $p_T(y|x)$ denotes the corrected posterior probability from the biased prediction $p_S(y|x) \in \mathbb{R}^C$ for each sample x in the target set \mathcal{D}_T . Therefore, the key to the prior-shift adaptation problem lies in how to obtain the test-time prior $p_T(y)$ or the prior ratio $w(y) = p_T(y)/p_S(y)$. Moreover, the prior ratio could be also considered as the importance weight [551] ($p_T(x, y)/p_S(x, y)$), which further facilitates re-training the source classifier with a weighted empirical risk minimization. Note that, we only review different strategies of estimating the prior or prior ratio without the source training data, regardless of the follow-up correction strategy.

6.2 Taxonomy on TTPA Algorithms

6.2.1 Confusion Matrix

Prior estimation. As introduced in [36], [552], a standard procedure used for estimating the test-time prior involves the computation of the confusion matrix, $[C_{\hat{y}|y}]_{i,j} = p(\hat{y} = j|y = i)$, denoting the probability of predicting class j when the observation in fact belongs to the i -th class. Specifically, the confusion matrix-based method tries to solve the following system of C linear equations with respect to $p_T(\hat{y})$,

$$p_T(\hat{y} = i) = \sum_{j=1}^C [C_{\hat{y}|y}]_{i,j} p_T(y = j), \quad i = 1, \dots, C. \quad (33)$$

Intuitively, there exists a closed-form solution, $\hat{p}_T(y) = [C_{\hat{y}|y}]^{-1} p_T(\hat{y})$, where $p_T(\hat{y})$ denotes the estimated class label frequency over the original predictions $f_S(x)$ on the unlabeled test data set. Note that, the confusion matrix $C_{\hat{y}|y}$ could be obtained using a holdout training data set [17], which is proven to be unchanged under the label shift assumption. To avoid such infeasible solutions, a bootstrap method [553] is developed to alternately calculate the confusion matrix $C_{\hat{y}|y}$ on holdout data and estimate the predicted class frequency $p_T(\hat{y})$ on test data.

Prior ratio estimation. BBSE [17] exploits the confusion matrix with joint probability instead of the conditional one above, and directly estimates the prior ratio $w(y) = p_T(y)/p_S(y)$ based on the following equation,

$$p_T(\hat{y} = i) = \sum_{j=1}^C [C_{\hat{y},y}]_{i,j} w(y = j), \quad i = 1, \dots, C, \quad (34)$$

where the confusion matrix $[C_{\hat{y},y}]_{i,j}$ is computed with hard predictions over the holdout training data set, and a soft variant is further explored in [20]. To diminish large variances in prior ratio estimation, RLLS [37] proposes to incorporate an additional regularization term $\|\theta\|_2$, forming the constrained optimization problem as follows,

$$\min_{\theta} \|C_{\hat{y},y} \cdot (\theta + \mathbf{1}) - p_T(\hat{y})\|_2 + \lambda \|\theta\|_2, \quad (35)$$

where $\theta = w - \mathbf{1}$ is called the amount of weight shift, and λ is a regularization hyper-parameter.

6.2.2 Maximum Likelihood Estimation (MLE)

MLLS [15], [36] proposes a straightforward instance of the Expectation-Maximization (EM) algorithm to maximize the likelihood, providing a simple iterative procedure for estimating the prior $p_T(y)$. In particular, MLLS alternately re-estimates the posterior probabilities $\hat{p}_T(y|x)$ for each test sample and the prior probability $\hat{p}_T(y)$ as follows,

$$\begin{aligned} \hat{p}_T(y = i|x) &= \frac{\hat{w}(y = i) \cdot p_S(y = i|x)}{\sum_{j=1}^C \hat{w}(y = j) \cdot p_S(y = j|x)}, \quad (\text{E-step}) \\ \hat{p}_T(y = i) &= \frac{1}{n_t} \sum_{x \in \mathcal{X}_t} \hat{p}_T(y = i|x), \quad (\text{M-step}) \\ \hat{w}(y = i) &= \hat{p}_T(y = i)/p_S(y = i), \quad \forall i \in [1, C], \end{aligned} \quad (36)$$

where $p_S(y|x)$ is modeled by the classifier's output $f_S(x)$, and $\hat{p}_T(y)$ is initialized by the estimation of class frequencies in the target domain. As proven in [36], this iterative procedure will increase the likelihood of target data at each step. Besides, the EM algorithm could be also reformulated as minimizing the KL divergence between the target data distribution $p_T(x)$ and its approximation by a linear combination of class-wise distributions $p'_T(x)$ [554], [555],

$$p'_T(x) = \sum_c \lambda_c p_T(x|y = c), \quad s.t. \sum_c \lambda_c = 1, \quad (37)$$

where λ_c can be considered as an exact approximation of $p_T(y = c)$. Thereby, a variant is proposed by replacing the KL divergence with the Pearson divergence [556], which is computed efficiently and analytically, and shown to be more robust against noise and outliers.

Calibration. Two following works [557], [558] focus on the calibration of estimated probabilities in the EM algorithm. On the one hand, the probability estimates are calibrated

from naive Bayes [557] in the maximization step (M-step). On the other hand, BCTS [558] offers a temperature scaling solution to achieve an explicit calibration for $p_T(y|x)$ in the expectation step (E-step), which is inspired by a prominent method [559]. To be specific, BCTS scales the softmax logits $l(x)$ with a temperature parameter T and additionally considers the class-specific bias terms $\{b_i\}_{i=1}^C$ as follows,

$$p_T(y = i|x) = \frac{\exp([l(x)]_i/T) + b_i}{\sum_j \exp([l(x)]_j/T) + b_j}. \quad (38)$$

Further, the EM algorithm is proven to converge to a global maximum for the likelihood over holdout data [558]. On the contrary, TTLA [560] tries to learn a well-calibrated classifier in the training time using logit adjustment.

Maximum a posteriori estimation. As shown in [561], the Dirichlet hyper-prior on the class prior probabilities is integrated into the MLE framework, forming the Maximum a Posteriori (MAP) estimation framework. Besides, the MAP objective is optimized by the projected gradient descent algorithm under the simplex constraint.

6.2.3 MLE with Confusion Matrix

MLLS-CM [550] unifies the confusion matrix strategy [17] and the EM-based MLE strategy [36] under a common framework. In particular, the confusion matrix strategy is employed to obtain a calibrated predictor in the E-step. Besides, SCM^L [20], [562] provides another confusion-based MLE approach that inserts the confusion matrix-based prior into the log-likelihood maximization objective as follows,

$$\begin{aligned} \max_{p_T(y)} \quad & \sum_{i=1}^C m_i \log(C_{\hat{y}=i|y} \cdot p_T(y)), \\ s.t. \quad & \sum_i p_T(y = i) = 1, \quad p_T(y = i) \geq 0, \quad i \in [1, C], \end{aligned} \quad (39)$$

where m_i is the number of samples predicted to the i -th class on the target set. The projected gradient ascent algorithm with a simplex constraint is iteratively employed to maximize such a convex objective above.

Remarks. In addition, the prior over the target set could be directly used without the prior estimation [561], e.g., the uniform prior is utilized in the winning submission of FGVCx Fungi Classification Kaggle competition [563]. Moreover, the post-training prior rebalancing technique [564] even finds an interpolated distribution between $p_S(y|x)$ and $p_T(y|x)$. Recently, SADE [565] provides a different TTPA strategy from the perspective of model aggregation, requiring multiple source expert classifiers and only learning the expert-wise weights for the target domain by maximizing the consistency under different stochastic data augmentations.

6.2.4 Online TTPA Algorithms

Without requiring the whole dataset at test time, another line of TTPA methods [16], [566] focus on the case of online adaptation at prediction time (i.e., sample after sample). It is worth noting that offline TTPA algorithms could also address the online label shift scenario, for instance, estimating the priors from the already-seen examples [561]. In the following, we mainly introduce online TTPA methods when historical samples could not be stored.

OEM [566] provides an online (or evolving) prior estimation approach to adjust the posterior probability sample

by sample. OEM is built on the seminal work [36] in Eq. (36) that extends the prior estimation step as:

$$\hat{p}_{\mathcal{T}}(y = i) \leftarrow (1 - \alpha)\hat{p}_{\mathcal{T}}(y = i) + \alpha\hat{p}_{\mathcal{T}}(y = i|x_t), \quad (40)$$

where x_t denotes the t -th test data from the sequence, and α is the hyper-parameter to control the updating rate. Besides, PTCA_U [16] directly estimates a categorical distribution from a set of L most recent target samples below,

$$\hat{p}_{\mathcal{T}}(y = i|x_t) = \frac{\sum_{r=t-L+1}^t [\delta_r(y)]_i + \alpha}{L + C \cdot \alpha}, \quad (41)$$

where $\delta_r(y) \in \mathbb{R}^C$ denotes the prediction of x_r , and $\alpha > 0$ is the parameter of the Dirichlet prior. Then the corrected prediction is obtained using the Bayes theorem in Eq. (32). By contrast, OGD [567] obtains an unbiased estimate of the expected 0-1 loss in terms of the confusion matrix $C_{\hat{y}|y}$ and the label marginal probabilities $p_{\mathcal{T}}(y)$, then employs the online gradient descent technique to update the re-weighted classifier in Eq. (32) after each unlabeled sample.

6.3 Learning Scenarios of TTPA Algorithms

Almost all TTPA methods require not only the source classifier but also additional information about the source domain, *e.g.*, the source label distribution $p_S(y)$ [36], and the conditional confusion matrix over holdout data $C_{\hat{y}|y}$ [17].

Offline *v.s.* Online. Regarding the target domain, TTPA methods could be divided into two categories: offline TTPA [17], [36], [558], [565] where the whole target domain is available; and online TTPA [16], [566], [567] where the class distribution varies over time. Additionally, two supervised cases with online feedback are studied in [16], *i.e.*, online feedback (the correct label is revealed to the system after prediction) and bandit feedback (the decision made by the system is correct or not is revealed).

7 APPLICATIONS

7.1 Image Classification

The most common application of test-time adaptation is multi-class image classification. Firstly, SFDA methods are commonly evaluated and compared on widely used DA datasets, including Digits, Office, Office-Home, VisDA-C, and DomainNet, as described in previous studies [7], [98], [142], [183]. Secondly, TTBA and OTTA methods consider natural distribution shifts in object recognition datasets, *e.g.*, corruptions in CIFAR-10-C, CIFAR-100-C, and ImageNet-C, natural renditions in ImageNet-R, misclassified real-world samples in ImageNet-A, and unknown distribution shifts in CIFAR-10.1, as detailed in previous studies [8], [9], [13], [14]. In addition, TTBA and OTTA methods are also evaluated in DG datasets such as VLCS, PACS, and Office-Home, as described in previous studies [10], [365], [473], [528]. Thirdly, TTPA methods always conduct comparisons on CIFAR-10, CIFAR-100, and MNIST by simulating label shift based on a Dirichlet distribution, as detailed in previous studies [17], [20], [558]. Besides, long-tailed datasets, such as ImageNet-LT, CIFAR-100-LT, and Places-LT, are also utilized to evaluate TTPA methods, as depicted in [20], [565].

7.2 Semantic Segmentation

Semantic segmentation aims to categorize each pixel of the image into a set of semantic labels, which is a critical module in autonomous driving. Many domain adaptive semantic segmentation datasets, such as GTA5-to-Cityscapes, SYNTHIA-to-Cityscapes, and Cityscapes-to-Cross-City, are commonly adopted to evaluate SFDA methods, as depicted in [126], [131], [284]. In addition to these datasets, BDD100k, Mapillary, and WildDash2, and IDD are also used to conduct comparisons for TTBA and OTTA methods, as shown in [371], [387]. OTTA methods further utilize Cityscapes-to-ACDC and Cityscapes-to-Foggy&Rainy Cityscapes for evaluation and comparison, as described in [483], [529], [568].

7.3 Object Detection

Object detection is a fundamental computer vision task that involves locating instances of objects in images. While early TTA methods [569], [570], [571] focus on binary tasks such as pedestrian and face detection, lots of current efforts are devoted to generic multi-class object detection. Typically, many domain adaptive object detection tasks including Cityscapes-to-BDD100k, Cityscapes-to-Foggy Cityscapes, KITTI-to-Cityscapes, Sim10k-to-Cityscapes, Pascal-to-Clipart&Watercolor are commonly used by SFDA methods for evaluation and comparison, as detailed in [148], [199], [240], [262]. Additionally, datasets like VOC-to-Social Bikes and VOC-to-AMD are employed to evaluate TTBA methods, as shown in [373], [390].

7.4 Beyond Vanilla Object Images

Medical images. Medical image analysis is another important downstream field of TTA methods, *e.g.*, medical image classification [572], [573], medical image segmentation [375], [466], and medical image detection [4]. Among them, medical segmentation attracts the most attention in this field.

3D point clouds. Nowadays, 3D sensors have become a crucial component of perception systems. Many classic tasks for 2D images have been adapted for LiDAR point clouds, such as 3D object classification [143], [397], 3D semantic segmentation [574], and 3D object detection [209], [575].

Videos. As mentioned above, TTBA and OTTA methods can well address how to efficiently adapt an image model to real-time video data for problems such as segmentation [369], depth prediction [446], and frame interpolation [440], [576]. Besides, a few studies [291], [292], [505] investigate the SFDA scheme for video-based tasks like action recognition.

Multi-modal data. Researchers also develop different TTA methods for various multi-modal data, *e.g.*, RGB and audio [285], RGB and depth [269], [518], RGB and motion [292].

Face and body data. Facial data is also an important application of TTA methods, such as face recognition [577], face anti-spoofing [138], [307], [345], deepfake detection [441], and expression recognition [247]. For body data, TTA methods also pay attention to tasks such as pose estimation [134], [394], [415] and mesh reconstruction [418], [537].

7.5 Beyond Vanilla Recognition Problems

Low-level vision. TTA methods can be also applied to low-level vision problems, *e.g.*, image super-resolution [372],

[436], image denoising [423], [578], image deblurring [437] and image dehazing [438], [579]. Besides, the TTA paradigm is introduced to image registration [421], [422], inverse problems [429], [469], [580], [581], quality assessment and enhancement [254], [295].

Retrieval. Besides classification problems, TTA can also be applied to retrieval scenarios, e.g., person re-identification [457], [582] and sketch-to-image retrieval [433], [445].

Generative modeling. TTA method can also vary the pre-trained generative model for style transfer and data generation [425], [426], [427], [583].

Defense. Another interesting application of TTA is the test-time adversarial defense for image classification [475], [476], [479], [503], which tries to generate robust predictions for possible perturbed test samples.

7.6 Natural Language Processing (NLP)

The TTA paradigm is also studied in tasks of the NLP field, such as reading comprehension [430], question answering [527], [584], sentiment analysis [245], [585], entity recognition [376], and aspect prediction [453], [586]. In particular, a competition⁴ has been launched under data sharing restrictions, comprising two NLP semantic tasks [587]: negation detection and time expression recognition. Besides, early TTPA methods focus on word sense disambiguation [557].

7.7 Beyond CV and NLP

Graph data. For graph data (e.g., social networks), TTA methods are evaluated and compared on two tasks: graph classification [540], [588] and node classification [412], [589].

Speech processing. As far, there have been three TTA methods, i.e., audio classification [202], speaker verification [435] and speech recognition [407].

Miscellaneous signals. In addition, TTA methods are also validated on other types of signals, e.g., radar signals [186], EEG signals [590], and vibration signals [168].

Reinforcement learning. Some TTA methods [396], [591] also address the generalization of reinforcement learning policies across different environments.

7.8 Evaluation

As the name suggests, TTA methods should evaluate the performance of test data after test-time optimization immediately. However, there are different protocols for evaluating TTA methods in the field, making a rigorous evaluation protocol important. Firstly, some SFDA works, particularly for domain adaptive semantic segmentation [126], [131] and classification on DomainNet, adapt the source model to an unlabeled target set and evaluate the performance on the test set that shares the same distribution as the target set. However, this in principle violates the setting of TTA, although the performance on the test set is always consistent with that of the target set. *We suggest that such SFDA methods report the performance on the target set at the same time.* Secondly, some SFDA works such as BAIT [274] offer an online variant in their papers, but such online SFDA methods differ from OTTA in that the evaluation

is conducted after one full epoch. *We suggest online SFDA methods change the name to “one-epoch SFDA” to avoid confusion with OTTA methods.* Thirdly, for continual TTA methods [483], [484], the evaluation of each mini-batch is conducted before optimization on that mini-batch. This manner differs from the standard evaluation protocol of OTTA [8] where optimization is conducted ahead of evaluation. *We suggest that continual TTA methods follow the same protocol as vanilla OTTA methods.*

8 EMERGING TRENDS AND OPEN CHALLENGES

8.1 Emerging Trends

Diverse downstream fields. Although most existing efforts in the TTA field have been devoted to visual tasks such as image classification and semantic segmentation, a growing number of TTA methods are now focusing on other understanding problems over video data [291], multi-modal data [518], and 3D point clouds [574], as well as regression problems like pose estimation [134], [592].

Black-box pre-trained models. In contrast to existing SFDA methods that use white-box models from the source domain, some recent works [207], [316] have focused on adaptation with black-box models, which provide only probability scores or even discrete labels for the target data. Moreover, methods tailored for black-box scenarios can enable knowledge transfer from large models to edge devices.

Open-world adaptation. Existing TTA methods always follow the closed-set assumption; however, a growing number of SFDA methods [136], [352], [353] are beginning to explore model adaptation under an open-set setting. A recent OTTA method [514] further focus on the performance of out-of-distribution detection task at test time. Besides, for large distribution shifts, it is challenging to perform effective knowledge transfer by relying solely on unlabeled target data, thus several recent works [244], [360] also introduce active learning to involve human in the loop.

Memory-efficient continual adaptation. In real-world applications, test samples may come from a continually changing environment [483], [484], leading to catastrophic forgetting. To reduce memory consumption while maintaining accuracy, recent works [497], [510] propose different memory-friendly OTTA solutions for resource-limited end devices.

On-the-fly adaptation. The majority of existing TTA methods require a customized pre-trained model from the source domain, bringing the inconvenience for instant adaptation. Thus, fully test-time adaptation [9], which allows adaptation with an on-the-fly model, has attracted increasing attention.

8.2 Open Problems

Theoretical analysis. While most existing TTA works focus on developing effective methods to obtain better empirical performance, the theoretical analysis remains an open problem. We believe that rigorous analyses can provide in-depth insights and inspire the development of new TTA methods.

Benchmark and validation. As there does not exist a labeled validation set, validation also remains a significant and unsolved issue for TTA methods. Existing methods often determine hyper-parameters through grid search in the test data, which is infeasible in real-world applications. Alternatively,

4. <https://competitions.codalab.org/competitions/26152>

a new benchmark can be built where a labeled validation set and an unlabeled test set exist at test time, providing a more realistic evaluation scenario for TTA methods.

New applications. Tabular data [593] in vectors of heterogeneous features is essential for numerous industrial applications, and time series data [594] is predominant in many real-world applications including healthcare and manufacturing. To our knowledge, few prior work has studied TTA in the context of tabular data or time series data, despite their importance and prevalence in real-world scenarios.

Trustworthiness. Current TTA methods tend to focus more on recognition performance under different distribution shifts and robustness against different attacks, while ignoring other goals of trustworthy machine learning [595], e.g., fairness, security, privacy, and explainability.

Big models. Recently, big models (e.g., ChatGPT and GPT-4) have attracted widespread attention due to their surprisingly strong ability in a variety of machine learning tasks. However, it remains an open problem on how to leverage them for better generalization ability in downstream tasks.

9 CONCLUSION

Learning to adapt a pre-trained model to unlabeled data under distribution shifts is an emerging and critical problem in the field of machine learning. This survey provides a comprehensive review of four related topics: source-free domain adaptation, test-time batch adaptation, online test-time adaptation, and test-time prior adaptation. These topics are unified as a broad learning paradigm of test-time adaptation. For each topic, we first introduce its history and definition, followed by a new taxonomy of advanced algorithms. Additionally, we provide a review of applications related to test-time adaptation, as well as an outlook of emerging research trends and open problems. We believe that this survey will assist both newcomers and experienced researchers in better understanding the current state of research in test-time adaptation under distribution shifts.

ACKNOWLEDGMENTS

The authors would like to thank Lijun Sheng and Dapeng Hu for their valuable discussions and contributions to this work. Discussions, comments, and questions are all welcomed in <https://github.com/tim-learn/awesome-test-time-adaptation>.

REFERENCES

- [1] J. Quinonero-Candela, M. Sugiyama, A. Schwaighofer, and N. D. Lawrence, *Dataset shift in machine learning*. MIT Press, 2008.
- [2] K. Saenko, B. Kulis, M. Fritz, and T. Darrell, "Adapting visual category models to new domains," in *Proc. ECCV*, 2010, pp. 213–226.
- [3] Y.-H. Chen, W.-Y. Chen, Y.-T. Chen, B.-C. Tsai, Y.-C. Frank Wang, and M. Sun, "No more discrimination: Cross city adaptation of road scene segmenters," in *Proc. ICCV*, 2017, pp. 1992–2001.
- [4] X. Liu and Y. Yuan, "A source-free domain adaptive polyp detection framework with style diversification flow," *IEEE Transactions on Medical Imaging*, vol. 41, no. 7, pp. 1897–1908, 2022.
- [5] K. Zhou, Z. Liu, Y. Qiao, T. Xiang, and C. C. Loy, "Domain generalization: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.
- [6] W. M. Kouw and M. Loog, "A review of domain adaptation without target labels," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 3, pp. 766–785, 2019.
- [7] J. Liang, D. Hu, and J. Feng, "Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation," in *Proc. ICML*, 2020, pp. 6028–6039.
- [8] Y. Sun, X. Wang, Z. Liu, J. Miller, A. Efros, and M. Hardt, "Test-time training with self-supervision for generalization under distribution shifts," in *Proc. ICML*, 2020, pp. 9229–9248.
- [9] D. Wang, E. Shelhamer, S. Liu, B. Olshausen, and T. Darrell, "Tent: Fully test-time adaptation by entropy minimization," in *Proc. ICLR*, 2021.
- [10] Y. Iwasawa and Y. Matsuo, "Test-time classifier adjustment module for model-agnostic domain generalization," in *Proc. NeurIPS*, 2021, pp. 2427–2440.
- [11] J. N. Kundu, N. Venkat, and R. V. Babu, "Universal source-free domain adaptation," in *Proc. CVPR*, 2020, pp. 4544–4553.
- [12] R. Li, Q. Jiao, W. Cao, H.-S. Wong, and S. Wu, "Model adaptation: Unsupervised domain adaptation without source data," in *Proc. CVPR*, 2020, pp. 9641–9650.
- [13] S. Schneider, E. Rusak, L. Eck, O. Bringmann, W. Brendel, and M. Bethge, "Improving robustness against common corruptions by covariate shift adaptation," in *Proc. NeurIPS*, 2020, pp. 11 539–11 551.
- [14] M. Zhang, S. Levine, and C. Finn, "Memo: Test time robustness via adaptation and augmentation," in *Proc. NeurIPS*, 2022.
- [15] P. Latinne, M. Saerens, and C. Decaestecker, "Adjusting the outputs of a classifier to new a priori probabilities may significantly improve classification accuracy: evidence from a multi-class problem in remote sensing," in *Proc. ICML*, 2001, pp. 298–305.
- [16] A. Royer and C. H. Lampert, "Classifier adaptation at prediction time," in *Proc. CVPR*, 2015, pp. 1401–1409.
- [17] Z. Lipton, Y.-X. Wang, and A. Smola, "Detecting and correcting for label shift with black box predictors," in *Proc. ICML*, 2018, pp. 3122–3130.
- [18] G. Csurka, "A comprehensive survey on domain adaptation for visual applications," *Domain Adaptation in Computer Vision Applications*, pp. 1–35, 2017.
- [19] G. Wilson and D. J. Cook, "A survey of unsupervised deep domain adaptation," *ACM Transactions on Intelligent Systems and Technology*, vol. 11, no. 5, pp. 1–46, 2020.
- [20] T. Šipka, M. Šulc, and J. Matas, "The hitchhiker's guide to prior-shift adaptation," in *Proc. WACV*, 2022, pp. 1516–1524.
- [21] Y. Fang, P.-T. Yap, W. Lin, H. Zhu, and M. Liu, "Source-free unsupervised domain adaptation: A survey," *arXiv preprint arXiv:2301.00265*, 2023.
- [22] Z. Yu, J. Li, Z. Du, L. Zhu, and H. T. Shen, "A comprehensive survey on source-free domain adaptation," *arXiv preprint arXiv:2302.11803*, 2023.
- [23] Y. Liu, W. Zhang, J. Wang, and J. Wang, "Data-free knowledge transfer: A survey," *arXiv preprint arXiv:2112.15278*, 2021.
- [24] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2009.
- [25] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan, "A theory of learning from different domains," *Machine Learning*, vol. 79, pp. 151–175, 2010.
- [26] T. Joachims, "Transductive inference for text classification using support vector machines," in *Proc. ICML*, 1999, pp. 200–209.
- [27] J. G. Moreno-Torres, T. Raeder, R. Alaiz-Rodríguez, N. V. Chawla, and F. Herrera, "A unifying view on dataset shift in classification," *Pattern Recognition*, vol. 45, no. 1, pp. 521–530, 2012.
- [28] K. Bousmalis, N. Silberman, D. Dohan, D. Erhan, and D. Krishnan, "Unsupervised pixel-level domain adaptation with generative adversarial networks," in *Proc. CVPR*, 2017, pp. 3722–3731.
- [29] J. Hoffman, E. Tzeng, T. Park, J.-Y. Zhu, P. Isola, K. Saenko, A. Efros, and T. Darrell, "Cycada: Cycle-consistent adversarial domain adaptation," in *Proc. ICML*, 2018, pp. 1989–1998.
- [30] M. Long, Y. Cao, J. Wang, and M. Jordan, "Learning transferable features with deep adaptation networks," in *Proc. ICML*, 2015, pp. 97–105.
- [31] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," in *Proc. ICML*, 2015, pp. 1180–1189.
- [32] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *Proc. CVPR*, 2017, pp. 7167–7176.

- [33] M. Chen, H. Xue, and D. Cai, "Domain adaptation for semantic segmentation with maximum squares loss," in *Proc. ICCV*, 2019, pp. 2090–2099.
- [34] S. Cui, S. Wang, J. Zhuo, L. Li, Q. Huang, and Q. Tian, "Towards discriminability and diversity: Batch nuclear-norm maximization under label insufficient situations," in *Proc. CVPR*, 2020, pp. 3941–3950.
- [35] Y. Jin, X. Wang, M. Long, and J. Wang, "Minimum class confusion for versatile domain adaptation," in *Proc. ECCV*, 2020, pp. 464–480.
- [36] M. Saerens, P. Latinne, and C. Decaestecker, "Adjusting the outputs of a classifier to new a priori probabilities: a simple procedure," *Neural Computation*, vol. 14, no. 1, pp. 21–41, 2002.
- [37] K. Azizzadenesheli, A. Liu, F. Yang, and A. Anandkumar, "Regularized learning for domain adaptation under label shifts," in *Proc. ICLR*, 2019.
- [38] Y. Luo, P. Liu, T. Guan, J. Yu, and Y. Yang, "Adversarial style mining for one-shot unsupervised domain adaptation," in *Proc. NeurIPS*, 2020, pp. 20 612–20 623.
- [39] Z. Wan, L. Li, H. Li, H. He, and Z. Ni, "One-shot unsupervised domain adaptation for object detection," in *Proc. IJCNN*, 2020, pp. 1–8.
- [40] T. Varsavsky, M. Orbes-Arteaga, C. H. Sudre, M. S. Graham, P. Nachev, and M. J. Cardoso, "Test-time unsupervised domain adaptation," in *Proc. MICCAI*, 2020, pp. 428–436.
- [41] R. A. Marsden, M. Döbler, and B. Yang, "Gradual test-time adaptation by self-training and style transfer," *arXiv preprint arXiv:2208.07736*, 2022.
- [42] B. Zhang, R. Shao, J. Du, and P. Yuen, "Mixup for test-time training," *arXiv preprint arXiv:2210.01640*, 2022.
- [43] J. Moon, D. Das, and C. G. Lee, "Multi-step online unsupervised domain adaptation," in *Proc. ICASSP*, 2020, pp. 41 172–41 576.
- [44] L. Yang, M. Gao, Z. Chen, R. Xu, A. Shrivastava, and C. Ramaiah, "Burn after reading: Online adaptation for cross-domain streaming data," in *Proc. ECCV*, 2022, pp. 404–422.
- [45] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," *IEEE Signal Processing Magazine*, vol. 37, no. 3, pp. 50–60, 2020.
- [46] X. Peng, Z. Huang, Y. Zhu, and K. Saenko, "Federated adversarial domain adaptation," in *Proc. ICLR*, 2020.
- [47] H.-Z. Feng, Z. You, M. Chen, T. Zhang, M. Zhu, F. Wu, C. Wu, and W. Chen, "Kd3a: Unsupervised multi-source decentralized domain adaptation via knowledge distillation," in *Proc. ICML*, 2021, pp. 3274–3283.
- [48] G. Wu and S. Gong, "Collaborative optimization and aggregation for decentralized domain generalization and adaptation," in *Proc. ICCV*, 2021, pp. 6484–6493.
- [49] D. Li, Y. Yang, Y.-Z. Song, and T. M. Hospedales, "Learning to generalize: meta-learning for domain generalization," in *Proc. AAAI*, 2018, pp. 3490–3497.
- [50] D. Li, J. Zhang, Y. Yang, C. Liu, Y.-Z. Song, and T. M. Hospedales, "Episodic training for domain generalization," in *Proc. ICCV*, 2019, pp. 1446–1455.
- [51] F. M. Carlucci, A. D'Innocente, S. Bucci, B. Caputo, and T. Tommasi, "Domain generalization by solving jigsaw puzzles," in *Proc. CVPR*, 2019, pp. 2229–2238.
- [52] J. Wang, C. Lan, C. Liu, Y. Ouyang, T. Qin, W. Lu, Y. Chen, W. Zeng, and P. Yu, "Generalizing to unseen domains: A survey on domain generalization," *IEEE Transactions on Knowledge and Data Engineering*, 2022.
- [53] I. Kuzborskij, "Theory and algorithms for hypothesis transfer learning," Ph.D. dissertation, École polytechnique fédérale de Lausanne (EPFL), 2018.
- [54] J. Yang, R. Yan, and A. G. Hauptmann, "Cross-domain video concept detection using adaptive svms," in *Proc. ACM-MM*, 2007, pp. 188–197.
- [55] Y. Aytaç and A. Zisserman, "Tabula rasa: Model transfer for object category detection," in *Proc. ICCV*, 2011, pp. 2252–2259.
- [56] T. Tommasi, F. Orabona, and B. Caputo, "Learning categories from few examples with multi model knowledge transfer," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 5, pp. 928–941, 2013.
- [57] I. Kuzborskij and F. Orabona, "Stability and hypothesis transfer learning," in *Proc. ICML*, 2013, pp. 942–950.
- [58] F. Orabona, C. Castellini, B. Caputo, A. E. Fiorilla, and G. Sandini, "Model adaptation with least-squares svm for adaptive hand prosthetics," in *Proc. ICRA*, 2009, pp. 2897–2903.
- [59] S. M. Ahmed, A. R. Lejbolle, R. Panda, and A. K. Roy-Chowdhury, "Camera on-boarding for person re-identification using hypothesis transfer learning," in *Proc. CVPR*, 2020, pp. 12 144–12 153.
- [60] S. Ao, X. Li, and C. Ling, "Fast generalized distillation for semi-supervised domain adaptation," in *Proc. AAAI*, 2017, pp. 1719–1725.
- [61] A. R. Nelakurthi, R. Maciejewski, and J. He, "Source free domain adaptation using an off-the-shelf classifier," in *Proc. IEEE BigData*, 2018, pp. 140–145.
- [62] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Proc. NeurIPS*, 2014, pp. 3320–3328.
- [63] M. De Lange, R. Aljundi, M. Masana, S. Parisot, X. Jia, A. Leonardis, G. Slabaugh, and T. Tuytelaars, "A continual learning survey: Defying forgetting in classification tasks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 7, pp. 3366–3385, 2021.
- [64] G. M. van de Ven, T. Tuytelaars, and A. S. Tolias, "Three types of incremental learning," *Nature Machine Intelligence*, pp. 1–13, 2022.
- [65] J. N. Kundu, R. M. Venkatesh, N. Venkat, A. Revanur, and R. V. Babu, "Class-incremental domain adaptation," in *Proc. ECCV*, 2020, pp. 53–69.
- [66] A. K. Ambastha and L. T. Yun, "Adversarial learning networks: Source-free unsupervised domain incremental learning," *arXiv preprint arXiv:2301.12054*, 2023.
- [67] H. Shin, J. K. Lee, J. Kim, and J. Kim, "Continual learning with deep generative replay," in *Proc. NeurIPS*, 2017, pp. 2994–3003.
- [68] J. Smith, Y.-C. Hsu, J. Balloch, Y. Shen, H. Jin, and Z. Kira, "Always be dreaming: A new approach for data-free class-incremental learning," in *Proc. ICCV*, 2021, pp. 9374–9384.
- [69] S.-A. Rebuffi, A. Kolesnikov, G. Sperl, and C. H. Lampert, "icarl: Incremental classifier and representation learning," in *Proc. CVPR*, 2017, pp. 2001–2010.
- [70] D. Rolnick, A. Ahuja, J. Schwarz, T. P. Lillicrap, and G. Wayne, "Experience replay for continual learning," in *Proc. NeurIPS*, 2019, pp. 350–360.
- [71] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, D. Hassabis, C. Clopath, D. Kumaran, and R. Hadsell, "Overcoming catastrophic forgetting in neural networks," *Proceedings of the National Academy of Sciences*, vol. 114, no. 13, pp. 3521–3526, 2017.
- [72] F. Zenke, B. Poole, and S. Ganguli, "Continual learning through synaptic intelligence," in *Proc. ICML*, 2017, pp. 3987–3995.
- [73] A. Carta, A. Cossu, V. Lomonaco, and D. Bacciu, "Ex-model: Continual learning from a stream of trained models," in *Proc. CVPR Workshops*, 2022.
- [74] T. Hospedales, A. Antoniou, P. Micaelli, and A. Storkey, "Meta-learning in neural networks: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 9, pp. 5149–5169, 2021.
- [75] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proc. ICML*, 2017, pp. 1126–1135.
- [76] J. Gou, B. Yu, S. J. Maybank, and D. Tao, "Knowledge distillation: A survey," *International Journal of Computer Vision*, vol. 129, no. 6, pp. 1789–1819, 2021.
- [77] P. Micaelli and A. Storkey, "Zero-shot knowledge transfer via adversarial belief matching," in *Proc. NeurIPS*, 2019, pp. 9551–9561.
- [78] G. Fang, J. Song, C. Shen, X. Wang, D. Chen, and M. Song, "Data-free adversarial distillation," *arXiv preprint arXiv:1912.11006*, 2019.
- [79] Y. Liu, W. Zhang, and J. Wang, "Zero-shot adversarial quantization," in *Proc. CVPR*, 2021, pp. 1512–1521.
- [80] G. K. Nayak, K. R. Mopuri, V. Shaj, V. B. Radhakrishnan, and A. Chakraborty, "Zero-shot knowledge distillation in deep networks," in *Proc. ICML*, 2019, pp. 4743–4751.
- [81] H. Chen, Y. Wang, C. Xu, Z. Yang, C. Liu, B. Shi, C. Xu, C. Xu, and Q. Tian, "Data-free learning of student networks," in *Proc. ICCV*, 2019, pp. 3514–3522.
- [82] H. Yin, P. Molchanov, J. M. Alvarez, Z. Li, A. Mallya, D. Hoiem, N. K. Jha, and J. Kautz, "Dreaming to distill: Data-free knowledge transfer via deepinversion," in *Proc. CVPR*, 2020, pp. 8715–8724.
- [83] Z. Wang, "Zero-shot knowledge distillation from a decision-based black-box model," in *Proc. ICML*, 2021, pp. 10 675–10 685.

- [84] L. Jing and Y. Tian, "Self-supervised visual feature learning with deep neural networks: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 11, pp. 4037–4058, 2020.
- [85] R. Zhang, P. Isola, and A. A. Efros, "Colorful image colorization," in *Proc. ECCV*, 2016, pp. 649–666.
- [86] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in *Proc. CVPR*, 2016, pp. 2536–2544.
- [87] S. Gidaris, P. Singh, and N. Komodakis, "Unsupervised representation learning by predicting image rotations," in *Proc. ICLR*, 2018.
- [88] C. Doersch, A. Gupta, and A. A. Efros, "Unsupervised visual representation learning by context prediction," in *Proc. ICCV*, 2015, pp. 1422–1430.
- [89] N. Srivastava, E. Mansimov, and R. Salakhudinov, "Unsupervised learning of video representations using lstms," in *Proc. ICML*, 2015, pp. 843–852.
- [90] M. Caron, P. Bojanowski, A. Joulin, and M. Douze, "Deep clustering for unsupervised learning of visual features," in *Proc. ECCV*, 2018, pp. 132–149.
- [91] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin, "Unsupervised learning of visual features by contrasting cluster assignments," in *Proc. NeurIPS*, 2020, pp. 9912–9924.
- [92] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proc. CVPR*, 2020, pp. 9729–9738.
- [93] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. ICML*, 2020, pp. 1597–1607.
- [94] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. Richemond, E. Buchatskaya, C. Doersch, B. Avila Pires, Z. Guo, M. Gheshlaghi Azar, B. Piot, K. Kavukcuoglu, R. Munos, and M. Valko, "Bootstrap your own latent - a new approach to self-supervised learning," in *Proc. NeurIPS*, 2020, pp. 21 271–21 284.
- [95] J. D. M.-W. C. Kenton and L. K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proc. NAACL*, 2019, pp. 4171–4186.
- [96] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," in *Proc. NeurIPS*, 2020, pp. 12 449–12 460.
- [97] Y. You, T. Chen, Y. Sui, T. Chen, Z. Wang, and Y. Shen, "Graph contrastive learning with augmentations," in *Proc. NeurIPS*, 2020, pp. 5812–5823.
- [98] J. Liang, D. Hu, Y. Wang, R. He, and J. Feng, "Source data-absent unsupervised domain adaptation through hypothesis transfer and labeling transfer," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 11, pp. 8602–8617, 2022.
- [99] Y. Chen, M. Mancini, X. Zhu, and Z. Akata, "Semi-supervised and unsupervised deep visual learning: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.
- [100] Y. Grandvalet and Y. Bengio, "Semi-supervised learning by entropy minimization," in *Proc. NeurIPS*, 2004, pp. 529–536.
- [101] D.-H. Lee, "Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks," in *Proc. ICML Workshops*, 2013.
- [102] T. Miyato, S.-i. Maeda, M. Koyama, and S. Ishii, "Virtual adversarial training: a regularization method for supervised and semi-supervised learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 8, pp. 1979–1993, 2018.
- [103] D. Berthelot, N. Carlini, I. Goodfellow, A. Oliver, N. Papernot, and C. Raffel, "Mixmatch: a holistic approach to semi-supervised learning," in *Proc. NeurIPS*, 2019, pp. 5049–5059.
- [104] K. Sohn, D. Berthelot, N. Carlini, Z. Zhang, H. Zhang, C. A. Raffel, E. D. Cubuk, A. Kurakin, and C.-L. Li, "Fixmatch: Simplifying semi-supervised learning with consistency and confidence," in *Proc. NeurIPS*, 2020, pp. 596–608.
- [105] S. Laine and T. Aila, "Temporal ensembling for semi-supervised learning," in *Proc. ICLR*, 2017.
- [106] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," in *Proc. NeurIPS*, 2017, pp. 1195–1204.
- [107] A. Iscen, G. Tolias, Y. Avrithis, and O. Chum, "Label propagation for deep semi-supervised learning," in *Proc. CVPR*, 2019, pp. 5070–5079.
- [108] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of Big Data*, vol. 6, no. 1, pp. 1–48, 2019.
- [109] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. NeurIPS*, 2012, pp. 1106–1114.
- [110] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. CVPR*, 2016, pp. 770–778.
- [111] L. Smith and Y. Gal, "Understanding measures of uncertainty for adversarial example detection," in *Proc. UAI*, 2018, pp. 560–569.
- [112] C. Guo, M. Rana, M. Cisse, and L. van der Maaten, "Countering adversarial images using input transformations," in *Proc. ICLR*, 2018.
- [113] J. C. Pérez, M. Alfara, G. Jeanneret, L. Rueda, A. Thabet, B. Ghanem, and P. Arbeláez, "Enhancing adversarial robustness via test-time transformation ensembling," in *Proc. ICCV*, 2021, pp. 81–91.
- [114] I. Kim, Y. Kim, and S. Kim, "Learning loss for test-time augmentation," in *Proc. NeurIPS*, 2020, pp. 4163–4174.
- [115] D. Shanmugam, D. Blalock, G. Balakrishnan, and J. Guttag, "Better aggregation in test-time augmentation," in *Proc. ICCV*, 2021, pp. 1214–1223.
- [116] Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," in *Proc. ICML*, 2016, pp. 1050–1059.
- [117] J. Huang, A. J. Smola, A. Gretton, K. M. Borgwardt, and B. Scholkopf, "Correcting sample selection bias by unlabeled data," in *Proc. NeurIPS*, 2006, pp. 601–608.
- [118] B. Chidlovskii, S. Clinchant, and G. Csurka, "Domain adaptation in the absence of source domain data," in *Proc. KDD*, 2016, pp. 451–460.
- [119] S. Clinchant, B. Chidlovskii, and G. Csurka, "Transductive adaptation of black box predictions," in *Proc. ACL*, 2016, pp. 326–331.
- [120] T. van Laarhoven and E. Marchiori, "Unsupervised domain adaptation with random walks on target labelings," *arXiv preprint arXiv:1706.05335*, 2017.
- [121] J. Liang, R. He, Z. Sun, and T. Tan, "Distant supervised centroid shift: A simple and efficient approach to visual domain adaptation," in *Proc. CVPR*, 2019, pp. 2975–2984.
- [122] S. Qu, G. Chen, J. Zhang, Z. Li, W. He, and D. Tao, "Bmd: A general class-balanced multicentric dynamic prototype strategy for source-free domain adaptation," in *Proc. ECCV*, 2022, pp. 165–182.
- [123] S. Yang, J. van de Weijer, L. Herranz, and S. Jui, "Exploiting the intrinsic neighborhood structure for source-free domain adaptation," in *Proc. NeurIPS*, 2021, pp. 29 393–29 405.
- [124] W. Chen, L. Lin, S. Yang, D. Xie, S. Pu, Y. Zhuang, and W. Ren, "Self-supervised noisy label learning for source-free unsupervised domain adaptation," in *Proc. IROS*, 2022, pp. 10 185–10 192.
- [125] F. You, J. Li, L. Zhu, Z. Chen, and Z. Huang, "Domain adaptive semantic segmentation without source data," in *Proc. ACM-MM*, 2021, pp. 3293–3302.
- [126] Y. Wang, J. Liang, and Z. Zhang, "Source data-free cross-domain semantic segmentation: align, teach and propagate," *arXiv preprint arXiv:2106.11653*, 2022.
- [127] H. Yan, Y. Guo, and C. Yang, "Augmented self-labeling for source-free unsupervised domain adaptation," in *Proc. NeurIPS Workshops*, 2021.
- [128] T. Sun, C. Lu, and H. Ling, "Prior knowledge guided unsupervised domain adaptation," in *Proc. ECCV*, 2022, pp. 639–655.
- [129] S. Yang, Y. Wang, J. van de Weijer, L. Herranz, and S. Jui, "Generalized source-free domain adaptation," in *Proc. ICCV*, 2021, pp. 8978–8987.
- [130] T. Sun, C. Lu, and H. Ling, "Domain adaptation with adversarial training on penultimate activations," in *Proc. AAAI*, 2023.
- [131] P. T. Sivaprasad and F. Fleuret, "Uncertainty reduction for model adaptation in semantic segmentation," in *Proc. CVPR*, 2021, pp. 9613–9623.
- [132] Q. Peng, Z. Ding, L. Lyu, L. Sun, and C. Chen, "Toward better target representation for source-free and black-box domain adaptation," *arXiv preprint arXiv:2208.10531*, 2022.
- [133] D. Chen, D. Wang, T. Darrell, and S. Ebrahimi, "Contrastive test-time adaptation," in *Proc. CVPR*, 2022, pp. 295–305.
- [134] Y. Ding, J. Liang, B. Jiang, A. Zheng, and R. He, "Maps: A noise-robust progressive learning approach for source-free domain adaptive keypoint detection," *arXiv preprint arXiv:2302.04589*, 2023.

- [135] K. Xia, L. Deng, W. Duch, and D. Wu, "Privacy-preserving domain adaptation for motor imagery-based brain-computer interfaces," *IEEE Transactions on Biomedical Engineering*, vol. 69, no. 11, pp. 3365–3376, 2022.
- [136] J. Liang, D. Hu, J. Feng, and R. He, "Umad: Universal model adaptation under domain and category shift," *arXiv preprint arXiv:2112.08553*, 2021.
- [137] X. Li, J. Li, L. Zhu, G. Wang, and Z. Huang, "Imbalanced source-free domain adaptation," in *Proc. ACM-MM*, 2021, pp. 3330–3339.
- [138] Y. Liu, Y. Chen, W. Dai, M. Gou, C.-T. Huang, and H. Xiong, "Source-free domain adaptation with contrastive domain alignment and self-supervised exploration for face anti-spoofing," in *Proc. ECCV*, 2022, pp. 511–528.
- [139] G. K. Nayak, K. R. Mopuri, S. Jain, and A. Chakraborty, "Mining data impressions from deep models as substitute for the unavailable training data," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 11, pp. 8465–8481, 2022.
- [140] Y. Hou and L. Zheng, "Source free domain adaptation with image translation," *arXiv preprint arXiv:2008.07514*, 2020.
- [141] S. Hu, Z. Liao, and Y. Xia, "Prosfa: Prompt learning based source-free domain adaptation for medical image segmentation," *arXiv preprint arXiv:2211.11514*, 2022.
- [142] Z. Zhang, W. Chen, H. Cheng, Z. Li, S. Li, L. Lin, and G. Li, "Divide and contrast: Source-free domain adaptation via adaptive contrastive learning," in *Proc. NeurIPS*, 2022.
- [143] J. Tian, J. Zhang, W. Li, and D. Xu, "Vdm-da: Virtual domain modeling for source data-free domain adaptation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 6, pp. 3749–3760, 2022.
- [144] Z. Qiu, Y. Zhang, H. Lin, S. Niu, Y. Liu, Q. Du, and M. Tan, "Source-free domain adaptation via avatar prototype generation and adaptation," in *Proc. IJCAI*, 2021, pp. 2921–2927.
- [145] J. N. Kundu, S. Bhambri, A. Kulkarni, H. Sarkar, V. Jampani, and R. V. Babu, "Concurrent subsidiary supervision for unsupervised source-free domain adaptation," in *Proc. ECCV*, 2022, pp. 177–194.
- [146] J. E. Van Engelen and H. H. Hoos, "A survey on semi-supervised learning," *Machine Learning*, vol. 109, no. 2, pp. 373–440, 2020.
- [147] Y. Kim, D. Cho, K. Han, P. Panda, and S. Hong, "Domain adaptation without source data," *IEEE Transactions on Artificial Intelligence*, vol. 2, no. 6, pp. 508–518, 2021.
- [148] X. Li, W. Chen, D. Xie, S. Yang, P. Yuan, S. Pu, and Y. Zhuang, "A free lunch for unsupervised domain adaptive object detection without source data," in *Proc. AAAI*, 2021, pp. 8474–8481.
- [149] Q. Peng, "Multi-source and source-private cross-domain learning for visual recognition," Master's thesis, Purdue University, May 2022.
- [150] C. Chen, Q. Liu, Y. Jin, Q. Dou, and P.-A. Heng, "Source-free domain adaptive fundus image segmentation with denoised pseudo-labeling," in *Proc. MICCAI*, 2021, pp. 225–235.
- [151] B. Song, "Source-free few-shot semi-supervised domain adaptation for ct image classification," 2022.
- [152] L. Zhou, M. Ye, and S. Xiao, "Domain adaptation based on source category prototypes," *Neural Computing and Applications*, vol. 34, no. 23, pp. 21 191–21 203, 2022.
- [153] V. Prabhu, S. Khare, D. Kartik, and J. Hoffman, "Augco: Augmentation consistency-guided self-training for source-free domain adaptive semantic segmentation," *arXiv preprint arXiv:2107.10140*, 2022.
- [154] D. Kothandaraman, R. Chandra, and D. Manocha, "Ss-sfda: Self-supervised source-free domain adaptation for road segmentation in hazardous environments," in *Proc. ICCV Workshops*, 2021.
- [155] A. Dasgupta, C. Jawahar, and K. Alahari, "Overcoming label noise for source-free unsupervised video domain adaptation," in *Indian Conference on Computer Vision, Graphics, and Image Processing (ICVGIP)*. ACM, 2022.
- [156] L. Tian, L. Zhou, H. Zhang, Z. Wang, and M. Ye, "Robust self-supervised learning for source-free domain adaptation," *Signal, Image and Video Processing*, pp. 1–9, 2023.
- [157] Y. Hou and L. Zheng, "Visualizing adapted knowledge in domain transfer," in *Proc. CVPR*, 2021, pp. 13 824–13 833.
- [158] R. Wang, Z. Wu, Z. Weng, J. Chen, G.-J. Qi, and Y.-G. Jiang, "Cross-domain contrastive learning for unsupervised domain adaptation," *IEEE Transactions on Multimedia*, 2022, in Press.
- [159] Y. Du, H. Yang, M. Chen, J. Jiang, H. Luo, and C. Wang, "Generation, augmentation, and alignment: A pseudo-source domain based method for source-free domain adaptation," *arXiv preprint arXiv:2109.04015*, 2021.
- [160] G. Yang, Z. Zhong, M. Ding, N. Sebe, and E. Ricci, "Self-training transformer for source-free domain adaptation," *Applied Intelligence*, pp. 1–15, 2022.
- [161] X. Zhao, R. Stanislawski, P. Gardoni, M. Sulowicz, A. Glowacz, G. Krolczyk, and Z. Li, "Adaptive contrastive learning with label consistency for source data free unsupervised domain adaptation," *Sensors*, vol. 22, no. 11, p. 4238, 2022.
- [162] W. Li, M. Cao, and S. Chen, "Jacobian norm for unsupervised source-free domain adaptation," *arXiv preprint arXiv:2204.03467*, 2022.
- [163] J. Pei, Z. Jiang, A. Men, L. Chen, Y. Liu, and Q. Chen, "Uncertainty-induced transferability representation for source-free unsupervised domain adaptation," *arXiv preprint arXiv:2208.13986*, 2022.
- [164] W. Li, K. Fan, and H. Yang, "Teacher-student mutual learning for efficient source-free unsupervised domain adaptation," *Knowledge-Based Systems*, p. 110204, 2022.
- [165] S. Tang, Y. Shi, Z. Ma, J. Li, J. Lyu, Q. Li, and J. Zhang, "Model adaptation through hypothesis transfer with gradual knowledge distillation," in *Proc. IROS*, 2021, pp. 5679–5685.
- [166] I. Diamant, R. H. Jennings, O. Dror, H. V. Habi, and A. Netzer, "Reconciling a centroid-hypothesis conflict in source-free domain adaptation," *arXiv preprint arXiv:2212.03795*, 2022.
- [167] A. M. N. Taufique, C. S. Jahan, and A. Savakis, "Continual unsupervised domain adaptation in data-constrained environments," *IEEE Transactions on Artificial Intelligence*, 2023.
- [168] J. Jiao, H. Li, T. Zhang, and J. Lin, "Source-free adaptation diagnosis for rotating machinery," *IEEE Transactions on Industrial Informatics*, 2022.
- [169] P. Chen and A. J. Ma, "Source-free temporal attentive domain adaptation for video action recognition," in *Proc. ICMR*, 2022, p. 489–497.
- [170] H. Zhang and L. Tian, "Source-free unsupervised domain adaptation via denoising mutual learning," in *International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*, 2022, pp. 1–7.
- [171] Y. Feng, Y. Luo, and J. Yang, "Cross-platform privacy-preserving ct image covid-19 diagnosis based on source-free domain adaptation," *Knowledge-Based Systems*, p. 110324, 2023.
- [172] L. Yuan, M. Erdt, R. Li, and M. Y. Siyal, "Data privacy protection domain adaptation by roughing and finishing stage," *The Visual Computer*, pp. 1–18, 2023.
- [173] Y. Kim, D. Cho, and S. Hong, "Towards privacy-preserving domain adaptation," *IEEE Signal Processing Letters*, vol. 27, pp. 1675–1679, 2020.
- [174] O. Bohdal, D. Li, and T. Hospedales, "Feed-forward source-free domain adaptation via class prototypes," in *Proc. ECCV Workshops*, 2022.
- [175] C. Liu, L. Wang, L. Lyu, C. Sun, X. Wang, and Q. Zhu, "Twofer: Tackling continual domain shift with simultaneous domain generalization and adaptation," in *Proc. ICLR*, 2023.
- [176] S. Tang, Y. Zou, Z. Song, J. Lyu, L. Chen, M. Ye, S. Zhong, and J. Zhang, "Semantic consistency learning on manifold for source data-free unsupervised domain adaptation," *Neural Networks*, vol. 152, pp. 467–478, 2022.
- [177] W. Zhang, L. Shen, and C.-S. Foo, "Co-learning with pre-trained networks improves source-free domain adaptation," *arXiv preprint arXiv:2212.07585*, 2022.
- [178] X. Guan, H. Sun, N. Liu, and H. Zhou, "Polycentric clustering and structural regularization for source-free unsupervised domain adaptation," in *Proc. BMVC*, 2022.
- [179] L. Wang, S. Liang, and F. Gao, "Providing domain specific model via universal no data exchange domain adaptation," in *International Conference on Electronic Technology and Information Science (ICETIS)*, 2021.
- [180] J. Lee, D. Jung, J. Yim, and S. Yoon, "Confidence score for source-free unsupervised domain adaptation," in *Proc. ICML*, 2022, pp. 12 365–12 377.
- [181] J. Lee and G. Lee, "Feature alignment by uncertainty and self-training for source-free unsupervised domain adaptation," *arXiv preprint arXiv:2208.14888*, 2022.
- [182] T. Han, X. Gong, F. Feng, J. Zhang, Z. Sun, and Y. Zhang, "Privacy-preserving multi-source domain adaptation for medical data," *IEEE Journal of Biomedical and Health Informatics*, 2022.

- [183] H. Yan and Y. Guo, "Dual moving average pseudo-labeling for source-free inductive domain adaptation," in *Proc. BMVC*, 2022.
- [184] F. Wang, Z. Han, Y. Gong, and Y. Yin, "Exploring domain-invariant parameters for source free domain adaptation," in *Proc. CVPR*, 2022, pp. 7151–7160.
- [185] S. Tang, Y. Yang, Z. Ma, N. Hendrich, F. Zeng, S. S. Ge, C. Zhang, and J. Zhang, "Nearest neighborhood-based deep clustering for source data-absent unsupervised domain adaptation," *arXiv preprint arXiv:2107.12585*, 2021.
- [186] Z. Cao, Z. Li, X. Guo, and G. Wang, "Towards cross-environment human activity recognition based on radar without source data," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 11, pp. 11 843–11 854, 2021.
- [187] Y. Ding, L. Sheng, J. Liang, A. Zheng, and R. He, "Proxymix: Proxy-based mixup training with label refinery for source-free domain adaptation," *arXiv preprint arXiv:2205.14566*, 2022.
- [188] J. Dong, Z. Fang, A. Liu, G. Sun, and T. Liu, "Confident anchor-induced multi-source free domain adaptation," in *Proc. NeurIPS*, 2021, pp. 2848–2860.
- [189] J. Liang, D. Hu, and J. Feng, "Domain adaptation with auxiliary target domain-oriented classifier," in *Proc. CVPR*, 2021, pp. 16 632–16 642.
- [190] Y. Gao, X. Shi, Y. Zhu, H. Wang, Z. Tang, X. Zhou, M. Li, and D. N. Metaxas, "Visual prompt tuning for test-time domain adaptation," *arXiv preprint arXiv:2210.04831*, 2022.
- [191] M. Litrico, A. Del Bue, and P. Morerio, "Guiding pseudo-labels with uncertainty estimation for test-time adaptation," in *Proc. CVPR*, 2023.
- [192] Y. Kim, J. Yim, J. Yun, and J. Kim, "Nlnl: Negative learning for noisy labels," in *Proc. ICCV*, 2019, pp. 101–110.
- [193] X. Luo, W. Chen, Y. Tan, C. Li, Y. He, and X. Jia, "Exploiting negative learning for implicit pseudo label rectification in source-free domain adaptive semantic segmentation," *arXiv preprint arXiv:2106.12123*, 2021.
- [194] W. Xie, M. Li, J. Wu, Y. Huang, Y. Bu, J. You, and X. Liu, "Learn complementary pseudo-label for source-free domain adaptive medical segmentation," in *Proc. NeurIPS Workshops*, 2022.
- [195] W. Ahmed, P. Morerio, and V. Murino, "Cleaning noisy labels by negative ensemble learning for source-free unsupervised domain adaptation," in *Proc. WACV*, 2022, pp. 1616–1625.
- [196] B. Xie, L. Yuan, S. Li, C. H. Liu, and X. Cheng, "Towards fewer annotations: Active learning via region impurity and prediction uncertainty for domain adaptive semantic segmentation," in *Proc. CVPR*, 2022, pp. 8068–8078.
- [197] C.-Y. Yang, Y.-J. Kuo, and C.-T. Hsu, "Source free domain adaptation for semantic segmentation via distribution transfer and adaptive class-balanced self-training," in *Proc. ICME*, 2022, pp. 1–6.
- [198] Y. Zou, Z. Yu, B. Kumar, and J. Wang, "Unsupervised domain adaptation for semantic segmentation via class-balanced self-training," in *Proc. ECCV*, 2018, pp. 289–305.
- [199] J. Huang, D. Guan, A. Xiao, and S. Lu, "Model adaptation: Historical contrastive learning for unsupervised domain adaptation without source data," in *Proc. NeurIPS*, 2021, pp. 3635–3649.
- [200] Y. Zhao, Z. Zhong, Z. Luo, G. H. Lee, and N. Sebe, "Source-free open compound domain adaptation in semantic segmentation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 10, pp. 7019–7032, 2022.
- [201] H. Zhang, Y. Zhang, K. Jia, and L. Zhang, "Unsupervised domain adaptation of black-box source models," in *Proc. BMVC*, 2021.
- [202] M. Boudiaf, T. Denton, B. van Merriënboer, V. Dumoulin, and E. Triantafillou, "In search for a generalizable method for source free domain adaptation," *arXiv preprint arXiv:2302.06658*, 2023.
- [203] Z. Xu, D. Lu, Y. Wang, J. Luo, D. Wei, Y. Zheng, and R. K.-y. Tong, "Denoising for relaxing: Unsupervised domain adaptive fundus image segmentation without source data," in *Proc. MICCAI*, 2022, pp. 214–224.
- [204] N. Karim, N. C. Mithun, A. Rajvanshi, H.-p. Chiu, S. Samarasera, and N. Rahnavard, "C-sfda: A curriculum learning aided self-training framework for efficient source free domain adaptation," in *Proc. CVPR*, 2023.
- [205] J. Yang, S. Shi, Z. Wang, H. Li, and X. Qi, "St3d: Self-training for unsupervised domain adaptation on 3d object detection," in *Proc. CVPR*, 2021, pp. 10 368–10 378.
- [206] L. Yi, G. Xu, P. Xu, J. Li, R. Pu, C. Ling, A. I. McLeod, and B. Wang, "When source-free domain adaptation meets learning with noisy labels," in *Proc. ICLR*, 2023.
- [207] J. Liang, D. Hu, J. Feng, and R. He, "Dine: Domain adaptation from single and multiple black-box predictors," in *Proc. CVPR*, 2022, pp. 8003–8013.
- [208] T. Panagiotakopoulos, P. L. Dovesi, L. Härenstam-Nielsen, and M. Poggi, "Online domain adaptation for semantic segmentation in ever-changing conditions," in *Proc. ECCV*, 2022, pp. 128–146.
- [209] D. Hegde, V. Sindagi, V. Kilic, A. B. Cooper, M. Foster, and V. M. Patel, "Uncertainty-aware mean teacher for source-free unsupervised domain adaptive 3d object detection," *arXiv preprint arXiv:2109.14651*, 2021.
- [210] T.-D. Truong, P. Helton, A. Moustafa, J. D. Cothren, and K. Luu, "Conda: Continual unsupervised domain adaptation learning in visual perception for self-driving cars," *arXiv preprint arXiv:2212.00621*, 2022.
- [211] Z. Li, R. Togo, T. Ogawa, and M. Haseyama, "Union-set multi-source model adaptation for semantic segmentation," in *Proc. ECCV*, 2022, pp. 579–595.
- [212] —, "Improving model adaptation for semantic segmentation by learning model-invariant features with multiple source-domain models," in *Proc. ICIP*, 2022, pp. 421–425.
- [213] J. N. Kundu, A. Kulkarni, A. Singh, V. Jampani, and R. V. Babu, "Generalize then adapt: Source-free domain adaptive semantic segmentation," in *Proc. ICCV*, 2021, pp. 7046–7056.
- [214] C. Li, W. Chen, X. Luo, Y. He, and Y. Tan, "Adaptive pseudo labeling for source-free domain adaptation in medical image segmentation," in *Proc. ICASSP*, 2022, pp. 1091–1095.
- [215] M. Ye, J. Zhang, J. Ouyang, and D. Yuan, "Source data-free unsupervised domain adaptation for semantic segmentation," in *Proc. ACM-MM*, 2021, pp. 2233–2242.
- [216] L. Xiong, M. Ye, D. Zhang, Y. Gan, and Y. Liu, "Source data-free domain adaptation for a faster r-cnn," *Pattern Recognition*, vol. 124, p. 108436, 2022.
- [217] H.-W. Yeh, T. Westfechtel, J.-B. Huang, and T. Harada, "Boosting source-free domain adaptation via confidence-based subsets feature alignment," in *Proc. ICPR*, 2022, pp. 2857–2863.
- [218] V. Kumar, R. Lal, H. Patil, and A. Chakraborty, "Conmix for source-free single and multi-target domain adaptation," in *Proc. WACV*, 2023, pp. 4178–4188.
- [219] X. Luo, W. Chen, C. Li, B. Zhou, and Y. Tan, "Multi-level consistency learning for source-free model adaptation," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 12 419–12 426, 2022.
- [220] B. Deng, Y. Zhang, H. Tang, C. Ding, and K. Jia, "On universal black-box domain adaptation," *arXiv preprint arXiv:2104.04665*, 2021.
- [221] Q. Tian, S. Peng, and T. Ma, "Source-free unsupervised domain adaptation with trusted pseudo samples," *ACM Transactions on Intelligent Systems and Technology*, 2022.
- [222] R. Müller, S. Kornblith, and G. E. Hinton, "When does label smoothing help?" in *Proc. NeurIPS*, 2019, pp. 4694–4703.
- [223] S. Paul, A. Khurana, and G. Aggarwal, "Unsupervised adaptation of semantic segmentation models without source data," *arXiv preprint arXiv:2112.02359*, 2021.
- [224] M. Chen, W. Chen, S. Yang, J. Song, X. Wang, L. Zhang, Y. Yan, D. Qi, Y. Zhuang, D. Xie, and S. Pu, "Learning domain adaptive object detection with probabilistic teacher," in *Proc. ICML*, 2022, pp. 3040–3055.
- [225] X. Liu, F. Xing, G. El Fakhri, and J. Woo, "Memory consistent unsupervised off-the-shelf model adaptation for source-relaxed medical image segmentation," *Medical Image Analysis*, p. 102641, 2022.
- [226] Z. Song, L. Chen, H. Sun, and G. Kou, "Ss8: Source data-free domain adaptation via deep clustering with weighted self-labelling," in *IEEE International Conference on Networking, Sensing and Control (ICNSC)*, 2022, pp. 1–6.
- [227] E. Rusak, S. Schneider, G. Pachitariu, L. Eck, P. V. Gehler, O. Bringmann, W. Brendel, and M. Bethge, "If your data distribution shifts, use self-learning," *Transactions on Machine Learning Research*, 2022.
- [228] Y. Wang, X. Ma, Z. Chen, Y. Luo, J. Yi, and J. Bailey, "Symmetric cross entropy for robust learning with noisy labels," in *Proc. ICCV*, 2019, pp. 322–330.
- [229] X. Ma, H. Huang, Y. Wang, S. Romano, S. Erfani, and J. Bailey, "Normalized loss functions for deep learning with noisy labels," in *Proc. ICML*, 2020, pp. 6543–6553.
- [230] K. Thopalli, P. Turaga, and J. J. Thiagarajan, "Domain alignment meets fully test-time adaptation," *arXiv preprint arXiv:2207.04185*, 2022.

- [231] H. Yao, D.-l. Zhu, B. Jiang, and P. Yu, "Negative log likelihood ratio loss for deep neural network classification," in *Proceedings of the Future Technologies Conference (FTC)*, 2020, pp. 276–282.
- [232] X. Yang, Z. Song, I. King, and Z. Xu, "A survey on deep semi-supervised learning," *IEEE Transactions on Knowledge and Data Engineering*, 2022.
- [233] T. DeVries and G. W. Taylor, "Improved regularization of convolutional neural networks with cutout," *arXiv preprint arXiv:1708.04552*, 2017.
- [234] E. D. Cubuk, B. Zoph, J. Shlens, and Q. V. Le, "RandAugment: Practical automated data augmentation with a reduced search space," in *Proc. CVPR Workshops*, 2020.
- [235] Q. Xie, Z. Dai, E. Hovy, T. Luong, and Q. Le, "Unsupervised data augmentation for consistency training," in *Proc. NeurIPS*, 2020, pp. 6256–6268.
- [236] C. Wu, G. Cao, Y. Li, X. Xi, W. Cao, and H. Wang, "When source-free domain adaptation meets label propagation," *arXiv preprint arXiv:2301.08413*, 2023.
- [237] X. Wang, J. Zhuo, S. Cui, and S. Wang, "Learning invariant representation with consistency and diversity for semi-supervised source hypothesis transfer," *arXiv preprint arXiv:2107.03008*, 2021.
- [238] Y. Yang and S. Soatto, "Fda: Fourier domain adaptation for semantic segmentation," in *Proc. CVPR*, 2020, pp. 4085–4095.
- [239] C. Yang, X. Guo, Z. Chen, and Y. Yuan, "Source free domain adaptation for medical image segmentation with fourier style mining," *Medical Image Analysis*, vol. 79, p. 102457, 2022.
- [240] S. Sinha, P. Gehler, F. Locatello, and B. Schiele, "Test: Test-time self-training under distribution shift," in *Proc. WACV*, 2023, pp. 2759–2769.
- [241] Z. Feng, C. Xu, and D. Tao, "Open-set hypothesis transfer with semantic consistency," *IEEE Transactions on Image Processing*, vol. 30, pp. 6473–6484, 2021.
- [242] C. Zhou, W. Zhang, H. Chen, and L. Chen, "Domain adaptation for medical image classification without source data," in *Proc. BIMB*, 2022, pp. 2224–2230.
- [243] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," in *Proc. ICLR*, 2018.
- [244] D. Kothandaraman, S. Shekhar, A. Sancheti, M. Ghuman, T. Shukla, and D. Manocha, "Salad: source-free active label-agnostic domain adaptation for classification, segmentation and detection," in *Proc. WACV*, 2023, pp. 382–391.
- [245] B. Zhang, X. Zhang, Y. Liu, L. Cheng, and Z. Li, "Matching distributions between model and data: Cross-domain knowledge distillation for unsupervised domain adaptation," in *Proc. ACL*, 2021, pp. 5423–5433.
- [246] X. Liu, C. Yoo, F. Xing, C.-C. J. Kuo, G. E. Fakhri, J.-W. Kang, and J. Woo, "Unsupervised black-box model domain adaptation for brain tumor segmentation," *Frontiers in Neuroscience*, vol. 16, p. 341, 2022.
- [247] A. Conti, P. Rota, Y. Wang, and E. Ricci, "Cluster-level pseudo-labelling for source-free cross-domain facial expression recognition," in *Proc. BMVC*, 2022.
- [248] A. Bigalke, L. Hansen, J. Diesel, C. Hennigs, P. Rostalski, and M. P. Heinrich, "Anatomy-guided domain adaptation for 3d in-bed human pose estimation," *arXiv preprint arXiv:2211.12193*, 2022.
- [249] V. VS, P. Oza, V. A. Sindagi, and V. M. Patel, "Mixture of teacher experts for source-free domain adaptive object detection," in *Proc. ICIP*, 2022, pp. 3606–3610.
- [250] Q. Lao, X. Jiang, and M. Havaei, "Hypothesis disparity regularized mutual information maximization," in *Proc. AAAI*, 2021, pp. 8243–8251.
- [251] Z. Zong, J. He, L. Zhang, and H. Huan, "Domain gap estimation for source free unsupervised domain adaptation with many classifiers," *arXiv preprint arXiv:2112.00955*, 2022.
- [252] H. Xu and U. Kang, "Transfer alignment network for double blind unsupervised domain adaptation," 2020.
- [253] D. Zhang, M. Ye, L. Xiong, S. Li, and X. Li, "Source-style transferred mean teacher for source-data free object detection," in *ACM Multimedia Asia*, 2021, pp. 1–8.
- [254] Z. Wang, "An unsupervised domain adaptation method for compressed video quality enhancement," in *International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*, 2022, pp. 1–5.
- [255] L. Xiong, M. Ye, D. Zhang, Y. Gan, X. Li, and Y. Zhu, "Source data-free domain adaptation of object detector through domain-specific perturbation," *International Journal of Intelligent Systems*, vol. 36, no. 8, pp. 3746–3766, 2021.
- [256] V. VS, P. Oza, and V. M. Patel, "Towards online domain adaptive object detection," in *Proc. WACV*, 2023, pp. 478–488.
- [257] Q. Chu, S. Li, G. Chen, K. Li, and X. Li, "Adversarial alignment for source free object detection," *arXiv preprint arXiv:2301.04265*, 2023.
- [258] W. Zhao, B. Chen, W. Chen, S. Yang, D. Xie, S. Pu, and Y. Zhuang, "1st place solution for eccv 2022 ood-cv challenge object detection track," *arXiv preprint arXiv:2301.04796*, 2023.
- [259] S. Zhang, L. Zhang, and Z. Liu, "Refined pseudo labeling for source-free domain adaptive object detection," in *Proc. ICASSP*, 2023.
- [260] V. VS, J. M. J. Valanarasu, and V. M. Patel, "Target and task specific source-free domain adaptive image segmentation," *arXiv preprint arXiv:2203.15792*, 2022.
- [261] V. VS, P. Oza, and V. M. Patel, "Instance relation graph guided source-free domain adaptive object detection," *arXiv preprint arXiv:2203.15793*, 2022.
- [262] S. Li, M. Ye, X. Zhu, L. Zhou, and L. Xiong, "Source-free object detection by learning to overlook domain style," in *Proc. CVPR*, 2022, pp. 8014–8023.
- [263] H. Yang, Z. Yang, A. Hu, C. Liu, T. J. Cui, and J. Miao, "Source-free domain adaptive detection of concealed objects in passive millimeter-wave images," *IEEE Transactions on Instrumentation and Measurement*, 2023.
- [264] D. Wang, S. Liu, S. Ebrahimi, E. Shelhamer, and T. Darrell, "On-target adaptation," *arXiv preprint arXiv:2109.01087*, 2021.
- [265] B. Chen, K. Thandiackal, P. Pati, and O. Goksel, "Generative appearance replay for continual unsupervised domain adaptation," *arXiv preprint arXiv:2301.01211*, 2023.
- [266] Y. Liu, C. Zhao, Y. Lu, W. Xing, and X. Qiao, "A source free domain adaptation model based on adversarial learning for image classification," *Applied Intelligence*, pp. 1–14, 2022.
- [267] L. Yuan, M. Erdt, and L. Wang, "Source protection domain adaptation by gumbel-min-max entropy minimization," 2022.
- [268] M. Ishii and M. Sugiyama, "Source-free domain adaptation via distributional alignment by matching batch normalization statistics," *arXiv preprint arXiv:2101.10842*, 2021.
- [269] S. M. Ahmed, S. Lohit, K.-C. Peng, M. Jones, and A. K. Roy-Chowdhury, "Cross-modal knowledge transfer without task-relevant source data," in *Proc. ECCV*, 2022, pp. 111–127.
- [270] S. Li, S. Zhao, J. Hong, and W. Chen, "Source-free unsupervised adaptive segmentation for knee joint mri," 2022.
- [271] X. Liu, F. Xing, C. Yang, G. El Fakhri, and J. Woo, "Adapting off-the-shelf source segmenter for target medical image segmentation," in *Proc. MICCAI*, 2021, pp. 549–559.
- [272] Y. Liu, P. Kothari, B. van Delft, B. Bellot-Gurlet, T. Mordan, and A. Alahi, "Ttt++: When does self-supervised test-time training fail or thrive?" in *Proc. NeurIPS*, 2021, pp. 21 808–21 820.
- [273] B. Bozorgtabar, D. Mahapatra, and J.-P. Thiran, "Anomaly detection and localization using attention-guided synthetic anomaly and test-time adaptation," in *Proc. BMVC*, 2022.
- [274] S. Yang, Y. Wang, J. van de Weijer, L. Herranz, and S. Jui, "Casting a bait for offline and online source-free domain adaptation," *arXiv preprint arXiv:2010.12427*, 2021.
- [275] Q. Tian, S. Peng, H. Sun, J. Zhou, and H. Zhang, "Source-free unsupervised domain adaptation with maintaining model balance and diversity," *Computers and Electrical Engineering*, vol. 104, p. 108408, 2022.
- [276] H. Xia, H. Zhao, and Z. Ding, "Adaptive adversarial network for source-free domain adaptation," in *Proc. ICCV*, 2021, pp. 9010–9019.
- [277] O. Chapelle, J. Weston, and B. Schölkopf, "Cluster kernels for semi-supervised learning," in *Proc. NeurIPS*, 2002, pp. 585–592.
- [278] O. Chapelle and A. Zien, "Semi-supervised classification by low density separation," in *Proc. AISTATS*, 2005, pp. 57–64.
- [279] Y. Ye, Z. Liu, Y. Zhang, J. Li, and H. Shen, "Alleviating style sensitivity then adapting: Source-free domain adaptation for medical image segmentation," in *Proc. ACM-MM*, 2022, pp. 1935–1944.
- [280] C. Tsallis, "Possible generalization of boltzmann-gibbs statistics," *Journal of Statistical Physics*, vol. 52, no. 1, pp. 479–487, 1988.
- [281] M. Bateson, H. Kervadec, J. Dolz, H. Lombaert, and I. Ben Ayed, "Source-relaxed domain adaptation for image segmentation," in *Proc. MICCAI*, 2020, pp. 490–499.
- [282] M. Bateson, H. Kervadec, J. Dolz, H. Lombaert, and I. B. Ayed, "Source-free domain adaptation for image segmentation," *Medical Image Analysis*, vol. 82, p. 102617, 2022.

- [283] S. Kondo, "Source-free unsupervised domain adaptation with norm and shape constraints for medical image segmentation," *arXiv preprint arXiv:2209.01300*, 2022.
- [284] Y. Liu, W. Zhang, and J. Wang, "Source-free domain adaptation for semantic segmentation," in *Proc. CVPR*, 2021, pp. 1215–1224.
- [285] M. Plananamente, C. Plizzari, and B. Caputo, "Test-time adaptation for egocentric action recognition," in *Proc. ICIAP*, 2022, p. 206–218.
- [286] J. S. Bridle, A. J. Heading, and D. J. MacKay, "Unsupervised classifiers, mutual information and 'phantom targets'," in *Proc. NeurIPS*, 1991, pp. 1096–1101.
- [287] Y. Shi and F. Sha, "Information-theoretical learning of discriminative clusters for unsupervised domain adaptation," in *Proc. ICML*, 2012, pp. 1275–1282.
- [288] H. Yan, Y. Guo, and C. Yang, "Source-free unsupervised domain adaptation with surrogate data generation," in *Proc. BMVC*, 2021.
- [289] Z. Zhang and Z. Zhang, "Source-free implicit semantic augmentation for domain adaptation," in *Proc. PRICAI*, 2022, pp. 17–31.
- [290] Y. Dai, T. Zheng, C. Xue, and L. Zhou, "Mvit-pcd: A lightweight vit-based network for martian surface topographic change detection," *IEEE Geoscience and Remote Sensing Letters*, 2023.
- [291] Y. Xu, J. Yang, H. Cao, K. Wu, W. Min, and Z. Chen, "Learning temporal consistency for source-free video domain adaptation," in *Proc. ECCV*, 2022, pp. 147–164.
- [292] Y. Huang, X. Yang, J. Zhang, and C. Xu, "Relative alignment network for source-free multimodal video domain adaptation," in *Proc. ACM-MM*, 2022, pp. 1652–1660.
- [293] Y. Xu, J. Yang, M. Wu, X. Li, L. Xie, and Z. Chen, "Extern: Leveraging endo-temporal regularization for black-box video domain adaptation," *arXiv preprint arXiv:2208.05187*, 2022.
- [294] Q. An, R. Li, L. Gu, H. Zhang, Q. Chen, Z. Lu, F. Wang, and Y. Zhu, "A privacy-preserving unsupervised domain adaptation framework for clinical text analysis," *arXiv preprint arXiv:2201.07317*, 2022.
- [295] J. Liu, X. Li, S. An, and Z. Chen, "Source-free unsupervised domain adaptation for blind image quality assessment," *arXiv preprint arXiv:2207.08124*, 2022.
- [296] S. Roy, M. Trapp, A. Pilzer, J. Kannala, N. Sebe, E. Ricci, and A. Solin, "Uncertainty-guided source-free domain adaptation," in *Proc. ECCV*, 2022, pp. 537–555.
- [297] M. Jing, X. Zhen, J. Li, and C. G. M. Snoek, "Variational model perturbation for source-free domain adaptation," in *Proc. NeurIPS*, 2022.
- [298] A. Krause, P. Perona, and R. Gomes, "Discriminative clustering by regularized information maximization," in *Proc. NeurIPS*, 2010, pp. 775–783.
- [299] W. Hu, T. Miyato, S. Tokui, E. Matsumoto, and M. Sugiyama, "Learning discrete representations via information maximizing self-augmented training," in *Proc. ICML*, 2017, pp. 1558–1567.
- [300] K. Saito, D. Kim, S. Sclaroff, and K. Saenko, "Universal domain adaptation through self supervision," in *Proc. NeurIPS*, 2020, pp. 16282–16292.
- [301] S. Yang, Y. Wang, K. Wang, S. Jui, and J. van de Weijer, "Attracting and dispersing: A simple approach for source-free domain adaptation," in *Proc. NeurIPS*, 2022.
- [302] Y. Chen, X. Zhu, Y. Li, Y. Li, Y. Wei, and H. Fang, "Contrast and clustering: Learning neighborhood pair representation for source-free domain adaptation," *arXiv preprint arXiv:2301.13428*, 2023.
- [303] M. Zhu, "Source free domain adaptation by deep embedding clustering," in *International Computer Conference on Wavelet Active Media Technology and Information Processing*, 2021, pp. 309–312.
- [304] J. Xie, R. Girshick, and A. Farhadi, "Unsupervised deep embedding for clustering analysis," in *Proc. ICML*, 2016, pp. 478–487.
- [305] P. Morerio, R. Volpi, R. Ragonesi, and V. Murino, "Generative pseudo-label refinement for unsupervised domain adaptation," in *Proc. WACV*, 2020, pp. 3130–3139.
- [306] V. K. Kurmi, V. K. Subramanian, and V. P. Namboodiri, "Domain impression: A source data free domain adaptation method," in *Proc. WACV*, 2021, pp. 615–625.
- [307] Q. Zhou, K.-Y. Zhang, T. Yao, R. Yi, K. Sheng, S. Ding, and L. Ma, "Generative domain adaptation for face anti-spoofing," in *Proc. ECCV*, 2022, pp. 335–356.
- [308] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784*, 2014.
- [309] Q. Xu, Y. Shi, X. Yuan, and X. X. Zhu, "Universal domain adaptation for remote sensing image scene classification," *IEEE Transactions on Geoscience and Remote Sensing*, 2023.
- [310] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proc. CVPR*, 2016, pp. 2414–2423.
- [311] B. Yang, A. J. Ma, and P. C. Yuen, "Revealing task-relevant model memorization for source-protected unsupervised domain adaptation," *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 716–731, 2022.
- [312] Y. Li, N. Wang, J. Liu, and X. Hou, "Demystifying neural style transfer," in *Proc. IJCAI*, 2017, pp. 2230–2236.
- [313] R. Sahoo, D. Shanmugam, and J. Guttag, "Unsupervised domain adaptation in the absence of source data," in *Proc. ICML Workshops*, 2020.
- [314] H. Bahng, A. Jahanian, S. Sankaranarayanan, and P. Isola, "Visual prompting: Modifying pixel space to adapt pre-trained models," *arXiv preprint arXiv:2203.17274*, 2022.
- [315] J. Hong, Y.-D. Zhang, and W. Chen, "Source-free unsupervised domain adaptation for cross-modality abdominal multi-organ segmentation," *Knowledge-Based Systems*, vol. 250, p. 109155, 2022.
- [316] J. Yang, X. Peng, K. Wang, Z. Zhu, J. Feng, L. Xie, and Y. You, "Divide to adapt: Mitigating confirmation bias for domain adaptation of black-box predictors," in *Proc. ICLR*, 2023.
- [317] C. Liu, L. Zhou, M. Ye, and X. Li, "Self-alignment for black-box domain adaptation of image classification," *IEEE Signal Processing Letters*, vol. 29, pp. 1709–1713, 2022.
- [318] W. Zhang and D. Wu, "Lightweight source-free transfer for privacy-preserving motor imagery classification," *IEEE Transactions on Cognitive and Developmental Systems*, 2022.
- [319] M. Shen, Y. Bu, and G. Wornell, "On the benefits of selectivity in pseudo-labeling for unsupervised multi-source-free domain adaptation," *arXiv preprint arXiv:2202.00796*, 2022.
- [320] Q. Tian, C. Ma, F.-Y. Zhang, S. Peng, and H. Xue, "Source-free unsupervised domain adaptation with sample transport learning," *Journal of Computer Science and Technology*, vol. 36, no. 3, pp. 606–616, 2021.
- [321] N. Ma, J. Bu, Z. Zhang, and S. Zhou, "Uncertainty-guided mixup for semi-supervised domain adaptation without source data," *arXiv preprint arXiv:2107.06707*, 2021.
- [322] T. Chu, Y. Liu, J. Deng, W. Li, and L. Duan, "Denoised maximum classifier discrepancy for source free unsupervised domain adaptation," in *Proc. AAAI*, 2022, pp. 472–480.
- [323] N. Ding, Y. Xu, Y. Tang, C. Xu, Y. Wang, and D. Tao, "Source-free domain adaptation via distribution estimation," in *Proc. CVPR*, 2022, pp. 7212–7222.
- [324] S. Stan and M. Rostami, "Unsupervised model adaptation for continual semantic segmentation," in *Proc. AAAI*, 2021, pp. 2593–2601.
- [325] M. Rostami, "Lifelong domain adaptation via consolidated internal distribution," in *Proc. NeurIPS*, 2021, pp. 11172–11183.
- [326] S. Stan and M. Rostami, "Domain adaptation for the segmentation of confidential medical images," in *Proc. BMVC*, 2022.
- [327] W.-Y. Chen, Y.-C. Liu, Z. Kira, Y.-C. F. Wang, and J.-B. Huang, "A closer look at few-shot classification," in *Proc. ICLR*, 2018.
- [328] Z. Xu, W. Wei, L. Zhang, and J. Nie, "Source-free domain adaptation for cross-scene hyperspectral image classification," in *Proc. IGARSS*, 2022, pp. 3576–3579.
- [329] K. Saito, K. Watanabe, Y. Ushiku, and T. Harada, "Maximum classifier discrepancy for unsupervised domain adaptation," in *Proc. CVPR*, 2018, pp. 3723–3732.
- [330] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, "A kernel two-sample test," *Journal of Machine Learning Research*, vol. 13, no. 1, pp. 723–773, 2012.
- [331] S. Chhabra, H. Venkateswara, and B. Li, "Generative alignment of posterior probabilities for source-free domain adaptation," in *Proc. WACV*, 2023, pp. 4125–4134.
- [332] M. Shen, J. Ma, and P. chi Yuen, "E²: Entropy discrimination and energy optimization for source-free universal domain adaptation," 2023.
- [333] Y. Zhang, Z. Ren, K. Feng, K. Yu, M. Beer, and Z. Liu, "Universal source-free domain adaptation method for cross-domain fault diagnosis of machines," *Mechanical Systems and Signal Processing*, vol. 191, p. 110159, 2023.
- [334] A. Maracani, R. Camoriano, E. Maiettini, D. Talon, L. Rosasco, and L. Natale, "Key design choices for double-transfer in

- source-free unsupervised domain adaptation," *arXiv preprint arXiv:2302.05379*, 2023.
- [335] Y. Li, N. Wang, J. Shi, J. Liu, and X. Hou, "Revisiting batch normalization for practical domain adaptation," in *Proc. ICLR*, 2017.
- [336] M. Klingner, J.-A. Termöhlen, J. Ritterbach, and T. Fingscheidt, "Unsupervised batchnorm adaptation (ubna): A domain adaptation method for semantic segmentation without using source domain representations," in *Proc. WACV Workshops*, 2022.
- [337] M. Yazdanpanah and P. Moradi, "Visual domain bridge: A source-free domain adaptation for cross-domain few-shot learning," in *Proc. CVPR Workshops*, 2022.
- [338] N. Chattopadhyay, S. Gehlot, and N. Singhal, "Fusion: Fully unsupervised test-time stain adaptation via fused normalization statistics," in *Proc. ECCV Workshops*, 2022.
- [339] J. N. Kundu, A. Kulkarni, S. Bhambri, D. Mehta, S. Kulkarni, V. Jampani, and R. V. Babu, "Balancing discriminability and transferability for source-free domain adaptation," in *Proc. ICML*, 2022, pp. 11 710–11 728.
- [340] C. Eastwood, I. Mason, C. K. Williams, and B. Schölkopf, "Source-free adaptation to measurement shift via bottom-up feature restoration," in *Proc. ICLR*, 2022.
- [341] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. ICCV*, 2017, pp. 2980–2988.
- [342] K. Tanwisuth, X. Fan, H. Zheng, S. Zhang, H. Zhang, B. Chen, and M. Zhou, "A prototype-oriented framework for unsupervised domain adaptation," in *Proc. NeurIPS*, 2021, pp. 17 194–17 208.
- [343] B. Yang, H.-W. Yeh, T. Harada, and P. C. Yuen, "Model-induced generalization error bound for information-theoretic representation learning in source-data-free unsupervised domain adaptation," *IEEE Transactions on Image Processing*, vol. 31, pp. 419–432, 2021.
- [344] H.-W. Yeh, B. Yang, P. C. Yuen, and T. Harada, "Sofa: Source-data-free feature alignment for unsupervised domain adaptation," in *Proc. WACV*, 2021, pp. 474–483.
- [345] J. Wang, J. Zhang, Y. Bian, Y. Cai, C. Wang, and S. Pu, "Self-domain adaptation for face anti-spoofing," in *Proc. AAAI*, 2021, pp. 2746–2754.
- [346] O. Bohdal, D. Li, S. X. Hu, and T. Hospedales, "Feed-forward source-free latent domain adaptation via cross-attention," in *Proc. ICML Workshops*, 2022.
- [347] S. Li, H. Wu, L. Ding, and D. Wu, "Meta-learning for fast and privacy-preserving source knowledge transfer of eeg-based bcis," *IEEE Computational Intelligence Magazine*, vol. 17, no. 4, pp. 16–26, 2022.
- [348] A. Naik, Y. Wu, M. Naik, and E. Wong, "Do machine learning models learn common sense?" *arXiv preprint arXiv:2303.01433*, 2023.
- [349] J. Liang, Y. Wang, D. Hu, R. He, and J. Feng, "A balanced and uncertainty-aware approach for partial domain adaptation," in *Proc. ECCV*, 2020, pp. 123–140.
- [350] J. N. Kundu, N. Venkat, A. Revanur, and R. V. Babu, "Towards inheritable models for open-set domain adaptation," in *Proc. CVPR*, 2020, pp. 12 376–12 385.
- [351] K. You, M. Long, Z. Cao, J. Wang, and M. I. Jordan, "Universal domain adaptation," in *Proc. CVPR*, 2019, pp. 2720–2729.
- [352] S. Yang, Y. Wang, K. Wang, S. Jui, and J. van de Weijer, "One ring to bring them all: Model adaptation under domain and category shift," *arXiv preprint arXiv:2206.03600*, 2022.
- [353] S. Qu, T. Zou, F. Roehrbein, C. Lu, G. Chen, D. Tao, and C. Jiang, "Upcycling models under domain and category shift," in *Proc. CVPR*, 2023.
- [354] H. Wan, L. Chen, and M. Deng, "scemail: Universal and source-free annotation method for scrna-seq data with novel cell-type perception," *Genomics, Proteomics & Bioinformatics*, 2023.
- [355] S. M. Ahmed, D. S. Raychaudhuri, S. Paul, S. Oymak, and A. K. Roy-Chowdhury, "Unsupervised multi-source domain adaptation without access to source data," in *Proc. CVPR*, 2021, pp. 10 103–10 112.
- [356] N. Ma, H. Wang, Z. Zhang, S. Zhou, H. Chen, and J. Bu, "Source-free semi-supervised domain adaptation via progressive mixup," *Knowledge-Based Systems*, p. 110208, 2022.
- [357] N. Ma, J. Bu, L. Lu, J. Wen, S. Zhou, Z. Zhang, J. Gu, H. Li, and X. Yan, "Context-guided entropy minimization for semi-supervised domain adaptation," *Neural Networks*, vol. 154, pp. 270–282, 2022.
- [358] F. Wang, Z. Han, Z. Zhang, and Y. Yin, "Active source free domain adaptation," *arXiv preprint arXiv:2205.10711*, 2022.
- [359] X. Su, Y. Zhao, and S. Bethard, "The university of arizona at semeval-2021 task 10: Applying self-training, active learning and data augmentation to source-free domain adaptation," in *International Workshop on Semantic Evaluation (SemEval)*, 2021, pp. 458–466.
- [360] X. Li, Z. Du, J. Li, L. Zhu, and K. Lu, "Source-free active domain adaptation via energy-based locality preserving transfer," in *Proc. ACM-MM*, 2022, pp. 5802–5810.
- [361] W. Sun, Q. Chen, C. Wang, D. Ye, and C. Chen, "Source-free unsupervised domain adaptation in imbalanced datasets," in *International Conference on Data Science and Information Technology (DSIT)*, 2022, pp. 1–6.
- [362] P. Agarwal, D. P. Paudel, J.-N. Zaech, and L. Van Gool, "Unsupervised robust domain adaptation without source data," in *Proc. WACV*, 2022, pp. 2009–2018.
- [363] L. Sheng, J. Liang, R. He, Z. Wang, and T. Tan, "Adaptguard: Defending against universal attacks for model adaptation," *arXiv preprint arXiv:2303.10594*, 2023.
- [364] K. Tanwisuth, S. Zhang, P. He, and M. Zhou, "A prototype-oriented clustering for domain shift with source privacy," *arXiv preprint arXiv:2302.03807*, 2023.
- [365] A. D'Innocente, S. Bucci, B. Caputo, and T. Tommasi, "Learning to generalize one sample at a time with self-supervision," *arXiv preprint arXiv:1910.03915*, 2019.
- [366] S. Wegmann, F. Scattoni, I. Carp, L. Gillick, R. Roth, and J. Yamron, "Dragon systems' 1997 broadcast news transcription system," in *Proc. DARPA Broadcast News Transcription and Understanding Workshop*, 1998.
- [367] S. Brahmabhatt, J. Gu, K. Kim, J. Hays, and J. Kautz, "Geometry-aware learning of maps for camera localization," in *Proc. CVPR*, 2018, pp. 2616–2625.
- [368] F. Azimi, S. Palacio, F. Raue, J. Hees, L. Bertinetto, and A. Dengel, "Self-supervised test-time adaptation on video data," in *Proc. WACV*, 2022, pp. 3439–3448.
- [369] R. Wang, Y. Sun, Y. Gandelsman, X. Chen, A. A. Efros, and X. Wang, "Test-time training on video streams," 2023.
- [370] Z. Nado, S. Padhy, D. Sculley, A. D'Amour, B. Lakshminarayanan, and J. Snoek, "Evaluating prediction-time batch normalization for robustness under covariate shift," in *Proc. ICML Workshops*, 2020.
- [371] Y. Zou, Z. Zhang, C.-L. Li, H. Zhang, T. Pfister, and J.-B. Huang, "Learning instance-specific adaptation for cross-domain segmentation," in *Proc. ECCV*, 2022, pp. 459–476.
- [372] S. Park, J. Yoo, D. Cho, J. Kim, and T. H. Kim, "Fast adaptation to super-resolution networks via meta-learning," in *Proc. ECCV*, 2020, pp. 754–769.
- [373] F. C. Borlino, S. Polizzotto, B. Caputo, and T. Tommasi, "Self-supervision & meta-learning for one-shot unsupervised cross-domain detection," *Computer Vision and Image Understanding*, vol. 223, p. 103549, 2022.
- [374] M. Shu, W. Nie, D.-A. Huang, Z. Yu, T. Goldstein, A. Anandkumar, and C. Xiao, "Test-time prompt tuning for zero-shot generalization in vision-language models," in *Proc. NeurIPS*, 2022.
- [375] N. Karani, E. Erdil, K. Chaitanya, and E. Konukoglu, "Test-time adaptable neural networks for robust medical image segmentation," *Medical Image Analysis*, vol. 68, p. 101907, 2021.
- [376] X. Wang, Y. Tsvetkov, S. Ruder, and G. Neubig, "Efficient test time adapter ensembling for low-resource language varieties," in *EMNLP Findings*, 2021, pp. 730–737.
- [377] Y. Zhang, J. Wang, J. Liang, Z. Zhang, B. Yu, L. Wang, X. Xie, and D. Tao, "Domain-specific risk minimization for out-of-distribution generalization," *arXiv preprint arXiv:2208.08661*, 2023.
- [378] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. ICML*, 2015, pp. 448–456.
- [379] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," in *Proc. NeurIPS Workshops*, 2016.
- [380] C. Burns and J. Steinhardt, "Limitations of post-hoc feature alignment for robustness," in *Proc. CVPR*, 2021, pp. 2525–2533.
- [381] E. Rosenfeld, P. Ravikumar, and A. Risteski, "Domain-adjusted regression or: Erm may already learn features sufficient for out-of-distribution generalization," *arXiv preprint arXiv:2202.06856*, 2022.

- [382] T. Duboudin, E. Dellandréa, C. Abgrall, G. Hénaff, and L. Chen, "Learning less generalizable patterns with an asymmetrically trained double classifier for better test-time adaptation," *arXiv preprint arXiv:2210.09834*, 2022.
- [383] P. Benz, C. Zhang, A. Karjauv, and I. S. Kweon, "Revisiting batch normalization for improving corruption robustness," in *Proc. WACV*, 2021, pp. 494–503.
- [384] J. Nandy, S. Saha, W. Hsu, L. Mong, and X. X. Zhu, "Covariate shift adaptation for adversarially robust classifier," in *Proc. ICLR Workshops*, 2021.
- [385] H. Lim, B. Kim, J. Choo, and S. Choi, "Ttn: A domain-shift aware batch normalization in test-time adaptation," in *Proc. ICLR*, 2023.
- [386] M. Klingner, M. Ayache, and T. Fingscheidt, "Continual batch-norm adaptation (cbna) for semantic segmentation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 11, pp. 20 899–20 911, 2022.
- [387] S. Bahmani, O. Hahn, E. Zamfir, N. Araslanov, D. Cremers, and S. Roth, "Semantic self-adaptation: Enhancing generalization with a single sample," in *Proc. ECCV Workshops*, 2022.
- [388] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Instance normalization: The missing ingredient for fast stylization," *arXiv preprint arXiv:1607.08022*, 2016.
- [389] A. Khurana, S. Paul, P. Rai, S. Biswas, and G. Aggarwal, "Sita: Single image test-time adaptation," *arXiv preprint arXiv:2112.02355*, 2021.
- [390] A. D'Innocente, F. C. Borlino, S. Bucci, B. Caputo, and T. Tommasi, "One-shot unsupervised cross-domain detection," in *Proc. ECCV*, 2020, pp. 732–748.
- [391] D. Wang, E. Shelhamer, B. Olshausen, and T. Darrell, "Dynamic scale inference by entropy minimization," *arXiv preprint arXiv:1908.03182*, 2019.
- [392] N. Reddy, A. Singhal, A. Kumar, M. Baktashmotlagh, and C. Arora, "Master of all: Simultaneous generalization of urban-scene segmentation to all adverse weather conditions," in *Proc. ECCV*, 2022, pp. 51–69.
- [393] Y. Sun, E. Tzeng, T. Darrell, and A. A. Efros, "Unsupervised domain adaptation through self-supervision," *arXiv preprint arXiv:1909.11825*, 2019.
- [394] J. Zhang, X. Nie, and J. Feng, "Inference stage optimization for cross-scenario 3d human pose estimation," in *Proc. NeurIPS*, 2020, pp. 2408–2419.
- [395] Y. Li, M. Hao, Z. Di, N. B. Gundavarapu, and X. Wang, "Test-time personalization with a transformer for human pose estimation," in *Proc. NeurIPS*, 2021, pp. 2583–2597.
- [396] N. Hansen, R. Jangir, Y. Sun, G. Alenyà, P. Abbeel, A. A. Efros, L. Pinto, and X. Wang, "Self-supervised policy adaptation during deployment," in *Proc. ICLR*, 2021.
- [397] M. J. Mirza, I. Shin, W. Lin, A. Schriebl, K. Sun, J. Choe, H. Possegger, M. Kozinski, I. S. Kweon, K.-J. Yoon, and H. Bischof, "Mate: Masked autoencoders are online 3d test-time learners," *arXiv preprint arXiv:2211.11432*, 2022.
- [398] M. Prabhudesai, S. Paul, S. van Steenkiste, M. S. Sajjadi, A. Goyal, D. Pathak, K. Fragkiadaki, G. Aggarwal, and T. Kipf, "Test-time adaptation with slot-centric models," in *Proc. NeurIPS Workshops*, 2022.
- [399] Y. Gandelsman, Y. Sun, X. Chen, and A. A. Efros, "Test-time training with masked autoencoders," in *Proc. NeurIPS*, 2022.
- [400] D. Osowiecki, G. A. V. Hakim, M. Noori, M. Cheraghali, I. Ben Ayed, and C. Desrosiers, "Tttflow: Unsupervised test-time training with normalizing flow," in *Proc. WACV*, 2023, pp. 2126–2134.
- [401] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *Proc. CVPR*, 2022, pp. 16 000–16 009.
- [402] T. Cohen, N. Shulman, H. Morgenstern, R. Mechrez, and E. Farhan, "Self-supervised dynamic networks for covariate shift robustness," *arXiv preprint arXiv:2006.03952*, 2020.
- [403] F. Lyu, M. Ye, A. J. Ma, T. C.-F. Yip, G. L.-H. Wong, and P. C. Yuen, "Learning from synthetic ct images via test-time training for liver tumor segmentation," *IEEE Transactions on Medical Imaging*, vol. 41, no. 9, pp. 2510–2520, 2022.
- [404] A. Sarkar, A. Sarkar, and V. N. Balasubramanian, "Leveraging test-time consensus prediction for robustness against unseen noise," in *Proc. WACV*, 2022, pp. 1839–1848.
- [405] A. Bartler, F. Bender, F. Wiewel, and B. Yang, "Ttaps: Test-time adaption by aligning prototypes using self-supervision," in *Proc. IJCNN*, 2022, pp. 1–8.
- [406] M. Bateson, H. Lombaert, and I. B. Ayed, "Test-time adaptation with shape moments for image segmentation," in *Proc. MICCAI*, 2022, pp. 736–745.
- [407] G.-T. Lin, S.-W. Li, and H.-y. Lee, "Listen, adapt, better wer: Source-free single-utterance test-time adaptation for automatic speech recognition," in *Proc. Interspeech*, 2022.
- [408] N. Karani, G. Brunner, E. Erdil, S. Fei, K. Tezcan, K. Chaitanya, and E. Konukoglu, "A field of experts prior for adapting neural networks at test time," *arXiv preprint arXiv:2202.05271*, 2022.
- [409] S. Roth and M. J. Black, "Fields of experts: A framework for learning image priors," in *Proc. CVPR*, 2005, pp. 860–867.
- [410] X. Wang and L. Aitchison, "Robustness to corruption in pre-trained bayesian neural networks," in *Proc. ICLR*, 2023.
- [411] Q. Liu, C. Chen, Q. Dou, and P.-A. Heng, "Single-domain generalization in medical image segmentation via test-time adaptation from shape dictionary," in *Proc. AAAI*, 2022, pp. 1756–1764.
- [412] W. Jin, T. Zhao, J. Ding, Y. Liu, J. Tang, and N. Shah, "Empowering graph representation learning with test-time graph transformation," in *Proc. ICLR*, 2023.
- [413] H.-Y. F. Tung, H.-W. Tung, E. Yumer, and K. Fragkiadaki, "Self-supervised learning of motion capture," in *Proc. NeurIPS*, 2017, pp. 5242–5252.
- [414] B. Leung, C.-H. Ho, and N. Vasconcelos, "Black-box test-time shape refinement for single view 3d reconstruction," in *Proc. CVPR Workshops*, 2022, pp. 4080–4090.
- [415] Z. Kan, S. Chen, Z. Li, and Z. He, "Self-constrained inference optimization on structural groups for human pose estimation," in *Proc. ECCV*, 2022, pp. 729–745.
- [416] V. Casser, S. Pirk, R. Mahjourian, and A. Angelova, "Depth prediction without the sensors: Leveraging structure for unsupervised learning from monocular videos," in *Proc. AAAI*, 2019, pp. 8001–8008.
- [417] Y. Chen, C. Schmid, and C. Sminchisescu, "Self-supervised learning with geometric constraints in monocular video: Connecting flow, depth, and camera," in *Proc. ICCV*, 2019, pp. 7063–7072.
- [418] X. Li, S. Liu, S. De Mello, K. Kim, X. Wang, M.-H. Yang, and J. Kautz, "Online adaptation for consistent mesh reconstruction in the wild," in *Proc. NeurIPS*, 2020, pp. 15 009–15 019.
- [419] X. Luo, J.-B. Huang, R. Szeliski, K. Matzen, and J. Kopf, "Consistent video depth estimation," *ACM Transactions on Graphics*, vol. 39, no. 4, pp. 71:1–71:13, 2020.
- [420] O. C. Mutlu, M. Honarmand, S. Surabhi, and D. P. Wall, "Temp: Temporal consistency for test-time adaptation," *arXiv preprint arXiv:2303.10536*, 2023.
- [421] W. Zhu, Y. Huang, D. Xu, Z. Qian, W. Fan, and X. Xie, "Test-time training for deformable multi-scale image registration," in *Proc. ICRA*, 2021, pp. 13 618–13 625.
- [422] S. Hong and S. Kim, "Deep matching prior: Test-time optimization for dense correspondence," in *Proc. ICCV*, 2021, pp. 9907–9917.
- [423] G. Vaksman, M. Elad, and P. Milanfar, "Lidia: Lightweight learned image denoising with instance adaptation," in *Proc. CVPR Workshops*, 2020.
- [424] S. Mohan, J. L. Vincent, R. Manzorro, P. Crozier, C. Fernandez-Granda, and E. Simoncelli, "Adaptive denoising via gaintuning," in *Proc. NeurIPS*, 2021, pp. 23 727–23 740.
- [425] D. Bau, H. Strobel, W. Peebles, J. Wulff, B. Zhou, J.-Y. Zhu, and A. Torralba, "Semantic photo manipulation with a generative image prior," *ACM Transactions on Graphics*, vol. 38, no. 4, pp. 1–11, 2019.
- [426] Y. Nitzan, K. Aberman, Q. He, O. Liba, M. Yarom, Y. Gandelsman, I. Mosseri, Y. Pritch, and D. Cohen-Or, "Mystyle: A personalized generative prior," *ACM Transactions on Graphics*, vol. 41, no. 6, pp. 1–10, 2022.
- [427] S. Kim, Y. Min, Y. Jung, and S. Kim, "Controllable style transfer via test-time training of implicit neural representation," *arXiv preprint arXiv:2210.07762*, 2022.
- [428] Y. Ding, J. Liang, J. Cao, A. Zheng, and R. He, "Modify: Model-driven face stylization without style images," in *Proc. ICASSP*, 2023.
- [429] D. Gilton, G. Ongie, and R. Willett, "Model adaptation for inverse problems in imaging," *IEEE Transactions on Computational Imaging*, vol. 7, pp. 661–674, 2021.
- [430] P. Banerjee, T. Gokhale, and C. Baral, "Self-supervised test-time learning for reading comprehension," in *Proc. NAACL*, 2021, pp. 1200–1211.

- [431] Y. Özer and M. Müller, "Source separation of piano concertos with test-time adaptation," in *Proc. ISMIR*, 2022.
- [432] A. Shocher, N. Cohen, and M. Irani, "zero-shot" super-resolution using deep internal learning," in *Proc. CVPR*, 2018, pp. 3118–3126.
- [433] S. Paul, A. Saha, and A. Samanta, "Ttt-ucdr: Test-time training for universal cross-domain retrieval," *arXiv preprint arXiv:2208.09198*, 2022.
- [434] A. Dubey, V. Ramanathan, A. Pentland, and D. Mahajan, "Adaptive methods for real-world domain generalization," in *Proc. CVPR*, 2021, pp. 14 340–14 349.
- [435] J. Kim, J.-T. Lee, S. Chang, and N. Kwak, "Variational on-the-fly personalization," in *Proc. ICML*, 2022, pp. 11 134–11 147.
- [436] J. W. Soh, S. Cho, and N. I. Cho, "Meta-transfer learning for zero-shot super-resolution," in *Proc. CVPR*, 2020, pp. 3516–3525.
- [437] Z. Chi, Y. Wang, Y. Yu, and J. Tang, "Test-time fast adaptation for dynamic scene deblurring via meta-auxiliary learning," in *Proc. CVPR*, 2021, pp. 9137–9146.
- [438] H. Liu, Z. Wu, L. Li, S. Salehkalibar, J. Chen, and K. Wang, "Towards multi-domain single image dehazing via test-time training," in *Proc. CVPR*, 2022, pp. 5831–5840.
- [439] C. Min, T. Kim, and J. Lim, "Meta-learning for adaptation of deep optical flow networks," in *Proc. WACV*, 2023, pp. 2145–2154.
- [440] M. Choi, J. Choi, S. Baik, T. H. Kim, and K. M. Lee, "Test-time adaptation for video frame interpolation via meta-learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 12, pp. 9615–9628, 2021.
- [441] L. Chen, Y. Zhang, Y. Song, J. Wang, and L. Liu, "Ost: Improving generalization of deepfake detection via one-shot test-time training," in *Proc. NeurIPS*, 2023.
- [442] F. Alet, M. Bauza, K. Kawaguchi, N. G. Kuru, T. Lozano-Perez, and L. P. Kaelbling, "Tailoring: Encoding inductive biases by optimizing unsupervised objectives at prediction time," in *Proc. NeurIPS*, 2021, pp. 29 206–29 217.
- [443] A. Bartler, A. Bühler, F. Wiewel, M. Döbler, and B. Yang, "Mt3: Meta test-time training for self-supervised test-time adaption," in *Proc. AISTATS*, 2022, pp. 3080–3090.
- [444] Z. Huang, C. Liu, M. Salzmann, S. Süsstrunk, and T. Zhang, "Test-time adaptation for better adversarial robustness," 2023.
- [445] A. Sain, A. K. Bhunia, V. Potlapalli, P. N. Chowdhury, T. Xiang, and Y.-Z. Song, "Sketch3t: Test-time training for zero-shot sbir," in *Proc. CVPR*, 2022, pp. 7462–7471.
- [446] H. Liu, Z. Chi, Y. Yu, Y. Wang, J. Chen, and J. Tang, "Meta-auxiliary learning for future depth prediction in videos," in *Proc. WACV*, 2023, pp. 5756–5765.
- [447] M. Zhang, H. Marklund, N. Dhawan, A. Gupta, S. Levine, and C. Finn, "Adaptive risk minimization: Learning to adapt to domain shift," in *Proc. NeurIPS*, 2021, pp. 23 664–23 678.
- [448] Y. Jiang, Y. Wang, R. Zhang, Q. Xu, Y. Zhang, X. Chen, and Q. Tian, "Domain-conditioned normalization for test-time domain generalization," in *Proc. ECCV Workshops*, 2022, pp. 291–307.
- [449] Y. Li, R. Dan, S. Wang, Y. Cao, X. Luo, C. Tan, G. Jia, H. Zhou, Y. Wang, and L. Wang, "Plug-and-play shape refinement framework for multi-site and lifespan brain skull stripping," in *Proc. MICCAI Workshops*, 2022.
- [450] Z. Sun, Z. Shen, L. Lin, Y. Yu, Z. Yang, S. Yang, and W. Chen, "Dynamic domain generalization," in *Proc. IJCAI*, 2022, pp. 1342–1348.
- [451] S. Hu, Z. Liao, J. Zhang, and Y. Xia, "Domain and content adaptive convolution based multi-source domain generalization in medical image segmentation," *IEEE Transactions on Medical Imaging*, vol. 42, no. 1, pp. 233–244, 2022.
- [452] Z. Zheng, X. Yue, K. Wang, and Y. You, "Prompt vision transformer for domain generalization," *arXiv preprint arXiv:2208.08914*, 2022.
- [453] E. Ben-David, N. Oved, and R. Reichart, "Pada: Example-based prompt learning for on-the-fly adaptation to unseen domains," *Transactions of the Association for Computational Linguistics*, vol. 10, pp. 414–433, 2022.
- [454] S. Seo, Y. Suh, D. Kim, G. Kim, J. Han, and B. Han, "Learning to optimize domain specific normalization for domain generalization," in *Proc. ECCV*, 2020, pp. 68–83.
- [455] Z. Xiao, X. Zhen, L. Shao, and C. G. Snoek, "Learning to generalize across domains on single test samples," in *Proc. ICLR*, 2022.
- [456] T. Zhong, Z. Chi, L. Gu, Y. Wang, Y. Yu, and J. Tang, "Meta-dmoe: Adapting to domain shift by meta-distillation from mixture-of-experts," in *Proc. NeurIPS*, 2022.
- [457] B. Xu, J. Liang, L. He, and Z. Sun, "Mimic embedding via adaptive aggregation: Learning generalizable person re-identification," in *Proc. ECCV*, 2022, pp. 372–388.
- [458] M. Segu, A. Tonioni, and F. Tombari, "Batch normalization embeddings for deep domain generalization," *Pattern Recognition*, vol. 135, p. 109115, 2023.
- [459] Y. He, A. Carass, L. Zuo, B. E. Dewey, and J. L. Prince, "Self domain adapted network," in *Proc. MICCAI*, 2020, pp. 437–446.
- [460] X. Zhao, C. Liu, A. Sicilia, S. J. Hwang, and Y. Fu, "Test-time fourier style calibration for domain generalization," in *Proc. IJCAI*, 2022, pp. 1721–1727.
- [461] T. Cordier, V. Bouvier, G. Hénaff, and C. Hudelot, "Test-time adaptation with principal component analysis," in *ECML/PKDD Workshops*, 2022.
- [462] J. Gao, J. Zhang, X. Liu, T. Darrell, E. Shelhamer, and D. Wang, "Back to the source: Diffusion-driven adaptation to test-time corruption," in *Proc. CVPR*, 2023.
- [463] B. Lester, R. Al-Rfou, and N. Constant, "The power of scale for parameter-efficient prompt tuning," *arXiv preprint arXiv:2104.08691*, 2021.
- [464] X. Zhang and Y.-C. Chen, "Adaptive domain generalization via online disagreement minimization," *arXiv preprint arXiv:2208.01996*, 2022.
- [465] Y.-Y. Tsai, C. Mao, Y.-K. Lin, and J. Yang, "Self-supervised convolutional visual prompts," *arXiv preprint arXiv:2303.00198*, 2023.
- [466] Y. He, A. Carass, L. Zuo, B. E. Dewey, and J. L. Prince, "Autoencoder based self-supervised test-time adaptation for medical image analysis," *Medical Image Analysis*, p. 102136, 2021.
- [467] H. Li, H. Liu, D. Hu, J. Wang, H. Johnson, O. Sherbini, F. Gavazzi, R. D'Aiello, A. Vanderver, J. Long *et al.*, "Self-supervised test-time adaptation for medical image segmentation," in *Proc. MICCAI Workshops*, 2022.
- [468] G. Valvano, A. Leo, and S. A. Tsaftaris, "Re-using adversarial mask discriminators for test-time training under distribution shifts," *Journal of Machine Learning for Biomedical Imaging*, vol. 1, pp. 1–27, 2022.
- [469] B. Song, L. Shen, and L. Xing, "Piner: Prior-informed implicit neural representation learning for test-time adaptation in sparse-view ct reconstruction," in *Proc. WACV*, 2023, pp. 1928–1938.
- [470] J.-A. Termöhlen, M. Klingner, L. J. Brettn, N. M. Schmidt, and T. Fingscheidt, "Continual unsupervised domain adaptation for semantic segmentation by online frequency domain style transfer," in *Proc. ITSC*, 2021, pp. 2881–2888.
- [471] J. Park, D.-J. Han, S. Kim, and J. Moon, "Style balancing and test-time style shifting for domain generalization," in *Proc. ICML Workshops*, 2022.
- [472] X. Huang and S. Belongie, "Arbitrary style transfer in real-time with adaptive instance normalization," in *Proc. ICCV*, 2017, pp. 1501–1510.
- [473] P. Pandey, M. Raman, S. Varambally, and P. AP, "Generalization on unseen domains via inference-time label-preserving target projections," in *Proc. CVPR*, 2021, pp. 12 924–12 933.
- [474] Z. Xiao, X. Zhen, S. Liao, and C. G. Snoek, "Energy-based test sample adaptation for domain generalization," in *Proc. ICLR*, 2023.
- [475] C. Shi, C. Holtz, and G. Mishne, "Online adversarial purification based on self-supervision," in *Proc. ICLR*, 2021.
- [476] J. Yoon, S. J. Hwang, and J. Lee, "Adversarial purification with score-based generative models," in *Proc. ICML*, 2021, pp. 12 062–12 072.
- [477] C. Mao, M. Chiquier, H. Wang, J. Yang, and C. Vondrick, "Adversarial attacks are reversible with natural supervision," in *Proc. ICCV*, 2021, pp. 661–671.
- [478] B. Wu, H. Pan, L. Shen, J. Gu, S. Zhao, Z. Li, D. Cai, X. He, and W. Liu, "Attacking adversarial attacks as a defense," *arXiv preprint arXiv:2106.04938*, 2021.
- [479] M. Alfara, J. C. Pérez, A. Thabet, A. Bibi, P. H. Torr, and B. Ghanem, "Combating adversaries with anti-adversaries," in *Proc. AAAI*, 2022, pp. 5992–6000.
- [480] T. Wu, F. Jia, X. Qi, J. T. Wang, V. Sehwag, S. Mahloulifar, and P. Mittal, "Uncovering adversarial risks of test-time adaptation," *arXiv preprint arXiv:2301.12576*, 2023.
- [481] S. Föll, A. Dubatovka, E. Ernst, M. Maritsch, P. Okanovic, G. Thäter, J. M. Buhmann, F. Wortmann, and K. Muandet, "Gated

- domain units for multi-source domain generalization," *arXiv preprint arXiv:2206.12444*, 2022.
- [482] V. Jain and E. Learned-Miller, "Online domain adaptation of a pre-trained cascade of classifiers," in *Proc. CVPR*, 2011, pp. 577–584.
- [483] Q. Wang, O. Fink, L. Van Gool, and D. Dai, "Continual test-time domain adaptation," in *Proc. CVPR*, 2022, pp. 7201–7211.
- [484] S. Niu, J. Wu, Y. Zhang, Y. Chen, S. Zheng, P. Zhao, and M. Tan, "Efficient test-time model adaptation without forgetting," in *Proc. ICML*, 2022, pp. 16 888–16 905.
- [485] M. J. Mirza, J. Micorek, H. Possegger, and H. Bischof, "The norm must go on: Dynamic unsupervised domain adaptation by normalization," in *Proc. CVPR*, 2022, pp. 14 765–14 775.
- [486] B. Zhao, C. Chen, and S.-T. Xia, "Delta: Degradation-free fully test-time adaptation," in *Proc. ICLR*, 2023.
- [487] S. Niu, J. Wu, Y. Zhang, Z. Wen, Y. Chen, P. Zhao, and M. Tan, "Towards stable test-time adaptation in dynamic wild world," in *Proc. ICLR*, 2023.
- [488] M. Jang and S.-Y. Chung, "Test-time adaptation via self-training with nearest neighbor information," in *Proc. ICLR*, 2022.
- [489] T. Kojima, Y. Matsuo, and Y. Iwasawa, "Robustifying vision transformer without retraining from scratch by test-time class-conditional feature alignment," in *Proc. IJCAI*, 2022, pp. 1009–1016.
- [490] M. Boudiaf, R. Mueller, I. Ben Ayed, and L. Bertinetto, "Parameter-free online test-time adaptation," in *Proc. CVPR*, 2022, pp. 8344–8353.
- [491] M. Hu, T. Song, Y. Gu, X. Luo, J. Chen, Y. Chen, Y. Zhang, and S. Zhang, "Fully test-time adaptation for image segmentation," in *Proc. MICCAI*, 2021, pp. 251–260.
- [492] J. Song, K. Park, I. Shin, S. Woo, and I. S. Kweon, "Cd-tta: Compound domain test-time adaptation for semantic segmentation," *arXiv preprint arXiv:2212.08356*, 2022.
- [493] F. You, J. Li, and Z. Zhao, "Test-time batch statistics calibration for covariate shift," *arXiv preprint arXiv:2110.04065*, 2021.
- [494] M. Mancini, H. Karaoguz, E. Ricci, P. Jensfelt, and B. Caputo, "Kitting in the wild through online domain adaptation," in *Proc. IROS*, 2018, pp. 1103–1109.
- [495] T. H. Park and S. D'Amico, "Robust multi-task learning and online refinement for spacecraft pose estimation across domain gap," *arXiv preprint arXiv:2203.04275*, 2022.
- [496] L. Zhang, S. Zhang, X. Yang, and Z. Liu, "Unseen object instance segmentation with fully test-time rgb-d embeddings adaptation," *arXiv preprint arXiv:2204.09847*, 2022.
- [497] J. Hong, L. Lyu, J. Zhou, and M. Spranger, "Mecta: Memory-economic continual test-time model adaptation," in *Proc. ICLR*, 2023.
- [498] T. Yang, S. Zhou, Y. Wang, Y. Lu, and N. Zheng, "Test-time batch normalization," *arXiv preprint arXiv:2205.10210*, 2022.
- [499] S. Ioffe, "Batch renormalization: Towards reducing minibatch dependence in batch-normalized models," in *Proc. NeurIPS*, 2017, pp. 1942–1950.
- [500] X. Hu, G. Uzunbas, S. Chen, R. Wang, A. Shah, R. Nevatia, and S.-N. Lim, "Mixnorm: Test-time adaptation through online normalization estimation," *arXiv preprint arXiv:2110.11478*, 2021.
- [501] T. Gong, J. Jeong, T. Kim, Y. Kim, J. Shin, and S.-J. Lee, "Note: Robust continual test-time adaptation against temporal correlation," in *Proc. NeurIPS*, 2022.
- [502] J. Zhang, L. Qi, Y. Shi, and Y. Gao, "Generalizable model-agnostic semantic segmentation via target-specific normalization," *Pattern Recognition*, vol. 122, p. 108292, 2022.
- [503] D. Wang, A. Ju, E. Shelhamer, D. Wagner, and T. Darrell, "Fighting gradients with gradients: Dynamic defenses against adversarial attacks," *arXiv preprint arXiv:2105.08714*, 2021.
- [504] Y. Tang, C. Zhang, H. Xu, S. Chen, J. Cheng, L. Leng, Q. Guo, and Z. He, "Neuro-modulated hebbian learning for fully test-time adaptation," in *Proc. CVPR*, 2023.
- [505] C. Yi, S. Yang, Y. Wang, H. Li, Y.-p. Tan, and A. Kot, "Temporal coherent test-time optimization for robust video classification," in *Proc. ICLR*, 2023.
- [506] A. Zhou and S. Levine, "Bayesian adaptation for covariate shift," in *Proc. NeurIPS*, 2021, pp. 914–927.
- [507] P. T. Sivaprasad and F. Fleuret, "Test time adaptation through perturbation robustness," in *Proc. NeurIPS Workshops*, 2021.
- [508] L. Jiang and T. Lin, "Test-time robust personalization for federated learning," in *Proc. ICLR*, 2023.
- [509] A. Singh and J. E. Ortega, "Addressing distribution shift at test time in pre-trained language models," in *Proc. AAAI Workshops*, 2022.
- [510] J. Song, J. Lee, I. S. Kweon, and S. Choi, "Ecotta: Memory-efficient continual test-time adaptation via self-distilled regularization," in *Proc. CVPR*, 2023.
- [511] H. Kingetsu, K. Kobayashi, Y. Okawa, Y. Yokota, and K. Nakazawa, "Multi-step test-time adaptation with entropy minimization and pseudo-labeling," in *Proc. ICIP*, 2022, pp. 4153–4157.
- [512] S. Choi, S. Yang, S. Choi, and S. Yun, "Improving test-time adaptation via shift-agnostic weight regularization and nearest source prototypes," in *Proc. ECCV*, 2022, pp. 440–458.
- [513] J. N. Kundu, S. Seth, P. YM, V. Jampani, A. Chakraborty, and R. V. Babu, "Uncertainty-aware adaptation for self-supervised 3d human pose estimation," in *Proc. CVPR*, 2022, pp. 20 448–20 459.
- [514] P. Yang, J. Liang, J. Cao, and R. He, "Auto: Adaptive outlier optimization for test-time ood detection," *arXiv preprint arXiv:2303.12267*, 2023.
- [515] P. Voigtlaender and B. Leibe, "Online adaptation of convolutional neural networks for video object segmentation," in *Proc. BMVC*, 2017.
- [516] D. Belli, D. Das, B. Major, and F. Porikli, "Online adaptive personalization for face anti-spoofing," in *Proc. ICIP*, 2022, pp. 351–355.
- [517] R. T. Mullapudi, S. Chen, K. Zhang, D. Ramanan, and K. Fatahalian, "Online model distillation for efficient video inference," in *Proc. ICCV*, 2019, pp. 3573–3582.
- [518] I. Shin, Y.-H. Tsai, B. Zhuang, S. Schuster, B. Liu, S. Garg, I. S. Kweon, and K.-J. Yoon, "Mm-tta: Multi-modal test-time adaptation for 3d semantic segmentation," in *Proc. CVPR*, 2022, pp. 16 928–16 937.
- [519] Z. Li, S. Shi, B. Schiele, and D. Dai, "Test-time domain adaptation for monocular depth estimation," in *Proc. ICRA*, 2023.
- [520] H. Yang, C. Chen, M. Jiang, Q. Liu, J. Cao, P. A. Heng, and Q. Dou, "Dlta: Dynamic learning rate for test-time adaptation on cross-domain medical images," *IEEE Transactions on Medical Imaging*, vol. 41, no. 12, pp. 3575–3586, 2022.
- [521] C. K. Mummadi, R. Huttmacher, K. Rambach, E. Levinkov, T. Brox, and J. H. Metzen, "Test-time adaptation to distribution shift by confidence maximization and input transformation," *arXiv preprint arXiv:2106.14999*, 2021.
- [522] S. Goyal, M. Sun, A. Raghunathan, and Z. Kolter, "Test-time adaptation via conjugate pseudo-labels," in *Proc. NeurIPS*, 2022.
- [523] J.-K. Wang and A. Wibisono, "Towards understanding gd with hard and conjugate pseudo-labels for test-time adaptation," in *Proc. ICLR*, 2023.
- [524] J. Han, L. Zeng, L. Du, W. Ding, and J. Feng, "Rethinking precision of pseudo label: Test-time adaptation via complementary learning," *arXiv preprint arXiv:2301.06013*, 2023.
- [525] S. Wang, D. Zhang, Z. Yan, J. Zhang, and R. Li, "Feature alignment and uniformity for test-time adaptation," in *Proc. CVPR*, 2023.
- [526] M. Döbler, R. A. Marsden, and B. Yang, "Robust mean teacher for continual and gradual test-time adaptation," *arXiv preprint arXiv:2211.13081*, 2022.
- [527] H. Ye, Y. Ding, J. Li, and H. T. Ng, "Robust question answering against distribution shifts with test-time adaptation: An empirical study," in *Proc. EMNLP Findings*, 2022.
- [528] Y. Gan, X. Ma, Y. Lou, Y. Bai, R. Zhang, N. Shi, and L. Luo, "Decorate the newcomers: Visual domain prompt for continual test time adaptation," in *Proc. AAAI*, 2023.
- [529] S. Yang, J. Wu, J. Liu, X. Li, Q. Zhang, M. Pan, Y. Gan, and S. Zhang, "Exploring sparse visual prompt for cross-domain semantic segmentation," *arXiv preprint arXiv:2303.09792*, 2023.
- [530] D. Brahma and P. Rai, "A probabilistic framework for lifelong test-time adaptation," *arXiv preprint arXiv:2212.09713*, 2022.
- [531] D. Tomar, G. Vray, B. Bozorgtabar, and J.-P. Thiran, "Tesla: Test-time self-learning with automatic adversarial augmentation," in *Proc. CVPR*, 2023.
- [532] W. Lin, M. J. Mirza, M. Kozinski, H. Possegger, H. Kuehne, and H. Bischof, "Video test-time adaptation for action recognition," *arXiv preprint arXiv:2211.15393*, 2022.
- [533] D. Das, S. Borse, H. Park, K. Azarian, H. Cai, R. Garrepalli, and F. Porikli, "Transadapt: A transformative framework for online test time adaptive semantic segmentation," *arXiv preprint arXiv:2302.14611*, 2023.

- [534] J. S. Lumentut and I. K. Park, "3d body reconstruction revisited: Exploring the test-time 3d body mesh refinement strategy via surrogate adaptation," in *Proc. ACM-MM*, 2022, pp. 5923–5933.
- [535] Y. Su, X. Xu, and K. Jia, "Revisiting realistic test-time training: Sequential inference and adaptation by anchored clustering," in *Proc. NeurIPS*, 2022.
- [536] J. Chen, X. Xian, Z. Yang, T. Chen, Y. Lu, Y. Shi, J. Pan, and L. Lin, "Open-world pose transfer via sequential test-time adaption," in *Proc. CVPR*, 2023.
- [537] S. Guan, J. Xu, Y. Wang, B. Ni, and X. Yang, "Bilevel online adaptation for out-of-domain human mesh reconstruction," in *Proc. CVPR*, 2021, pp. 10 472–10 481.
- [538] Y. Kuznietsov, M. Proesmans, and L. Van Gool, "Towards unsupervised online domain adaptation for semantic segmentation," in *Proc. WACV Workshops*, 2022, pp. 261–271.
- [539] J. Kim, I. Hwang, and Y. M. Kim, "Ev-tta: Test-time adaptation for event-based object recognition," in *Proc. CVPR*, 2022, pp. 17 745–17 754.
- [540] G. Chen, J. Zhang, X. Xiao, and Y. Li, "Graphtta: Test time adaptation on graph neural networks," in *Proc. ICML Workshops*, 2022.
- [541] Q. Wu, X. Yue, and A. Sangiovanni-Vincentelli, "Domain-agnostic test-time adaptation by prototypical training with auxiliary data," in *Proc. NeurIPS Workshops*, 2021.
- [542] Y. Su, X. Xu, T. Li, and K. Jia, "Revisiting realistic test-time training: Sequential inference and adaptation by anchored clustering regularized self-training," *arXiv preprint arXiv:2303.10856*, 2023.
- [543] M. J. Mirza, P. J. Soneira, W. Lin, M. Kozinski, H. Possegger, and H. Bischof, "Actmad: Activation matching to align distributions for test-time-training," *arXiv preprint arXiv:2211.12870*, 2022.
- [544] S. Jung, J. Lee, N. Kim, and J. Choo, "Cafa: Class-aware feature alignment for test-time adaptation," *arXiv preprint arXiv:2206.00205*, 2022.
- [545] S. Ambekar, Z. Xiao, J. Shen, X. Zhen, and C. G. Snoek, "Variational pseudo labels for meta test-time adaptation," 2023.
- [546] C. Wu, Y. Pan, Y. Li, and J. Z. Wang, "Learning to adapt to online streams with distribution shifts," *arXiv preprint arXiv:2303.01630*, 2023.
- [547] K. Fan, Y. Wang, Q. Yu, D. Li, and Y. Fu, "A simple test-time method for out-of-distribution detection," *arXiv preprint arXiv:2207.08210*, 2022.
- [548] T. Kerssies, J. Vanschoren, and M. Kılıçkaya, "Evaluating continual test-time adaptation for contextual and semantic domain shifts," *arXiv preprint arXiv:2208.08767*, 2022.
- [549] H. Zhao, Y. Liu, A. Alahi, and T. Lin, "On pitfalls of test-time adaptation," in *Proc. ICLR Workshops*, 2023.
- [550] S. Garg, Y. Wu, S. Balakrishnan, and Z. Lipton, "A unified view of label shift estimation," in *Proc. NeurIPS*, 2020, pp. 3290–3300.
- [551] C. Cortes, Y. Mansour, and M. Mohri, "Learning bounds for importance weighting," in *Proc. NeurIPS*, 2010, pp. 442–450.
- [552] G. J. McLachlan, *Discriminant analysis and statistical pattern recognition*. John Wiley & Sons, 1992.
- [553] S. Vucetic and Z. Obradovic, "Classification on data with biased class distribution," in *Proc. ECML*, 2001, pp. 527–538.
- [554] M. C. Du Plessis and M. Sugiyama, "Semi-supervised learning of class balance under class-prior change by distribution matching," in *Proc. ICML*, 2012, pp. 1219–1226.
- [555] —, "Semi-supervised learning of class balance under class-prior change by distribution matching," *Neural Networks*, vol. 50, pp. 110–119, 2014.
- [556] K. Perason, "On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling," *Philosophical Magazine*, vol. 50, pp. 157–175, 1900.
- [557] Y. S. Chan and H. T. Ng, "Estimating class priors in domain adaptation for word sense disambiguation," in *Proc. ACL*, 2006, pp. 89–96.
- [558] A. Alexandari, A. Kundaje, and A. Shrikumar, "Maximum likelihood with bias-corrected calibration is hard-to-beat at label shift adaptation," in *Proc. ICML*, 2020, pp. 222–232.
- [559] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in *Proc. ICML*, 2017, pp. 1321–1330.
- [560] Q. Sun, K. Murphy, S. Ebrahimi, and A. D'Amour, "Beyond invariance: Test-time label-shift adaptation for distributions with "spurious" correlations," *arXiv preprint arXiv:2211.15646*, 2022.
- [561] M. Sulc and J. Matas, "Improving cnn classifiers by estimating test-time priors," in *Proc. ICCV Workshops*, 2019.
- [562] T. Šipka, "Adaptation of cnn classifiers to prior shift," Master's thesis, Czech Technical University in Prague, August 2021.
- [563] M. Sulc, L. Pícek, J. Matas, T. Jeppesen, and J. Heilmann-Clausen, "Fungi recognition: A practical use case," in *Proc. WACV*, 2020, pp. 2316–2324.
- [564] J. Tian, Y.-C. Liu, N. Glaser, Y.-C. Hsu, and Z. Kira, "Posterior recalibration for imbalanced datasets," in *Proc. NeurIPS*, 2020, pp. 8101–8113.
- [565] Y. Zhang, B. Hooi, H. Lanqing, and J. Feng, "Self-supervised aggregation of diverse experts for test-agnostic long-tailed recognition," in *Proc. NeurIPS*, 2022.
- [566] C. Yang and J. Zhou, "Non-stationary data sequence classification using online class priors estimation," *Pattern Recognition*, vol. 41, no. 8, pp. 2656–2664, 2008.
- [567] R. Wu, C. Guo, Y. Su, and K. Q. Weinberger, "Online adaptation to label distribution shift," in *Proc. NeurIPS*, 2021, pp. 11 340–11 351.
- [568] R. Volpi, P. de Jorge, D. Larlus, and G. Csorka, "On the road to online adaptation for semantic image segmentation," in *Proc. CVPR*, 2022, pp. 19 184–19 195.
- [569] M. A. Jamal, H. Li, and B. Gong, "Deep face detector adaptation without negative transfer or catastrophic forgetting," in *Proc. CVPR*, 2018, pp. 5608–5618.
- [570] A. Roychowdhury, P. Chakrabarty, A. Singh, S. Jin, H. Jiang, L. Cao, and E. Learned-Miller, "Automatic adaptation of object detectors to new domains using self-training," in *Proc. CVPR*, 2019, pp. 780–790.
- [571] Z. Tang, Q. Jiao, J. Zhong, S. Wu, and H.-S. Wong, "Source-free unsupervised cross-domain pedestrian detection via pseudo label mining and screening," in *Proc. ICME*, 2022, pp. 1–6.
- [572] W. Ma, C. Chen, S. Zheng, J. Qin, H. Zhang, and Q. Dou, "Test-time adaptation with calibration of medical image classification nets for label distribution shift," in *Proc. MICCAI*, 2022, pp. 313–323.
- [573] Z. Wang, M. Ye, X. Zhu, L. Peng, L. Tian, and Y. Zhu, "Metateacher: Coordinating multi-model domain adaptation for medical image classification," in *Proc. NeurIPS*, 2022.
- [574] C. Saltori, E. Krivosheev, S. Lathuilière, N. Sebe, F. Galasso, G. Fiameni, E. Ricci, and F. Poiesi, "Gipso: Geometrically informed propagation for online adaptation in 3d lidar segmentation," in *Proc. ECCV*, 2022, pp. 567–585.
- [575] C. Saltori, S. Lathuilière, N. Sebe, E. Ricci, and F. Galasso, "Sf-uda^{3D}: Source-free unsupervised domain adaptation for lidar-based 3d object detection," in *Proc. 3DV*, 2020, pp. 771–780.
- [576] M. Choi, J. Choi, S. Baik, T. H. Kim, and K. M. Lee, "Scene-adaptive video frame interpolation via meta-learning," in *Proc. CVPR*, 2020, pp. 9444–9453.
- [577] T. Zhang, Y. Xiang, X. Li, Z. Weng, Z. Chen, and Y. Fu, "Free lunch for cross-domain occluded face recognition without source data," in *Proc. ICASSP*, 2022, pp. 2944–2948.
- [578] A. Gunawan, M. A. Nugroho, and S. J. Park, "Test-time adaptation for real image denoising via meta-transfer learning," *arXiv preprint arXiv:2207.02066*, 2022.
- [579] H. Yu, J. Huang, Y. Liu, Q. Zhu, M. Zhou, and F. Zhao, "Source-free domain adaptation for real-world image dehazing," in *Proc. ACM-MM*, 2022, pp. 6645–6654.
- [580] S. A. Hussein, T. Tirer, and R. Giryes, "Image-adaptive gan based reconstruction," in *Proc. AAAI*, 2020, pp. 3121–3129.
- [581] M. Z. Darestani, J. Liu, and R. Heckel, "Test-time training can close the natural distribution shift performance gap in deep learning based compressed sensing," in *Proc. ICML*, 2022, pp. 4754–4776.
- [582] A. Wu, W.-S. Zheng, X. Guo, and J.-H. Lai, "Distilled person re-identification: Towards a more scalable system," in *Proc. CVPR*, 2019, pp. 1187–1196.
- [583] R. Subramanyam, K. Thopalli, S. Berman, P. Turaga, and J. J. Thiagarajan, "Single-shot domain adaptation via target-aware generative augmentation," *arXiv preprint arXiv:2210.16692*, 2022.
- [584] J. Yin, B. Wang, Y. Dong, and C. Ling, "Source-free domain adaptation for question answering with masked self-training," *arXiv preprint arXiv:2212.09563*, 2022.
- [585] O. Antverg, E. Ben-David, and Y. Belinkov, "Idani: Inference-time domain adaptation via neuron-level interventions," in *Proc. NAACL Workshops*, 2022.

- [586] T. Volk, E. Ben-David, O. Amosy, G. Chechik, and R. Reichart, "Example-based hypernetworks for out-of-distribution generalization," *arXiv preprint arXiv:2203.14276*, 2022.
- [587] E. Laparra, X. Su, Y. Zhao, O. Uzuner, T. Miller, and S. Bethard, "Semeval-2021 task 10: Source-free domain adaptation for semantic processing," in *International Workshop on Semantic Evaluation (SemEval)*, 2021, pp. 348–356.
- [588] Y. Wang, C. Li, W. Jin, R. Li, J. Zhao, J. Tang, and X. Xie, "Test-time training for graph neural networks," *arXiv preprint arXiv:2210.08813*, 2022.
- [589] H. Mao, L. Du, Y. Zheng, Q. Fu, Z. Li, X. Chen, H. Shi, and D. Zhang, "Source free unsupervised graph domain adaptation," *arXiv preprint arXiv:2112.00955*, 2021.
- [590] P. Lee, S. Jeon, S. Hwang, M. Shin, and H. Byun, "Source-free subject adaptation for eeg-based visual recognition," in *Proc. BCI*, 2023.
- [591] Z. Liu and Y. Fang, "Learning adaptable risk-sensitive policies to coordinate in multi-agent general-sum games," *arXiv preprint arXiv:2303.07850*, 2023.
- [592] T. Lee, J. Tremblay, V. Blukis, B. Wen, B.-U. Lee, I. Shin, S. Birchfield, I. S. Kweon, and K.-J. Yoon, "Tta-cope: Test-time adaptation for category-level object pose estimation," in *Proc. CVPR*, 2023.
- [593] V. Borisov, T. Leemann, K. Seßler, J. Haug, M. Pawelczyk, and G. Kasneci, "Deep neural networks and tabular data: A survey," *IEEE Transactions on Neural Networks and Learning Systems*, 2022.
- [594] M. Ragab, E. Eldele, W. L. Tan, C.-S. Foo, Z. Chen, M. Wu, C.-K. Kwok, and X. Li, "Adatime: A benchmarking suite for domain adaptation on time series data," *ACM Transactions on Knowledge Discovery from Data*, 2023.
- [595] B. Eshete, "Making machine learning trustworthy," *Science*, vol. 373, no. 6556, pp. 743–744, 2021.