Domain Alignment Meets Fully Test-Time Adaptation

Kowshik Thopalli KTHOPALL@ASU.EDU
Arizona State University

Pavan Turaga PTURAGA@ASU.EDU
Arizona State University

Jayaraman J. Thiagarajan JJAYARAM@LLNL.GOV
Lawrence Livermore National Laboratory

Editors: Emtiyaz Khan and Mehmet Gönen

Abstract

A foundational requirement of a deployed ML model is to generalize to data drawn from a testing distribution that is different from training. A popular solution to this problem is to adapt a pre-trained model to novel domains using only unlabeled data. In this paper, we focus on a challenging variant of this problem, where access to the original source data is restricted. While fully test-time adaptation (FTTA) and unsupervised domain adaptation (UDA) are closely related, the advances in UDA are not readily applicable to TTA, since most UDA methods require access to the source data. Hence, we propose a new approach, CATTAn, that bridges UDA and FTTA, by relaxing the need to access entire source data, through a novel deep subspace alignment strategy. With a minimal overhead of storing the subspace basis set for the source data, CATTAn enables unsupervised alignment between source and target data during adaptation. Through extensive experimental evaluation on multiple 2D and 3D vision benchmarks (ImageNet-C, Office-31, OfficeHome, DomainNet, PointDA-10) and model architectures, we demonstrate significant gains in FTTA performance. Furthermore, we make a number of crucial findings on the utility of the alignment objective even with inherently robust models, pre-trained ViT representations and under low sample availability in the target domain.

Keywords: Test-Time Adaptation; Robustness; Domain Shifts; Geometric Alignment

1. Introduction

When the assumption that the training and testing data are drawn from the same distribution is violated, the performance of supervised models can drop drastically (Torralba and Efros, 2011). However, in practice, a deployed model is expected to generalize under unknown shifts in the data distribution (e.g., from synthetic to real). Consequently, understanding and improving the generalization of models under such shifts has become an active area of research (Hoffman et al., 2018; Ganin et al., 2016; Deng et al., 2018). This problem appears under a variety of formulations in the literature, including domain adaptation (Ben-David et al., 2006), domain generalization (Wang and Deng, 2018), few-shot adaptation (Triantafillou et al., 2021), and adversarial robustness (Chen et al., 2020).

. This work was performed under the auspices of the U.S. Department of Energy by the Lawrence Livermore National Laboratory under Contract No. DE-AC52-07NA27344, Lawrence Livermore National Security, LLC.

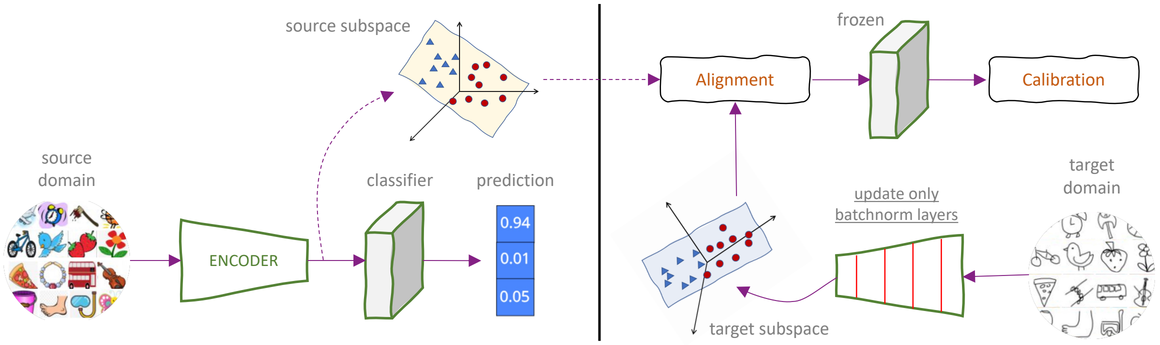


Figure 1: An overview of the proposed approach that incorporate subspace-based feature alignment for fully test-time adaptation. At test time, we only assume access to the trained source model and the subspace approximation of source latent features.

In this paper, we focus on unsupervised, fully test-time adaptation (`TTA`), where a deployed model is adapted using unlabeled data from the target domain, without assuming access to the original source data. This is a practically useful setting, since enabling access to source data during model deployment requires a large memory footprint for common datasets (e.g., ImageNet) and can also lead to shortcomings related to privacy and data usage rights. Existing `TTA` approaches can be organized based on a) whether data from the *source* domain can be accessed during adaptation; b) which parameters of the source model are updated; and c) whether data from the *target* domain is labeled or unlabeled. A closely related problem is unsupervised domain adaptation (`UDA`), which attempts to anticipate and adapt for distribution shifts between the labeled source data and unlabeled target data. Despite significant advances in UDA over the last decade, state-of-the-art solutions for `TTA` do not utilize explicit alignment objectives. This motivates our approach, `CATTAn` (Calibrate-by-Aligning for Test Time Adaptation) wherein we show that, by leveraging the latent space geometry, we can relax the requirement of source data access, and enable geometric alignment between source and target data at test-time. While our method requires access to source data in the form of basis vectors of a subspace spanned by the source features, it does not store the loadings (or coefficients). Consequently, this neither affects the memory overhead (the basis set requires less than 2 MB of storage in comparison to several GBs of training data) nor compromises the privacy needs, since state-of-the-art deep inversion methods (Behrmann et al., 2019; Yin et al., 2020; Dong et al., 2021) cannot effectively recover the training data using only features from later layers of a network, let alone with only the subspace basis. Using extensive empirical studies on several standard 2D image and 3D point cloud benchmarks, for the first time, we find that including unsupervised alignment in the cost function leads to significant performance gains over existing fully test-time adaptation methods.

Contributions:
(i) We propose a new test-time adaptation approach `CATTAn` to bridge UDA and `TTA`, while not requiring access to full source data;

(ii) We introduce a simple, post-hoc strategy to perform a distribution shift check after the model is already adapted to the target. Through this simple detector, we show that we can recover the source domain performance even after the model is adapted;
(iii) We perform rigorous empirical studies on large-scale vision benchmarks (ImageNet, DomainNet, OfficeHome,PointDA-10) and network architectures (ResNet50, ViT);
(iv) Our codes will be publicly released https://anonymous.4open.science/r/CATTAn.

Results:
(i) CATTAn produces SoTA results on all benchmarks, outperforming TENT (Wang et al., 2021), SHOT (Liang et al.), as well as the recent (Mummadi et al., 2021) – ImageNet-C (+2.1%), Office-home (+2.2%) and Office-31 (+1.7%);
(ii) To demonstrate the generality of our approach, we also conducted experiments on PointDA-10, a widely adopted 3D point cloud benchmark and observed that CATTAn improves over existing TTA baselines by +3.4%.
(iii) We conduct, for the first time, a FTTA experiment on the large-scale DomainNet (Peng et al., 2019) dataset, based on self-supervised representations from the recent ViT-based masked autoencoders (He et al., 2021). We find that CATTAn produces a boost of 1.1% over the best-performing TTA baseline, and matches the performance of a state-of-the art UDA approach (Roy et al., 2021);
(iv) We find that the proposed geometric alignment objective is beneficial even when the target sample size is limited or when the source model was obtained via robust training (Hendrycks et al., 2020).

2. Fully Test-Time Adaptation

Our goal is to improve the generalization of a model trained on the source dataset $\{(\mathrm{x}_s, y_s)\} \in \mathcal{D}_s$ to examples from the target domain $\{(\mathrm{x}_t)\} \in \mathcal{D}_t$ through adaptation under the following conditions – c1: $\mathcal{D}_s \neq \mathcal{D}_t$; c2: both source and target domain share the same set of labels; c3: examples from $\mathcal{D}_t$ are not labeled; and c4: there is no access to original source data samples during adaptation.

While this work focuses on unsupervised, fully test-time adaptation, a broad class of formulations have been considered in the literature for adapting models under distribution shifts. A popular formulation is conventional transfer learning, which first pre-trains a *source model* using data from $\mathcal{D}_s$, and uses labeled examples from $\mathcal{D}_t$ to perform end-to-end fine-tuning or partial adaptation of selected layers in the source network (Donahue et al., 2014; Yosinski et al., 2014). In contrast, unsupervised domain adaptation (UDA) jointly infers domain-invariant representations for both labeled source and unlabeled target domain examples, such that they both can utilize a shared classifier. Similarly, Sun et al. introduced a test-time training (TTT) protocol based on an auxiliary rotation angle prediction task, which also uses labeled source and unlabeled target examples.

Motivated by the need for source-free adaptation protocols, Liang et al. proposed SHOT that can effectively repurpose a source model, without requiring access to the original source data. Several variants of this approach have been proposed in the literature (Yang et al., 2021; Xia et al., 2021; Huang et al., 2021) and all of them rely on end-to-end fine-tuning, which can be a bottleneck in fully test-time adaptation (limited data as well as need for fast adaptation). Hence, recent methods such as `TENT` (Wang et al., 2021) and `IP` (Mummadi et al., 2021) update only the batch normalization layers of the source model.

Table 1: Comparing `CATTAn` to existing `FTTA` approaches. Conf. Max.: conditional entropy-/NLL, CB: class balance loss, BN: batchnorm.

SFTTA Methods	Losses				Updates		
	Conf. Max.	CB	pseudo lab.	Geom. Align.	BN params.	I/P Trans.	Align. Layer
Tent	✔	✕	✕	✕	✔	✕	✕
Tent+	✔	✔	✕	✕	✔	✕	✕
SHOT	✔	✔	✔	✕	✔	✕	✕
IP	✔	✔	✕	✕	✔	✔	✕
Ours	✔	✔	✕	✔	✔	✕	✔

Methodological gaps and Proposed Work. We begin by noting that the entropy minimization or other diversity promoting optimization strategies widely adopted by existing `TTA` methods can be viewed as calibrating predictions from a pre-trained classifier under distribution shifts (Shu et al., 2018). Furthermore, due to the source-free training assumption, they do not leverage any domain alignment objectives. Our work is aimed at closing this methodological gap by incorporating explicit domain alignment strategies from `UDA` into fully test-time adaptation. In particular, we employ a novel deep subspace alignment strategy to align the target and source subspaces during adaptation. This modification incurs only a negligible memory burden when compared to `TENT` (storing the basis vectors of a low-rank subspace). Table 1 shows how `CATTAn` compares to existing `FTTA` approaches.

3. Proposed Approach

As described in Section 2, our goal is to adapt a source model $\mathcal{F}_\theta$ with parameters θ to a (unlabeled) target domain at test-time. We express $\mathcal{F}_\theta := \mathcal{G}_\omega \circ \mathcal{H}_\psi$, as a composition of a feature extractor $\mathcal{G}_\omega$ with parameters ω and a classifier $\mathcal{H}_\psi$ with parameters ψ (*i.e.*, $\theta := \omega \cup \psi$). During adaptation, the classifier model $\mathcal{H}_\psi$ is frozen and only the target features are suitably modified.

3.1. Geometric Alignment Regularization

Upon training $\mathcal{F}_\theta$ on the source dataset, we extract the latent features for source data $\mathrm{Z}_s = \mathcal{G}_\omega(\mathrm{X}_s)$ where $\mathrm{Z}_s \in \mathbb{R}^{n_s \times D}$ and n_s is the number of source samples. We then compute a low-dimensional linear subspace with the basis $\mathrm{W}_s \in \mathbb{R}^{D \times d}$ that spans the source features Z_s using principal component analysis (PCA). Here, D denotes the ambient dimensionality of the latent space and d is the subspace dimension. For test-time adaptation, our approach stores this pre-computed basis set W_s in addition to the learned model parameters.

In order to introduce an alignment objective between the source and target features, we first extract features for the target data X_t *i.e.*, $\mathrm{Z}_t = \mathcal{G}_\omega(\mathrm{X}_t)$ and then perform a $d-$dimensional subspace approximation to obtain the corresponding basis W_t. Note, $\mathrm{W}_s^T\mathrm{W}_s = \mathbb{I}$ and $\mathrm{W}_t^T\mathrm{W}_t = \mathbb{I}$, where $\mathbb{I}$ is the identity matrix. The classical subspace alignment (`SA`) process estimates the transformation matrix Φ that aligns W_s and W_t:

$$\Phi^* = \arg\min_{\Phi} \|\mathrm{W}_t\Phi - \mathrm{W}_s\|_F^2 , \tag{1}$$

where, $\|.\|_F$ denotes the Frobenius norm. The solution to this objective can be obtained in closed form (Fernando et al., 2013) as

$$\Phi^* = (\mathrm{W}_t)^\top \mathrm{W}_s. \tag{2}$$

SA then projects Z_s onto the source subspace as $\mathrm{Z}_s\mathrm{W}_s$, and the target features Z_t onto aligned co-ordinate system (also referred to as the source-aligned target subspace) as $\mathrm{Z}_t\mathrm{W}_t\Phi$. However, naïve linear subspace alignment is known to be insufficient for modern datasets with large domain shifts. Hence, CATTAn uses deep subspace alignment (DSA) that addresses two main challenges: First, we equip DSA with the capability of re-utilizing the source classifier while performing alignment. To this end, we re-project the source-aligned target features into the ambient space as $\bar{\mathrm{W}}_t = \mathrm{W}_t\Phi^* = \mathrm{W}_t(\mathrm{W}_t)^T\mathrm{W}_s$ and solve

$$\hat{\mathrm{Z}}_t^* = \arg\min_{\hat{\mathrm{Z}}_t} \left\| \hat{\mathrm{Z}}_t\mathrm{W}_s - \hat{\mathrm{Z}}_t\bar{\mathrm{W}}_t \right\|_F^2 = \arg\min_{\hat{\mathrm{Z}}_t} \left\| \hat{\mathrm{Z}}_t\mathrm{W}_s - \hat{\mathrm{Z}}_t\mathrm{W}_t(\mathrm{W}_t)^\top\mathrm{W}_s \right\|_F^2, \tag{3}$$

where $\hat{\mathrm{Z}}_t^*$ denotes the modified target features. The solution to this optimization is

$$\hat{\mathrm{Z}}_t^* = \mathrm{Z}_t\mathrm{W}_t\Phi^*\mathrm{W}_s^\top. \tag{4}$$

Second, since the eventual goal is not optimal feature alignment but to maximally improve the performance of the model on target data, we include prediction calibration objectives. In such a setting, one can no longer obtain a closed-form solution for Φ^*. As a result, CATTAn uses the following subspace alignment cost in its objective:

$$\mathcal{L}_\Phi = \|\mathrm{W}_t\Phi - \mathrm{W}_s\|_F^2, \tag{5}$$

along with objectives that promote well-calibrated predictions on re-projected source-aligned target features $\hat{\mathrm{Z}}_t$ from (4). To enable end-to-end gradient-based training, we implement *subspace alignment* as a network $\mathcal{A}_\Phi(.)$ that parameterizes Φ using a fully connected layer of d neurons without any non-linear activation function or bias *i.e.* (4) now becomes

$$\hat{\mathrm{Z}}_t^* = \mathcal{A}_\Phi(\mathrm{Z}_t\mathrm{W}_t)\mathrm{W}_s^\top. \tag{6}$$

Note, we do not use non-linearity because if we include a non-linear activation, Φ and consequently $\mathrm{W}_t\Phi$ will fail to represent linear subspace alignment. Through extensive empirical studies in Section 4, we show that, this linear subspace alignment in deep latent spaces is highly effective at improving FTTA performance.

3.2. Prediction Calibration Objective

Calibrating the target predictions using methods such as conditional entropy minimization has been the most common objective in test-time adaptation under distribution shifts, which can be defined as $H(\hat{y}) = -\sum_c p(\hat{y}_c)\log p(y_c)$, where $\hat{y} = \mathcal{F}_\theta(\mathrm{x})$ are the predictions for x obtained using the model $\mathcal{F}_\theta$ and $p(\hat{y}_c)$ denotes the probability for sample x to be assigned to a specific class $c \in \mathcal{C}$. However, it has been found that entropy minimization can lead to vanishing gradients for high-confidence predictions, thus hindering the training process. Hence, we adopt the non-saturating loss function proposed by Mummadi et al.:

$$\mathcal{L}_{lr}(p(\hat{y})) = -\log\left(\frac{p(\hat{y}_{c^*})}{\sum_{i\neq c^*} p(\hat{y}_i)}\right) = -\log\left(\frac{e^{\hat{y}_{c^*}}}{\sum_{i\neq c^*} e^{\hat{y}_i}}\right) = -\hat{y}_{c^*} + \log\sum_{i\neq c^*} e^{\hat{y}_i},$$

Algorithm 1: Proposed algorithm for fully test-time adaptation

```
Input: Source-model $\mathcal{F}_\theta$; Source subspace $\mathrm{W}_s$; target data $\mathrm{X}_t$
Initialize: $\lambda_{lr}, \lambda_{cb}, n_{iter}$; Freeze classifier $\mathcal{H}_\psi$; Collect affine transformation parameters
 $\{\gamma_{l,m}, \beta_{l,m}\}$ for each normalization layer $l$ and channel $m$ in $\mathcal{G}_\omega$
Adaptation:
$\mathrm{Z}_t = \mathcal{G}_\omega \mathrm{X}_t$; // compute target features
$\mathrm{W}_t \leftarrow \mathrm{PCA}(\mathrm{Z}_t)$ // compute target subspace
Compute $\Phi^*$ using (2)
Initialize the weights of $\mathcal{A}_\Phi$ with $\Phi^*$
for iter in $n_{iter}$ do
    $\mathrm{Z}_t = \mathcal{G}_\omega(\mathrm{X}_t)$; // compute features for target samples
    $\hat{\mathrm{Z}}_t = \mathcal{A}_\Phi(\mathrm{Z}_t \mathrm{W}_t)\mathrm{W}_s^\top$ following (6) // project, align and re-project
    $\hat{\mathrm{y}}_t = \mathcal{F}_\theta(\hat{\mathrm{Z}}_t)$ // compute predictions for aligned target data
    $\mathcal{L} = \lambda_{lr}\mathcal{L}_{lr} + \mathcal{L}_\Phi + \lambda_{cb}\mathcal{L}_{CB}$ using (7) // compute overall objective
    Update alignment $\mathcal{A}_\Phi$ and parameters $\{\gamma_{l,m}, \beta_{l,m}\}$ of $\mathcal{G}_\omega$ w.r.t. $\mathcal{L}$
end
Output: $\mathcal{A}_\Phi^*, \mathcal{G}_\omega^*, \mathcal{F}_\theta^*$
```

where $c^* = \arg\max p(\hat{y})$. Since this likelihood ratio loss increases the gradient amplitude for high confidence predictions, this is found to be superior to entropy.

Class Balance Loss: We also include a popular class diversity objective $\mathcal{L}_{CB}$ to avoid trivial solutions that are biased towards a subset of the classes, since we perform adaptation using only unlabeled data. $\mathcal{L}_{CB}$ is implemented as the binary cross-entropy between the mean prediction from the network over a mini-batch and an uniform prior distribution.

Overall Objective: The overall objective of CATTAn is a combination of the alignment cost $\mathcal{L}_\Phi$, the prediction calibration term $\mathcal{L}_{lr}$ and the class balance loss $\mathcal{L}_{CB}$:

$$\mathcal{L} = \lambda_{lr}\mathcal{L}_{lr} + \mathcal{L}_\Phi + \lambda_{cb}\mathcal{L}_{CB}, \tag{7}$$

where the penalties λ_{lr}, λ_{cb} are hyper-parameters, the choice of which are not very sensitive as we discuss in our analysis (Sec. 7).

3.3. Algorithm

Initialization Phase: Similar to TENT (Wang et al., 2021), our method first collects the affine transformation parameters $\{\gamma_{l,m}, \beta_{l,m}\}$ for each normalization layer l and channel m in the source model. The remaining parameters $\theta \setminus \{\gamma_{l,m}, \beta_{l,m}\}$ are not updated during adaptation. As described in Section 3.1, our method computes the target features Z_t and fits a subspace to obtain W_t. We then initialize the deep subspace alignment layer $\mathcal{A}_\Phi$ with its weights initialized to Φ^* from (2).

Adaptation and Termination: In the forward pass, the outputs of the feature extractor $\mathcal{G}_\omega$ are transformed through $\mathcal{A}_\Phi$, re-projected using (6) and are passed to the classifier. We optimize for the parameters using the objective in (7). We repeat this process for the pre-specified number of epochs. We detail our approach in Algorithm 1.

Estimating subspace dimension: To select the optimal subspace dimension d, a hyper-parameter in our approach, we adopt the theoretical stability result from (Fernando et al., 2013) and modify it for the FTTA setting. For a given $\delta > 0$ and $\epsilon > 0$, we select the maximum subspace dimension d such that

$$\left(e_d^{\min} - e_{d+1}^{\min}\right) \geq \left(1 + \sqrt{\frac{\ln 2/\delta}{2}}\right)\left(\frac{16d^{3/2}}{\epsilon\sqrt{n_t}}\right), \quad (8)$$

where e_d represents the d^{th} eigenvalue and n_t denoting the number of samples in target domain. This theoretical bound gives us a selection rule for picking an optimal d. Given the principal components for both source and target datasets, and the corresponding eigenvalues, we compute the deviations $e_d - e_{d+1}$, $\forall d$, for both source and target data. Through (8), we then obtain a stable solution $d << D$ for a given δ and ϵ. In our experiments, we set $\delta = 0.1$ and $\epsilon = 10^6$. For example, we plot the values of the bound and $\left(e_d^{\min} - e_d^{\min}\right)$ w.r.t. to subspace dimension for the $A \rightarrow C$ case from OfficeHome in Figure 2 and we pick the value of $d = 800$.

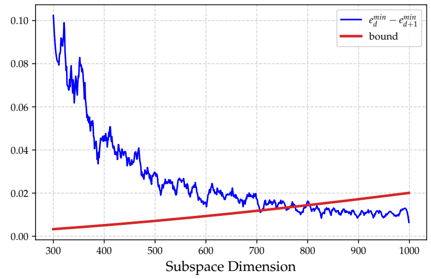


Figure 2: Estimating subspace dimensionality using (8) for the A→C setting from Office-Home. The lower bound is plotted in red and the difference between consecutive eigenvalues in blue.

4. Experiments

List of Experiments: In Table 2 we provide the details of different experiments we conducted, their goals and the models and datasets for a quick reference. In addition, we provide a discussion on hyper-parameter choices and ablations of our method.

Evaluation	Model	Datasets	Section
Utility of CATTAn for large-scale corruptions	ResNet-50	ImageNet → ImageNet-C	sec 5.1
Performance of CATTAn for common UDA benchmarks	ResNet-50	OfficeHome, Office-31	sec 5.2
CATTAn for 3D point-cloud classification	PointNET	PointDA-10	sec 5.3
Efficacy of CATTAn with pre-trained ViT embeddings	MAE with ViT as backbone	DomainNet	sec 6
Impact of target sample sizes	Resnet-50	OfficeHome	sec 7.2
Extending CATTAn to recover source performance	ResNet-50	OfficeHome	sec 7.3
Impact of robust training on CATTAn	Robust ResNet-50	OfficeHome	sec 7.4

Table 2: List of experiments.

Datasets: We evaluate CATTAn using standard UDA datasets along with a robustness benchmark, ImageNet → ImageNet-C. (i) The OfficeHome (Venkateswara et al., 2017) dataset is comprised of 15,500 images from 65 classes, where the images belong to 4 different domains; (ii) The Office-31 dataset (Saenko et al., 2010) contains 4110 images from 31 classes and represents three different domains; (iii) DomainNet (Peng et al., 2019) is a large scale UDA benchmark with 500K images from 6 domains with 345 classes each; (iv) ImageNet → ImageNet-C (Hendrycks and Dietterich, 2019) is a challenging corruption robustness benchmark that includes 15 types of synthetic corruptions with 5 severity levels; and (v) PointDA-10 is the first 3D point-cloud benchmark specifically designed for domain adaptation and comprises point-clouds belonging to 10 categories across 3 domains. In total, it contains approximately 27.7K training and 5.1K test samples.

Models: As our method operates under the FTTA setting, any arbitrary pre-trained model can be used. We experiment with the publicly available (pre-trained) Resnet-50 (He et al., 2016) model for evaluation on the ImageNet-C benchmark, and the modified Resnet-50 architecture from (Liang et al.) for the UDA benchmarks. Furthermore, we also experiment with a vision transformer(ViT) (Dosovitskiy et al., 2021)-based encoder (trained using masked auto-encoders (He et al., 2021)) finetuned on the ImageNet dataset. For PointDA-10, we use the PointNET (Qi et al., 2017) backbone proposed in PointDAN (Qin et al., 2019). As this model has only a single 1D BN layer, we extend the architecture with 4 additional 2D BN layers (*i.e.*, after the convolutional layers).

Baselines: We consider the following state-of-the-art FTTA methods for evaluation: (i) TENT (Wang et al., 2021); (ii) TENT+, a variant of TENTthat includes the class-balance loss defined in Section 3.2; (iii) The recent IP (Mummadi et al., 2021) approach that includes a learnable input transformation module (convolutional layers) to correct for the shifts; and (iv) SHOT (Liang et al.) that uses a pseudo-labeling based optimization strategy for test-time adaptation. Note that, the model architectures and the training protocols (e.g., update only BN layers) were fixed to be the same for all methods.

Metrics: We use the accuracy and empirical calibration error (ECE) (Guo et al., 2017) metrics for our evaluation.

Setup: For all UDA benchmarks, following standard practice, we considered each of the domains as source and adapted the source model to each of the target domains at test-time independently. We implemented CATTAn in PyTorch and used the Adam optimizer with learning rate $1e-4$ and set the batch size to 64. All experiments were repeated thrice with three different random seeds, and we report the average performance. Moreover, in cases where validation sets were not specified, we performed a $90-10$ random split, and used the validation split to select hyper-parameters. For IP and CATTAn, we set $\lambda_{lr} = 0.025$ in all our experiments. We implemented TENT and IP and generated results for Office-31, OfficeHome and DomainNet datasets, as their performance on these datasets have not been reported in their respective papers. We adapt TENT from the publicly available codebase [2], while we re-implemented IP, since their code was not publicly released. Following the strategy outlined in sec 3.3, we picked the subspace dimensionality d for our experiments. While TENT has been found to be useful for online adaptation (single epoch), Wang et al. found that performing the adaptation for more epochs consitently leads to better performance. Hence, in our experiments, we performed 5 epochs of adaptation for all methods.

5. FTTA Performance on 2D and 3D Benchmarks

5.1. ImageNet-C Benchmark

In Table 3, we report the performance of our proposed method, along with the baselines, on ImageNet-C at the highest severity level 5. It can be observed that the proposed method improves over TENT, TENT+ and SHOT by 6% points and IP by 2% points respectively. Among the baselines, IP performs the best - this can be attributed to the additional trainable input transformation module, which is typically well-suited for handling pixel-level corruptions.

[2] https://github.com/DequanWang/TENT

Table 3: Results on all 15 corruptions of **ImageNet-C** benchmark at the highest severity level-5 using standard Resnet50. Through the inclusion of an alignment objective, CATTAn improves significantly upon existing baselines.

Method	gauss	SHOT	impulse	defocus	glass	motion	zoom	snow	frost	fog	bright	contrast	elastic	pixel	jpeg	Avg.
Source Only	4.7	5.4	4.7	15.1	8.9	13.1	22.8	15.6	20.3	22.7	55.6	4.4	14.8	23.1	33.3	17.6
TENT	16.54	18.6	16.64	16.78	17	28.72	42.66	39.72	34.8	51.78	66.16	14.32	47.4	50.84	40.56	33.50
TENT+	16.96	19.1	17.3	17.1	17.56	29.22	42.82	40.04	35.4	51.82	65.82	15.78	47.64	50.88	40.68	33.87
SHOT	17.34	21.12	20	18.42	20.06	33.41	43.04	38.65	36.99	54.33	67.54	16.78	51.59	51.75	43.35	35.62
IP	23.94	26.88	25.06	23.2	22.62	36.28	48.7	46.58	39.44	56.08	67.58	18.6	53.1	55.58	48.76	39.49
Proposed	26.02	30.4	28.82	26.06	26.7	41.02	49.34	47.46	39.42	57.0	66.52	23.88	54.4	57	50.48	41.63

Table 4: Results on the **OfficeHome Dataset** obtained using Resnet50. Our approach improves upon existing FTTA baselines. Interestingly, IP and TENT+ baselines perform similarly, indicating that the input transformation module in IP is not effective at undoing larger domain shifts.

Method	A → C	A → P	A → R	C → A	C → P	C → R	P → A	P → C	P → R	R → A	R → C	R → P	Avg.	ECE
Source Only	43.73	65.35	72.94	52.62	61.07	64.77	51.17	40.53	73.01	64.65	45.25	77.27	59.36	0.56
TENT	47.88	65.98	73.26	58.76	65.94	68.07	60.16	47.31	75.4	70.83	53.95	78.73	63.85	0.09
TENT+	51.48	69.07	74.39	59.21	67.52	69.43	60.49	50.1	76.34	70.83	56.29	79.82	65.41	0.07
SHOT	50.61	68.69	74.71	58.34	67.63	70.07	57.73	49.14	76.38	69.47	54.89	79.88	64.795	0.06
IP	52.16	69.09	74.57	59.7	67.79	69.31	60.2	50.63	75.72	70.58	56.38	79.61	65.47	0.07
Proposed	52.81	73.89	77.07	61.93	71.12	72.94	61.89	52.35	79.05	72.11	56.68	80.27	67.68	0.08

However, by not adopting an explicit alignment objective and using only the prediction calibration process to guide the adaptation, IP produces lower performance than CATTAn, which does not employ any image-space transformation.

5.2. UDA Benchmarks

We demonstrate the efficacy of our method under large distribution shifts found in typical UDA problems by performing experiments with OfficeHome and Office-31 datasets. The comparative results for these two datasets can be found in Tables 4 and 5 respectively. Similar to the observations from the previous experiment, CATTAn consistently performs better than the existing FTTA baselines. On OfficeHome, CATTAn improves upon TENT, IP and SHOT by $3.8, 2.2$ and 2.8% points respectively, while on Office-31 CATTAn produces gains of $2.4, 1.8$ and 3.04% points. Interestingly, while the input transformation module proposed in IP is useful with pixel-level corruptions, it is not able to achieve invariance to the large semantic shifts that occur in typical domain adaptation benchmarks. As a result, the performance of IP tends to be similar to that of TENT+. In contrast, the latent subspace alignment strategy adopted by CATTAn produces large performance gains over TENT+. As we consider more complex datasets with large diversities between domains going forward, we compare our method against the more general and stronger baseline TENT+.

5.3. 3D point-cloud Dataset

As discussed earlier, our latent space alignment strategy is applicable to different model architectures or data modalities. In order to demonstrate this, we experimented with a recent 3D point-cloud classification DA benchmark (PointDA-10). As shown in Table 5.3,

Table 5: Adaptation results for **Office31 Dataset** obtained using Resnet50. We observe that `CATTAn` consistently improves upon state-of-the-art `FTTA` approaches.

Method	A → C	A → P	A → R	C → A	C → P	C → R	Avg.	ECE
Source Only	81.12	74.47	61.34	94.34	62.62	97.39	78.54	0.6
`TENT`	82.13	85.16	68.83	97.48	62.94	99.8	82.72	0.11
`TENT+`	82.33	85.66	69.72	97.61	65.03	99.8	83.35	0.10
`SHOT`	80.72	82.64	67.59	97.23	64.54	99.8	82.08	0.18
`IP`	82.73	85.28	69.12	97.99	65.35	100	83.41	0.07
Proposed	85.54	86.29	72.88	98.62	67.59	99.8	85.12	0.07

the adaptation performance of `CATTAn` is significantly superior to `TENT` and `TENT+` by 3.4% and 2.8% points respectively (averaged across 6 experiments). Especially, in cases such as Model→Shape and Scan→Model, `CATTAn` improves upon `TENT+` by more than 6% points while matching its performance in challenging settings such as Model→Scan. This clearly evidences the effectiveness of our approach across different problem settings.

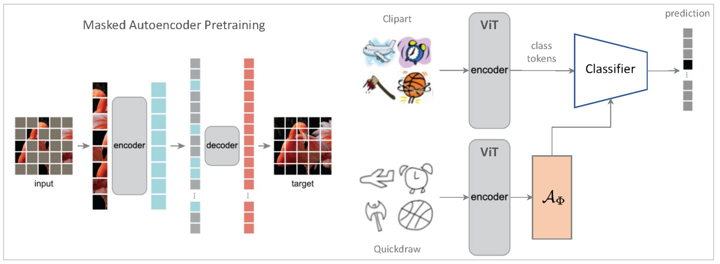
Table 6: Adaptation results on the **PointDA-10**, a 3D point cloud classification benchmark. We observe that the proposed approach provides an improvement of over 1.5%, thus evidencing its generality across model architectures and data modalities.

	Model→Shape	Model→Scan	Shape→Model	Shape→Scan	Scan→Model	Scan→Shape	Mean
Source Only	52.1	15.74	51.32	12.79	38.82	52.14	37.15
`TENT`	54.69	23.38	51.82	28.4	38.1	53.44	41.97
`TENT+`	56.2	23.32	52.33	27.41	42.48	53.71	42.58
`CATTAn`	62.41	22.83	54.44	27.11	48.58	57.07	45.41

6. `CATTAn` with Pre-Trained ViT Embeddings

As Transformer-based solutions such as vision transformers (ViT) (Dosovitskiy et al., 2021) and masked auto encoders (MAE) (He et al., 2021) are becoming increasingly popular and achieve state-of-art performance in solving vision problems, it is imperative to understand the efficacy of our alignment strategy on feature representations obtained from such large-scale pre-trained transformer encoders. To this end, we consider the encoder from MAE (He et al., 2021) fine-tuned on ImageNet as our feature extractor [3]. As illustrated in Figure 3, MAE first masks a large portion of the image and attempts to reconstruct the complete image from the masked image. Once trained via this self-supervision, the encoder is then fine-tuned with ImageNet data. We then freeze the encoder, obtain source features (class tokens) Z_s and perform PCA to obtain the basis W_s. Note that, we do not fine-tune the ViT with source domain data, but instead use it as an off-the-shelf feature extractor. A source model, which is comprised of a single MLP layer with batch normalization and a linear classifier layer, is then constructed and trained using Z_s. During adaptation with unlabeled target data, we extract features Z_t from the frozen encoder and obtain subspace

[3] https://github.com/facebookresearch/mae/blob/main/FINETUNE.md



Method	Avg.	ECE
Source Only	30.89	0.42
TENT	33.07	0.09
TENT+	33.22	0.1
Proposed	34.17	0.08
Multi-target adaptation (Roy et al., 2021)	34.4	-

Figure 3: Implementing CATTAn with pre-trained representations from Masked Autoencoders. The table shows the FTTA performance on **DomainNet** with representations from MAE (He et al., 2021).

basis vectors W_t. Using the optimization procedure of CATTAn, the batch normalization parameters and the subspace alignment module $\mathcal{A}_\Phi$ are then estimated.

In this experiment, we used the large-scale DomainNet (Peng et al., 2019) benchmark. Given that there are 6 domains in this dataset, we conducted a total of 30 test-time adaptation experiments, where we consider one domain as source and the other domains as target. In Figure 3, we report the average performance obtained on this benchmark. It can be seen that CATTAn improves upon TENT+ by almost 1% (averaged over 30 experiments), thus indicating that even under this challenging setting, the alignment objective plays an important role. For comparison, we also include the result from a state-of-the-art multi-target domain adaptation approach (Roy et al., 2021), which trains a Resnet-101 model end-to-end with combined source and target data. Our results show that, with a powerful feature encoder, simple test-time adaptation with CATTAn can produce similar performance. This clearly emphasizes the improved representational power of modern pre-training strategies, as well as the efficacy of CATTAn in aligning disparate domains even without accessing the entire source data.

7. Analysis

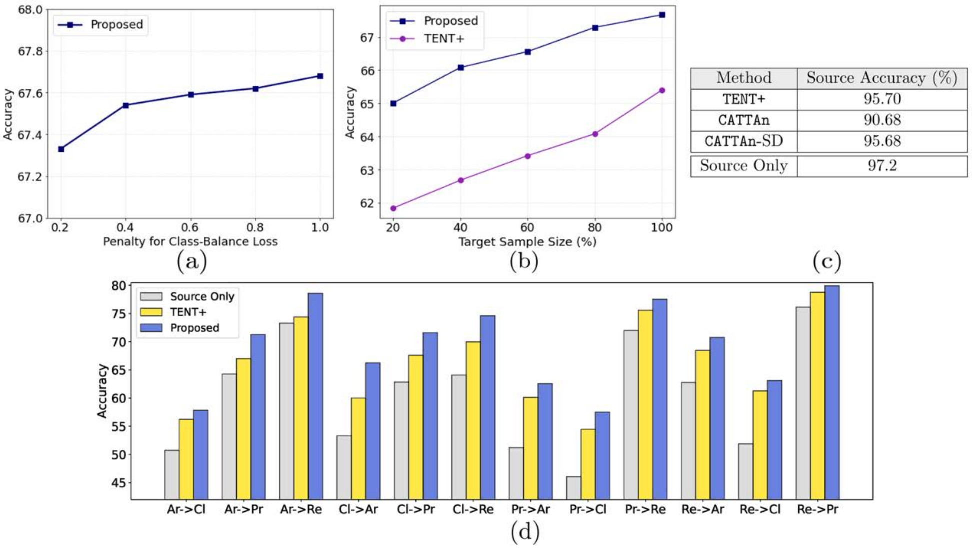
7.1. Behavior of CATTAn

Role of different loss terms: In Table 1, we highlight the differences between the different SoTA FTTA baselines and our method. Our extensive empirical study with multiple SoTA benchmarks and these baselines clearly show that the proposed geometric alignment is critical for the reported performance gains.

Choice of λ_{cb}: Using OfficeHome (Venkateswara et al., 2017) dataset, we study the sensitivity of CATTAn w.r.t. change in the value of λ_{cb}. As can be evidenced from Figure 4(a), for values greater than 0.4 the performance of CATTAn is stable with respect to changes in λ_{cb}. In our experiments, we fixed λ_{cb} to be 1.0.

7.2. Impact of Target Sample Sizes

Next, we study the impact of CATTAn under varying sample complexity in the target domain. While CATTAn is not expected to be very effective for online adaptation with a small batch (due to the poor quality of subspace fit in high-dimensions with sparse examples), we make



Method	Source Accuracy (%)
TENT+	95.70
CATTAn	90.68
CATTAn-SD	95.68
Source Only	97.2

Figure 4: Analysis of CATTAn - (a) Performance of TENT+ and CATTAn for varying target sample sizes; (b) Performance of CATTAn on the OfficeHome dataset with varying values of class-balance penalty λ_{cb}; (c) Enabling CATTAn to recover performance on the original source data through a simple test-time shift detection mechanism; (d) FTTA performance with a robust source model – CATTAn improves upon the baselines, this indicating that the proposed approach provides non-trivial invariances not captured by a robust model.

an interesting finding that, even at significantly reduced sample sizes, CATTAn produces superior adaptation performance than strong baselines such as TENT+. In Figure 4(b), we plot the performance of TENT+ and CATTAn at different sample sizes. Unsurprisingly, the performance drops as the target size decreases but importantly, the drop in performance of CATTAn is less severe than that of TENT+.

7.3. Extending CATTAn to Recover Source Performance

From our experiments with several DA benchmarks, we find that using an explicit alignment objective leads to significantly improved test performance. However, this comes at a cost of a drop in accuracy for samples from the original source domain after adaptation. This is an inherent challenge with methods that include an explicit alignment during adaptation. As a toy example, consider the case where the samples from target domain are rotated versions of samples from source domain by a certain angle. In this case, ideally $\mathcal{A}_\Phi$ would be the matrix that will undo this rotation. However, at test-time, if the same $\mathcal{A}_\Phi$ is applied to source data, the classifier will fail, as this creates a new domain shift of rotating by twice the angle. To address this issue, we propose a post-hoc strategy to determine if a test sample belongs to the target domain or OOD (*i.e.*, from the source domain). If the sample is OOD, $\mathcal{A}_\Phi$ is no longer applicable and we replace it with $\mathcal{A}_\Phi = \mathbb{I}$ (identity matrix). Note that, this approach for OOD detection does not require access to source data and hence is applicable with any off-the-shelf model.

Our post-hoc mechanism is inspired by the recent results in measuring generalization gap using inconsistencies between multiple hypotheses in a deep ensemble (Jiang et al., 2021). Let us assume that we have K different hypotheses $\{\mathcal{F}_\theta^1 \cdots \mathcal{F}_\theta^K\}$ for the prediction function that we want to approximate. We rely on the inter-hypothesis consistency to check for distribution-shifts. The intuition here is that, an OOD sample has a higher risk of having inconsistent predictions across the different hypotheses. Specifically, for each sample we average the prediction probabilities from $K-1$ models and assign the label corresponding to the class that has the highest probability. We compare this prediction against the prediction of K^{th} model. We repeat this for all k in K *i.e.*

$$q(x) = \sum_{k=1}^{K} \mathbb{1}\left[\frac{1}{K-1}\left(argmax\left(\sum_{i,i\neq k}^{K} \mathcal{F}_\theta^i(x)\right)\right) == argmax(\mathcal{F}_\theta^k(x))\right] \tag{9}$$

We finally obtain the normalized score for the inconsistency $\bar{q}(x)$ as $\bar{q}(x) = \frac{q(x)}{K}$. Intuitively, larger the q value for a given sample at test-time, higher is the likelihood for it to be drawn from the target distribution. To facilitate this comparison, we compare $\bar{q}(x)$ against a user defined threshold τ *i.e.*, if $\bar{q}(x) < \tau$ then use $\mathcal{A}_\Phi = \mathbb{I}$. We set the τ to be 0.75 in our experiments.

We adopt the following approach to construct the multiple source hypotheses. The first hypothesis construction follows the procedure explained in Section 3.1. For the second hypothesis, we fit another target subspace but only to two-thirds of the target samples that had the lowest confidence values and recompute the target subspace using them. Note, even with this new target subspace, the adaptation process uses the entire target data. In other words, the diversity in the hypotheses arises mainly from the different target subspace fits. In our experiments, we set $K = 3$. As can be seen from Figure 4(c) this simple test can effectively recover the accuracy on the source dataset. We also note that, this test can be applied to any `FTTA` method (e.g., to avoid applying the input transformation to a source sample in `IP`).

7.4. Impact of Robust Training on `CATTAn`

Having empirically established the effectiveness of `CATTAn` in achieving state-of-the-art results using standard models such as ResNet-50, a natural question to then ask is if the alignment cost still helps when we consider an inherently robust model instead. To answer this, we conducted experiments with a robust ResNet50 model (Hendrycks et al., 2021), trained with DeepAugment and Augmix (Hendrycks et al., 2020) strategies. From Figure 4(d), we first notice that the source-only performance is improved by more than 1.3% points when compared to the standard model, which indicates that a robust model already captures at least some of the invariant properties. Furthermore, we notice that both `TENT+` and `CATTAn` improve upon no adaptation performance significantly, with average accuracies 66.1% and 69.29% respectively. With a performance boost of more than 3% points than `TENT+`, `CATTAn` clearly evidences the importance of including an alignment objective for `FTTA` even with a robust model. We also remark that, since most standard robustness training paradigms do not include large distribution shifts such as the diverse shifts encountered in UDA datasets, alignment techniques such as `CATTAn` provide complementary benefits.

8. Conclusions

In this work, we explored the benefit of including an alignment objective in test time adaptation. First, we show that we can bridge UDA and TTA solutions without requiring access to complete source data. Through `CATTAn` we incorporated deep subspace alignment and demonstrated the effectiveness of alignment across different benchmarks. Using rigorous empirical studies, we showed that our method consistently improves and outperforms the state-of-the-art methods on several 2D image and 3D point-cloud benchmarks. We also showed the effectiveness of the proposed method when we consider a powerful feature extractor such as Vision transformers and robust models. Interestingly, our method is robust at even low sample sizes. We also proposed a novel post hoc algorithm that can be applied after the model is adapted to target domain such that the model is still effective on the source data. Future extensions of the work include extending to other vision tasks such as semantic segmentation and exploring other alignment techniques which do not require source data.

Acknowledgments

KT's and PT's work are supported by the Defense Advanced Research Projects Agency (DARPA) under Agreement No. HR00112290073. JT's work was supported by the LDRD Program under project 21-ERD-012.

References

Jens Behrmann, Will Grathwohl, Ricky TQ Chen, David Duvenaud, and Jörn-Henrik Jacobsen. Invertible residual networks. In *International Conference on Machine Learning*, pages 573–582. PMLR, 2019.

Shai Ben-David, John Blitzer, Koby Crammer, and Fernando Pereira. Analysis of representations for domain adaptation. In *NIPS*, 2006.

Tianlong Chen, Sijia Liu, Shiyu Chang, Yu Cheng, Lisa Amini, and Zhangyang Wang. Adversarial robustness: From self-supervised pre-training to fine-tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 699–708, 2020.

Weijian Deng, Liang Zheng, Qixiang Ye, Guoliang Kang, Yi Yang, and Jianbin Jiao. Image-image domain adaptation with preserved self-similarity and domain-dissimilarity for person reidentification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 994–1003, 2018.

J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *ICML*, 2014.

Xin Dong, Hongxu Yin, Jose M. Alvarez, Jan Kautz, and Pavlo Molchanov. Deep neural networks are surprisingly reversible: A baseline for zero-shot inversion, 2021. URL https://arxiv.org/abs/2107.06304.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

Basura Fernando, Amaury Habrard, Marc Sebban, and Tinne Tuytelaars. Unsupervised visual domain adaptation using subspace alignment. In *Proceedings of the IEEE International Conference on Computer Vision, (ICCV)*, pages 2960–2967, 2013.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The journal of machine learning research*, 17(1):2096–2030, 2016.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1321–1330. JMLR. org, 2017.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, June 2016.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. *arXiv preprint arXiv:2111.06377*, 2021.

Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=HJz6tiCqYm.

Dan Hendrycks, Norman Mu, Ekin D Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. Augmix: A simple data processing method to improve robustness and uncertainty. In *ICLR*, 2020.

Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8340–8349, 2021.

Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei Efros, and Trevor Darrell. Cycada: Cycle-consistent adversarial domain adaptation. In *International conference on machine learning*, pages 1989–1998. PMLR, 2018.

Jiaxing Huang, Dayan Guan, Aoran Xiao, and Shijian Lu. Model adaptation: Historical contrastive learning for unsupervised domain adaptation without source data. *Advances in Neural Information Processing Systems*, 34:3635–3649, 2021.

Yiding Jiang, Vaishnavh Nagarajan, Christina Baek, and J Zico Kolter. Assessing generalization of sgd via disagreement. *arXiv preprint arXiv:2106.13799*, 2021.

Jian Liang, Dapeng Hu, and Jiashi Feng. Do we really need to access the source data? Source hypothesis transfer for unsupervised domain adaptation. In *Proceedings of the 37th International Conference on Machine Learning*.

Chaithanya Kumar Mummadi, Robin Hutmacher, Kilian Rambach, Evgeny Levinkov, Thomas Brox, and Jan Hendrik Metzen. Test-time adaptation to distribution shift by confidence maximization and input transformation. *arXiv preprint arXiv:2106.14999*, 2021.

Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1406–1415, 2019.

Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017.

Can Qin, Haoxuan You, Lichen Wang, C-C Jay Kuo, and Yun Fu. Pointdan: A multi-scale 3d domain adaption network for point cloud representation. *Advances in Neural Information Processing Systems*, 32, 2019.

Subhankar Roy, Evgeny Krivosheev, Zhun Zhong, Nicu Sebe, and Elisa Ricci. Curriculum graph co-teaching for multi-target domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5351–5360, 2021.

Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting visual category models to new domains. In *European conference on computer vision*, pages 213–226. Springer, 2010.

Rui Shu, Hung H Bui, Hirokazu Narui, and Stefano Ermon. A dirt-t approach to unsupervised domain adaptation. In *ICLR*, 2018.

Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *Proceedings of the 37th International Conference on Machine Learning*.

Antonio Torralba and Alexei A Efros. Unbiased look at dataset bias. In *CVPR 2011*, pages 1521–1528. IEEE, 2011.

Eleni Triantafillou, Hugo Larochelle, Richard Zemel, and Vincent Dumoulin. Learning a universal template for few-shot dataset generalization. In *International Conference on Machine Learning*, pages 10424–10433. PMLR, 2021.

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

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=uXl3bZLkr3c.

Mei Wang and Weihong Deng. Deep visual domain adaptation: A survey. *Neurocomputing*, 312: 135–153, 2018.

Haifeng Xia, Handong Zhao, and Zhengming Ding. Adaptive adversarial network for source-free domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9010–9019, 2021.

Shiqi Yang, Joost van de Weijer, Luis Herranz, Shangling Jui, et al. Exploiting the intrinsic neighborhood structure for source-free domain adaptation. *Advances in Neural Information Processing Systems*, 34:29393–29405, 2021.

Hongxu Yin, Pavlo Molchanov, Jose M Alvarez, Zhizhong Li, Arun Mallya, Derek Hoiem, Niraj K Jha, and Jan Kautz. Dreaming to distill: Data-free knowledge transfer via deepinversion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8715–8724, 2020.

Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In *NeurIPS*, 2014.