On Generalizing Beyond Domains in Cross-Domain Continual Learning

Christian Simon[1,3,5] Masoud Faraki[2] Yi-Hsuan Tsai[6] Xiang Yu[2]
Samuel Schulter[2] Yumin Suh[2] Mehrtash Harandi[3,5] Manmohan Chandraker[2,4]

[1]The Australian National University [2]NEC Labs America [3]Monash University
[4]University of California, San Diego [5]Data61 [6]Phiar Technologies

sen.christiansimon@gmail.com, mfaraki@nec-labs.com, wasidennis@gmail.com
{xiangyu, samuel, yumin}@nec-labs.com, mehrtash.harandi@monash.edu, manu@nec-labs.com

Abstract

Humans have the ability to accumulate knowledge of new tasks in varying conditions, but deep neural networks often suffer from catastrophic forgetting of previously learned knowledge after learning a new task. Many recent methods focus on preventing catastrophic forgetting under the assumption of train and test data following similar distributions. In this work, we consider a more realistic scenario of continual learning under domain shifts where the model must generalize its inference to an unseen domain. To this end, we encourage learning semantically meaningful features by equipping the classifier with class similarity metrics as learning parameters which are obtained through Mahalanobis similarity computations. Learning of the backbone representation along with these extra parameters is done seamlessly in an end-to-end manner. In addition, we propose an approach based on the exponential moving average of the parameters for better knowledge distillation. We demonstrate that, to a great extent, existing continual learning algorithms fail to handle the forgetting issue under multiple distributions, while our proposed approach learns new tasks under domain shift with accuracy boosts up to 10% on challenging datasets such as DomainNet and OfficeHome.

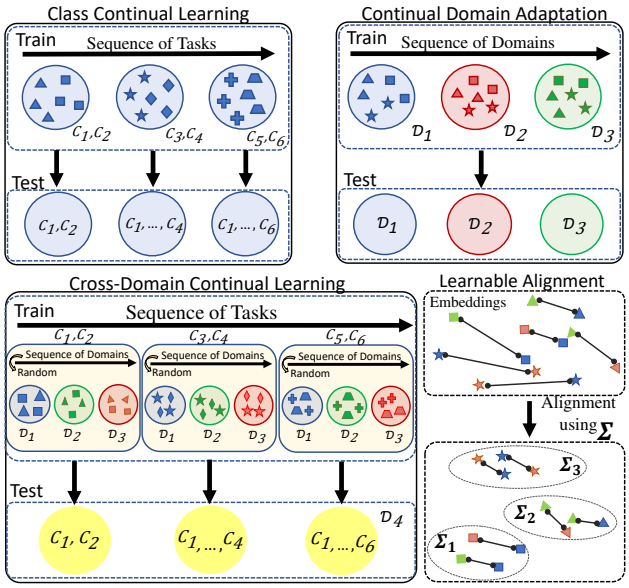


Figure 1. Top: Existing settings on 1) continually learning new visual categories from single domain (left), and 2) continually learning from new domains with evaluation on the same domains (right). Bottom: Our setting which has a sequence of visual categories coming from various domains, with evaluation on an unseen domain. The proposed approach utilizes a continual domain alignment strategy dubbed Mahalanobis Similarity Learning (MSL). Colors indicate domains, and shapes indicate categories.

1. Introduction

Humans possess the extraordinary capability of acquiring new knowledge in dynamically changing environments, while preserving knowledge learned in the past. The obtained knowledge can be further generalized to unseen situations without the need of re-educating. On the other hand, there has been a surge of efforts to devise machine learning based algorithms to build more intelligent models and mitigate the aforementioned challenges from two perspectives, namely continual learning [2, 4, 26, 32] and domain generalization [12, 13, 20, 21]. This is particularly more important when deployed in the real world under a life-long learning setup [18, 25, 28]. For instance, consider warehouse robots that might perceive new inventory or unseen room layouts that they require to adapt to function properly. The observations are captured at different time frames (*e.g.*, day or night) and different locations (*e.g.*, aisles) such that the observed domains come with an unpredictable sequence. In these situations, the key to success is to have certain embedded

adaptability in the robots to handle the challenges without costly re-training or entirely replacing them.

To put the discussion into perspective, on one hand, continual learning based methods mainly try to deal with *catastrophic forgetting*, which refers to the performance degradation of previously acquired knowledge when new concepts are learned. On the other side, domain generalization is to find a good feature representation that goes well beyond the training distributions, while at the same time being discriminative for the task at hand. While effective, there has been comparatively little efforts in research to provide answers for the two aforementioned challenges simultaneously. One effort is the work of Volpi *et al*. [35] which proposes continual domain adaptation, *i.e*., where different domains arrive in a continual fashion (top-right in Fig. 1). Other similar effort includes the work of Kundu *et al*. [19], which suggests class-continual learning with source-target domain adaptation as in the open-set setting. However, in both works the main aspect of *generalization* beyond seen domains is largely missing, limiting applicability of them in real-world scenarios. Moreover, the notion of incrementally adding training tasks is constrained to source and target domains only (*i.e*., two tasks).

In this work, we propose an approach for cross-domain continual learning, which also has the capability of generalization to unseen domains. Our setup considers a sequence of tasks (*i.e*. different visual categories), where each task's data is originated from various domains (Fig. 1 bottom-left). Note that, our setup does not have any prior assumptions about the domains (*e.g*., availability of domain identifiers or specific orderings) associated to the given training samples in each task. This is a realistic scenario where the model is deemed to be agnostic about the origin of training samples, *e.g*., when preserving privacy is important. We deem the domain alignment be done in a discriminative manner by equipping our classifier with class-specific Mahalanobis similarity metrics, as shown in Fig. 1 (bottom-right). Here, the classifier network also takes into account the underlying distribution of the class samples when generating the predictions. This is to encourage learning semantically meaningful features across training domains. We then learn the backbone representation parameters along with these extra parameters in an end-to-end fashion. In addition, we propose an approach based on the exponential moving average of the parameters for better knowledge distillation, preventing excessive divergence from the previously learned parameters.

To evaluate our method, we define highly dynamic environments with data coming from various domains and expanding visual categories. We perform extensive experiments on four different datasets – DomainNet [27], OfficeHome [34], PACS [21], and NICO [14]. The results show that our method consistently leads to an improvement of up to 10% compared to baselines [16, 23, 30, 38, 40] on 10-task and 5-task protocols. Furthermore, our proposed method also prevents *catastrophic forgetting*, achieving the lowest backward transfer rate [25] on average, *e.g*., $\sim$10% and $\sim$8% on DomainNet and OfficeHome, respectively.

To summarize, our contributions include

1. We provide a unified testbed for cross-domain continual learning with comparison to continual learning methods and techniques for domain generalization.

2. We propose a projection technique in an end-to-end scheme for domain generalization. In particular, we make use of learnable Mahalanobis similarity metrics for robust classifiers against unseen domains.

3. We devise an exponential moving average framework for knowledge distillation. The proposed module is integrated with our learnable projection technique to alleviate the degrading impact of catastrophic forgetting and distributional shifts by adaption to a history of the old parameters.

2. Related Work

Continual Learning. To tackle the forgetting problem in continual learning, neural networks must maintain the performance of past visual categories. Knowledge Distillation [3, 15] (KD) between an old model and a current model is an effective approach to prevent catastrophic forgetting. The standard baseline exploits KD as proposed by Li and Hoiem [23], for which the predictions between old and current models are preserved. Hou *et al*. [16] propose a KD method in feature space, in order to maintain the features from old and current models. In the same line of research, Simon *et al*. [30] introduce a smooth property to learn from one task to another task such that the geometry aspect is taken into account. Another stream of continual learning methods consider memory selection and generation for memory replay. A classical approach is known as herding [36] by picking the nearest neighbors from the mean of exemplars in each class. Another approach in this category is gradient episodic memory [5, 25] that uses old training data to impose optimization constraints when learning new tasks. Liu *et al*. [24] use bi-level optimization for synthesizing the memory, and a more optimal memory replay is expected compared to storing exemplars from training data. Despite the wide use of generating and selecting exemplars for continual learning, it does not guarantee robustness under change of train and test distributions.

Domain Generalization. Domain generalization techniques aim to generalize beyond training domains, which is a different goal compared to domain adaptation that reduces the distributional shifts between source and target domains. The problem of domain generalization also differs from few-shot or unsupervised domain adaptation where

in these problems the test data is accessible during training [11, 37]. A standard approach is to expose a model with various domains in training as recommended in [33] under empirical risk minimization. This straightforward idea from supervised learning is effective for domain generalization as also shown in [13]. An extension is to adapt the risk minimization loss to a context network with a meta learning strategy as proposed in [38]. To improve generalization, Zhou *et al*. [40] propose a smooth style transfer applied to the feature statistics. Though these techniques are effective to generalize to unseen distributions, their ability to deal with a stream of data containing multiple tasks is still questionable.

Learning Embeddings. To compute the distance between a pair of points, a projection matrix (*e.g*., covariance, Positive Semi-Definite matrices) plays a crucial role in image recognition. Bardes *et al*. [1] apply covariances for correlation and decorrelation among samples to avoid *collapse* (*i.e*. non-informative feature vectors). Faraki *et al*. [9] propose a cross-domain triplet loss using covariances for domain alignment. The projection matrices are also known to be effective to compute similarity between two entities as proposed in [31, 39]. In comparison, our proposed approach employs discriminative projection matrices for the learned features in a form of Mahalanobis metrics and bias terms to generate robust predictors.

3. Proposed Method

In this section, we present our approach to learning tasks sequentially with: **1**) constraints on the storage of the previously observed learning samples, and **2**) severe distribution shifts within the learning tasks, without suffering from the so-called issue of *catastrophic forgetting*. Our learning scheme identifies the feature and similarity metric learning jointly. In particular, we learn class-specific similarity metrics defined in the latent space to increase the discriminatory power of features in the space. This is done seamlessly along with learning the features themselves.

Below, we first review some basic concepts used in our framework. Our method tackles the domain generalization and catastrophic forgetting by incorporating two components: **1**) domain generalization strategy by learning Mahalanobis metrics and **2**) preserving past knowledge based on knowledge distillation using exponential moving average of parameters, which are discussed afterwards.

3.1. Notation and Preliminaries

Throughout the paper, we denote vectors and matrices in bold lower-case (*e.g*., $\boldsymbol{x}$) and bold upper-case letters (*e.g*. $\boldsymbol{X}$), respectively. On $\boldsymbol{x}$, $[\boldsymbol{x}]_i$ denotes the element at position i while $\|\boldsymbol{x}\|_2^2 = \boldsymbol{x}^\top \boldsymbol{x}$ shows its squared l_2 norm. We denote a set by $\mathcal{S}$.

Formally, in continual learning, a model is trained in several steps called *tasks*. Each task $T_i, 1 \leq i \leq q$, consists of samples of a set of novel classes $\mathcal{Y}_i^N$ as well as samples of a set of old classes $\mathcal{Y}_i^O$. The aim is to train a model to classify all seen classes, *i.e*., $\mathcal{Y}_i^O \cup \mathcal{Y}_i^N$. The allowed number of training samples for $\mathcal{Y}_i^O$ is severely constrained (called *rehearsal memory* $\mathcal{M}$).

In our cross-domain continual learning setup, we tackle the recognition scenario where during training we observe m source domains, *i.e*., $\mathcal{D}_1, \ldots, \mathcal{D}_m$, each with different distributions. The learning sequence is defined as learning through a stream of tasks $T_1, \ldots, T_q$, where the data from each task is composed of a sequence of m source domains. Note that, in our setup, we do not require the information about the domains (*e.g*., domain identifier) from which the samples in each task are given. When feeding the training data in each episode, we are interested in averaging the performance measures when the data domains are in random orders and the process is repeated for a number of times (*e.g*., 5). Like the standard continual learning setup, knowledge from a new set of classes is learned from each novel task. At the test time, we follow the domain generalization evaluation protocol in which the trained model has to predict $y \in \bigcup_{i=1}^{q} \mathcal{Y}_i$, values of inputs from an unseen/target domain $\mathcal{D}_{m+1}$. We note that $\mathcal{D}_{m+1}$ has samples from an unknown distribution. Our setting is presented conceptually in Fig. 2.

Like a standard continual learning method, we also apply experience replay by storing exemplars in the memory $\mathcal{M}$. To some extent, this would help preventing the forgetting issue. The exemplars stored in the memory are constructed from each class and each domain. We store randomly selected exemplars in the memory and ensure that every run uses this same set of exemplars. In the following, for simplicity, we drop the task indicator i and assume the size of label space is C.

3.2. Domain Generalization by Learning Similarity Metrics

In this part, we present our approach to learn class similarity metrics in a cross-domain continual learning setting, with the focus of generalization to unseen domains. To this end, we encourage learning semantically meaningful features by equipping the classifier with class similarity metrics which are obtained through Mahalanobis similarity computations. Here, we deem the domain alignment be done in a discriminative manner. In doing so, our idea is consistent with recent works that utilize a notion of feature semantics in their domain alignment inference to avoid undesirable effects of aligning semantically different samples from different domains. To name a few, the Contrastive Adaptation Network (CAN) for unsupervised domain adaptation [17], the Covariance Metric Networks (CovaMNet) for few-shot learning [22], the Model-Agnostic learning of Semantic Features (MASF) for standard domain generalization [6] and

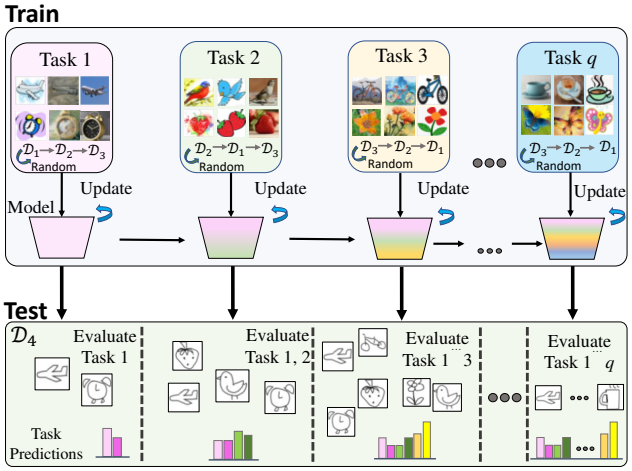


Figure 2. The overall setting in cross-domain continual learning. The training problem is divided into several tasks, where each new task has a subset of novel object categories coming from various training domains. While the training data from old tasks is discarded at each time, the model has to learn sequentially from the incoming tasks to evaluate on inputs from an unseen domain with a different distribution.

the Cross-Domain Triplet (CDT) loss for face recognition from unseen domains [9].

We begin by introducing the overall network architecture. Our architecture closely follows a typical image recognition design used in continual learning setting. Let $f_\theta : \mathcal{X} \rightarrow \mathcal{H}$ represents a backbone CNN parametrized by θ which provides a mapping from the image input space $\mathcal{X}$ to a latent space. Furthermore, let $f_\phi : \mathcal{H} \rightarrow \mathcal{Y}$ be a classifier network parametrized by ϕ that maps the outputs of f_θ to class label values. More specifically, forwarding an image x through $f_\theta(\cdot)$ outputs a tensor that, after being flattened (*i.e.*, $f_\theta(x) \in \mathbb{R}^n$) acts as input to the classifier network $f_\phi(\cdot)$. In a typical pipeline, the goal is to train a model on each task $T_i, 1 \leq i \leq q$, while expanding the output size of the classifier to match the number of classes. Note that the sequential learning protocol in our setting does not have strong priors and assumptions *e.g.*, domain identities and overlapping classes.

In most continual learning methods [16,23,30], the classifier network f_ϕ is often implemented by a Fully-Connected (FC) layer with weight $\boldsymbol{W} = [\boldsymbol{w}_1, \ldots, \boldsymbol{w}_C]^\top \in \mathbb{R}^{C \times n}$ with $\boldsymbol{w}_i \in \mathbb{R}^n$. When learning a new task, $\boldsymbol{W}$ is expanded to cover k new task categories by accommodating k new rows, *i.e.*, $\boldsymbol{W} = [\boldsymbol{w}_1, \ldots, \boldsymbol{w}_C, \boldsymbol{w}_{C+1}, \ldots, \boldsymbol{w}_{C+k}]^\top$. A similarity score between a class weight $\boldsymbol{w}_c$ and a feature $f_\theta(x) = \boldsymbol{h} \in \mathbb{R}^n$ associated with an image x is then defined by projection as $\langle \boldsymbol{w}_c, \boldsymbol{h} \rangle = \boldsymbol{w}_c^\top \boldsymbol{h}$ to be optimized by a loss function (see Fig. 3 (a)). Despite its wide use, we argue that this approach is not robust to distributional shifts as it is not explicitly designed to align samples that are seen in previous classes but from different distributions.

Here, we deem the classifier network also take into account the underlying distribution of the class samples when generating the class predictions. To this end, we equip the classifier network with Positive Semi-Definite (PSD) Mahalanobis similarity metrics $\boldsymbol{\Sigma}_c$ as learnable parameters, to encourage learning semantically meaningful features across different domains. Furthermore, category features are allowed to shift by learning a bias vector $\boldsymbol{b}_c$. We store these parameters in the network and expand to match the number of new classes when learning a new task. Therefore, after learning a new task, the prediction layer in our framework consists of extra learnable parameters $\phi = \{\boldsymbol{\Sigma}_1, \boldsymbol{b}_1, \ldots, \boldsymbol{\Sigma}_C, \boldsymbol{b}_C\}$. We then learn the backbone representation parameters θ along with ϕ in an end-to-end manner.

Utilizing ϕ, the proposed similarity score with respect to class c for an image x passing through the network can be obtained by

$$\mathrm{sim}_c(x; \theta, \phi) = (f_\theta(x) - \boldsymbol{b}_c)^\top \boldsymbol{\Sigma}_{\boldsymbol{c}} \, (f_\theta(x) - \boldsymbol{b}_c). \quad (1)$$

Intuition. The motivation behind Mahalanobis similarity learning is to determine $\boldsymbol{\Sigma}_c$ such that by learning to expand or shrink axes of $f_\theta(x) \in \mathbb{R}^n$, certain useful properties are achieved when generating (1). To better understand the behavior of our learning algorithm, let $\boldsymbol{r_c} = (f_\theta(x) - \boldsymbol{b}_c)$ and the eigendecomposition of $\boldsymbol{\Sigma}_c$ be $\boldsymbol{\Sigma}_c = \boldsymbol{V}_c \boldsymbol{\Lambda}_c \boldsymbol{V}_c^\top$. Then,

$$\begin{aligned} \boldsymbol{r_c}^\top \boldsymbol{\Sigma_c} \boldsymbol{r_c} &= \left(\boldsymbol{\Lambda_c}^{\frac{1}{2}} \boldsymbol{V_c}^\top \boldsymbol{r_c}\right)^\top \left(\boldsymbol{\Lambda_c}^{\frac{1}{2}} \boldsymbol{V_c}^\top \boldsymbol{r_c}\right) \\ &= \left\| \boldsymbol{\Lambda_c}^{\frac{1}{2}} \boldsymbol{V_c}^\top \boldsymbol{r_c} \right\|_2^2, \end{aligned} \quad (2)$$

which associates $\boldsymbol{r_c}$ with the eigenvectors of $\boldsymbol{\Sigma_c}$ weighted by the eigenvalues. When $\boldsymbol{r_c}$ is in the direction of leading eigenvectors of $\boldsymbol{\Sigma_c}$, it obtains its maximum value. Then, optimizing this term over the associated class samples leads to a more discriminative alignment of the data sources.

A computationally more efficient alternative. Taking advantage of the structure of $\boldsymbol{\Sigma}$, we can further decompose it to have a more efficient version. The similarity metric matrix can be decomposed to $\boldsymbol{\Sigma}_c = \boldsymbol{L}_c^\top \boldsymbol{L}_c$, where $\boldsymbol{L} \in \mathbb{R}^{r \times n}$ with $r \ll n$. This will ensure $\boldsymbol{\Sigma}$ remains PSD and yields valid similarity scores [7,8]. Furthermore, it can substantially reduce storage needs and increase the scalability of our method when a large-scale application is deemed. In practice, this lets us conveniently implement $\boldsymbol{\Sigma}$ by a FC layer into any neural network. Using the decomposition, (1) boils down to

$$\mathrm{sim}_c(x; \theta, \phi) = \left\| \boldsymbol{L}_c \big(f_\theta(x) - \boldsymbol{b}_c\big) \right\|_2^2. \quad (3)$$

Overall training pipeline. Finally, the updated classifier parameters become $\phi = \{\boldsymbol{L}_1, \boldsymbol{b}_1, \ldots, \boldsymbol{L}_C, \boldsymbol{b}_C\}$. Later, in

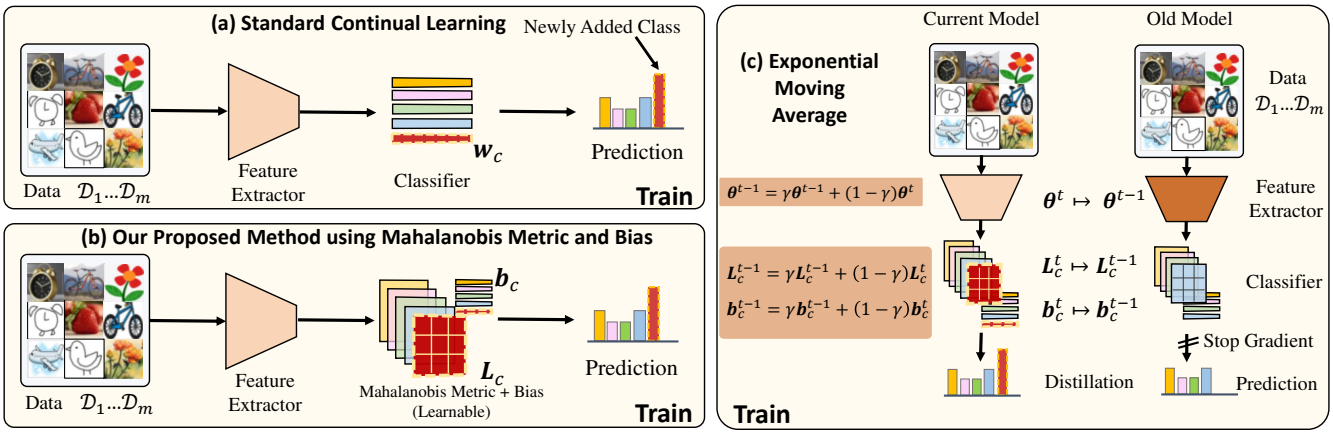


Figure 3. The pipeline of our approach. (a) For comparison, we show a standard continual learning approach with expanding parameters when a new class is presented. (b) Our approach also expands the classifier with Mahalanobis metrics and biases as learnable parameters to learn semantically meaningful features across training domains. (c) Extension of our proposed domain generalization method with knowledge distillation to allow smooth updates when learning new tasks.

experiments, we will study the effect of different values of r in our framework. We store some examples from seen tasks and various domains. During training, the samples x in mini-batches come from the samples in the current task and the memory. Thus, our objective becomes minimizing the loss function across domains and samples. Our parameters ϕ that represents each class are updated during training in conjunction with the feature extractor parameters θ. We train our model using the cross-entropy loss, which is widely used for Empirical Risk Minimization (ERM) [33]

$$\mathcal{L}_{\text{CE}} = -\sum_{x \in \mathcal{X} \cup \mathcal{M}} \delta_{y=c} \log \frac{\exp\big(\text{sim}_c(x; \theta, \boldsymbol{L}_c, \boldsymbol{b}_c)\big)}{\sum_{c'} \exp\big(\text{sim}_{c'}(x; \theta, \boldsymbol{L}_{c'}, \boldsymbol{b}_{c'})\big)}, \tag{4}$$

where δ is an indicator function corresponding with the label y.

As mentioned earlier, we have a memory of exemplars from various domains. Thus, the learnable parameters can be updated towards a more generalized classifier as an attempt to improve classification on unseen domains. We conceptually show our proposed method for Mahalanobis Similarity Learning (MSL) in Fig. 3 (b).

3.3. Knowledge Distillation with Exponential Moving Average

In this section, we develop an effective Knowledge Distillation (KD) strategy to take advantage of previously learned knowledge without requiring old tasks' images and labels. While many other methods focus on applying KD [10] using only the old and current models [3, 16, 23, 30], we utilize a history of previous knowledge to limit the divergence from the old model. Let $\Psi^t = \{\theta^t, \phi^t\}$ and $\Psi^{t-1} = \{\theta^{t-1}, \phi^{t-1}\}$ be all learnable parameters in our framework at the current and old tasks, respectively. Then, given a temperature τ, we propose the following KD on the predictions of the current and old models

$$\mathcal{L}_{\text{Dis}}(\Psi^t, \Psi^{t-1}; x) = -\sum_{c=1}^{C} p_c^{t-1}(x) \log p_c^t(x), \tag{5}$$

with

$$p_c^{t-1}(x) = \frac{\exp\big(\text{sim}_c(x; \Psi^{t-1})/\tau\big)}{\sum_{c'=1}^{C} \exp\big(\text{sim}_{c'}(x; \Psi^{t-1})/\tau\big)}, \; p_c^t(x) = \frac{\exp\big(\text{sim}_c(x; \Psi^t)/\tau\big)}{\sum_{c'=1}^{C} \exp\big(\text{sim}_{c'}(x; \Psi^t)/\tau\big)},$$

where the similarity score, $\text{sim}(\cdot)$, is obtained by (3).

As also empirically observed in [10], temporal ensembling methods applied to the old model stabilizes the training. Here, the idea is that the outputs of the current model must not significantly deviate form the old ones. To this end, we employ a smooth parameter update strategy using Exponential Moving Average (EMA) updates. Connecting to KD, the idea is to smoothly guide the learning of the parameters of the current model while taking into account the predictions from the old ones. Therefore, we define the EMA update in our framework as

$$\begin{aligned} \theta^{t-1} &= \gamma\theta^{t-1} + (1-\gamma)\theta^t, \\ \boldsymbol{b}_c^{t-1} &= \gamma\boldsymbol{b}_c^{t-1} + (1-\gamma)\boldsymbol{b}_c^t, \\ \boldsymbol{L}_c^{t-1} &= \gamma\boldsymbol{L}_c^{t-1} + (1-\gamma)\boldsymbol{L}_c^t, \end{aligned} \tag{6}$$

where γ is a positive smoothing coefficient hyperparameter.

Furthermore, we apply a stop-gradient operator to the old model. Once the training is done, the old model is discarded. The process is depicted in Fig. 3 (c). Our overall loss then becomes $\mathcal{L}_{\text{CE}} + \lambda\mathcal{L}_{\text{Dis}}$, with λ showing the weight of KD loss. We dub this method MSL + Mov in our experiments.

4. Experiments

In this section, we compare and contrast our method against existing methods in both (class) Continual Learning (CL) and Domain Generalization (DG). We start with introducing our competitor methods and experimental details.

Baselines. To evaluate our proposed method, we compare with the competitor methods in CL, namely LwF [23], LUCIR [16], and GeoDL [30]. Concisely, LwF [23] applies knowledge distillation on the predictions, while LUCIR [16] employs preservation of features from old and current models, and GeoDL [30] extends the preservation in the feature space with smooth transitions of two subspaces from the old and the current models. These models are a combination of widely used and very recent methods. Furthermore, we include a baseline Empirical Risk Minimization (ERM) [33] as well as recent DG methods – MixStyle [40] and Adaptive Risk Minimization (ARM) [38]. MixStyle [40] makes use of style transfer, interpolating means and standard deviations in a normalization layer with inputs in a mini-batch coming from different domains. Additionally, to handle distribution shifts at test time, ARM [38] uses a contextual network that utilizes extra domain information via a meta-learning strategy. For fair comparison, we adopt all baselines to our setup without substantial modification.

Datasets. In our experiments, we use popular DG benchmarks, namely DomainNet [29], OfficeHome [34], PACS [21] and NICO [14]. These datasets are ideal candidates for training and evaluating domain generalizable CL methods with multiple domains and a large number of classes. Specifically, DomainNet is a large-scale dataset containing images of 126 classes from 4 domains: Real, Clipart, Painting and Sketch. OfficeHome is another large-scale benchmark that contains 15K images spanning a total of 65 classes in 4 domains: Real, Clipart, Art and Product. We also consider the PACS dataset which has images from 4 domains: Art, Cartoon, Photo and Sketch. PACS provides challenging recognition scenarios with large domain shifts as described in [21]. Finally, we evaluate on the NICO dataset that has multiple domains called contexts. We consider four domains (Eating, Ground, Water and Grass) from the animal split of the dataset since only these domains contain all classes. Comparatively, we consider smaller task experiments on the PACS and NICO-Animal datasets since the number of categories is limited.

Implementation details. For all datasets, we follow the provided splits for training and testing. Furthermore, the images are resized to 224×224. We adopt three cross-domain CL protocols, which consist of 2, 5, and 10 tasks. In our experiments, we exclude one domain for evaluation and consider the remaining ones for training, *e.g.*, for DomainNet, we hold the Clip domain for testing while training on Paint, Real and Sketch samples. Following the common practice in CL, some exemplars are also stored in the memory to replay in future iterations. The memory sizes in our experiments are set to 10 for DomainNet and 5 for all other datasets. Note that exemplar selection strategy is not the main focus in this work. Thus, we opt to use random selection and replay the same images in the memory for all methods in our experiments. We train a model for 200 epochs with standard data augmentations (*e.g.* flipping, cropping and color jittering) by using the SGD optimizer with the learning rates of $1e{-}4$ for DomainNet and $2e{-}5$ for other datasets. We use a ResNet-34 model pretrained on the ImageNet as our backbone network. As for the distillation loss, we experimentally observed that setting the hyperparameter λ to $1e{-}3$, $1e{-}3$, $1e{-}2$, $1e{-}3$ works well for LwF [23], LUCIR [16], GeoDL [30] and our methods, respectively. The exponential moving average hyperparameter in our method is set to $\gamma = 0.96$. As suggested in [15, 23], we set $\tau = 2$ to achieve softer probabilities among classes. Finally, we found a maximum rank of $r = 64$ for the Mahalanobis metric matrices to work well across all protocols and datasets.

Evaluation measurements. We assess the baselines and our method for cross-domain CL using two important measurements. The average accuracy of all tasks is considered to evaluate the model capability when continually learning new tasks. Another measurement is the ability to transfer backward from new tasks to old tasks, which is related to the forgetting rate in cross-domain CL. We follow the backward transfer formulation proposed in [25], where $\mathcal{A}_t$ is the accuracy on task t (*i.e.*, where $y \in \bigcup_{i=1}^{t} \mathcal{Y}_i$ from domains $\mathcal{D}_1, \ldots, \mathcal{D}_m$). Let $\mathcal{A}_t|_{\mathbf{\Psi}_j}$ be the accuracy for task t evaluated using a model trained from task 1 to j, where $j \leq t$. Then the average accuracy and backward transfer are defined as

$$\mathcal{A} = \frac{1}{q}\sum_{t=1}^{q} \mathcal{A}_t, \quad \mathcal{BW} = \frac{1}{q}\sum_{t=1}^{q} \mathcal{A}_t|_{\mathbf{\Psi}_t} - \mathcal{A}_t|_{\mathbf{\Psi}_q} \,. \tag{7}$$

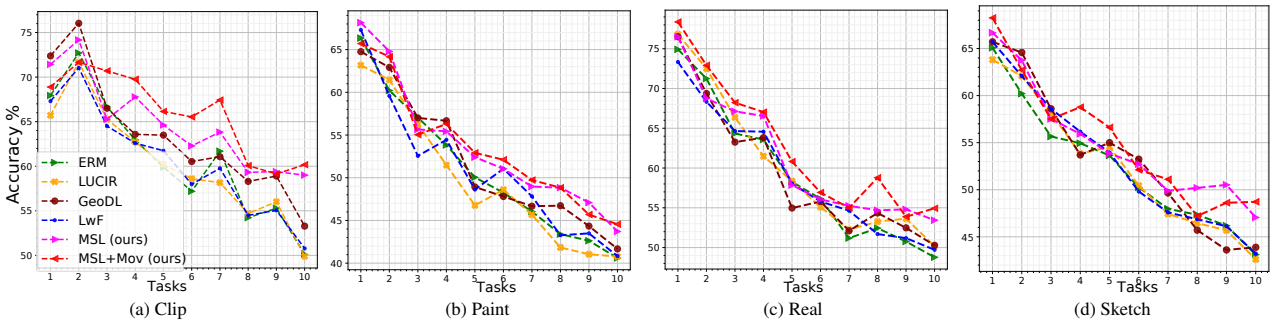
A better model is identified with a larger value of the average accuracy and a lower value of the backward transfer rate.

4.1. Supervised Cross-Domain Continual Learning

We evaluate our cross-domain CL methods in supervised fashions when available classes from a benchmark are split into 5-tasks and 10-tasks. As shown in Table 1, each column corresponds to the performances when samples of a single domain are entirely excluded (considered unseen) from the training. In addition, we report results of the accuracy numbers on the seen domains in our supplementary material. As can be seen in the Table, both our methods with Mahalanobis metrics and biases (MSL and MSL + Mov) comfortably outperform all the competitors in CL and DG. Here, MSL with knowledge distillation and exponential moving average (MSL + Mov) improves over MSL and achieves

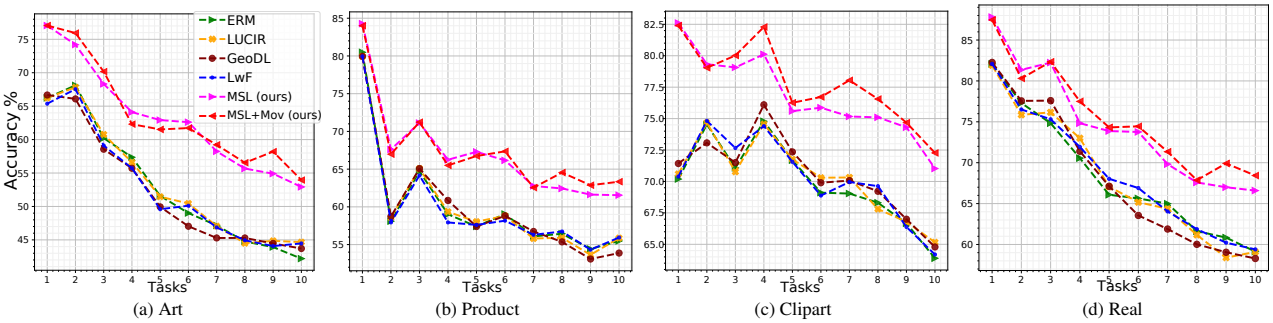
Method	DomainNet								OfficeHome							
	10-Task Acc. (% ↑)				5-Tasks Acc. (% ↑)				10-Task Acc. (% ↑)				5-Tasks Acc. (% ↑)			
	Clip	Paint	Real	Sketch	Clip	Paint	Real	Sketch	Art	Product	Clipart	Real	Art	Product	Clipart	Real
ERM [33]	60.0	51.4	60.3	53.1	59.7	50.2	57.7	51.7	48.8	52.3	64.7	62.4	49.7	51.6	64.9	61.9
LwF [23]	61.3	51.9	60.0	53.5	62.2	52.1	62.6	54.9	49.4	53.8	65.2	63.2	49.9	51.3	67.5	63.1
LUCIR [16]	61.1	52.1	59.7	53.0	61.3	52.7	61.1	55.4	49.3	53.6	65.7	62.3	49.7	51.6	67.5	64.9
GeoDL [30]	61.0	50.5	58.5	54.1	62.1	52.8	61.1	55.5	50.6	53.0	67.1	63.1	50.5	52.4	67.4	64.2
ARM [38]	57.0	49.3	62.3	51.2	55.4	51.8	60.2	47.7	39.8	55.0	54.3	51.7	43.6	56.3	54.5	55.4
MixStyle [40]	58.0	51.4	59.5	52.5	59.6	48.5	56.0	53.5	47.3	54.9	56.3	56.0	48.9	56.9	57.7	59.8
MixStyle + LUCIR	62.4	50.0	59.5	52.8	58.2	47.4	54.8	51.8	51.3	52.2	65.1	62.0	49.3	49.5	65.5	63.4
MSL (ours)	63.2	51.3	61.8	55.6	63.3	**55.4**	63.6	57.4	**61.6**	61.4	71.7	72.7	54.3	**63.6**	68.0	67.3
MSL + Mov (ours)	**63.7**	**55.0**	**63.1**	**56.4**	**63.8**	55.3	**64.6**	**58.3**	61.2	**63.0**	**75.3**	**73.1**	**57.9**	60.2	**71.4**	**70.9**

Table 1. Cross-domain continual learning average accuracy numbers for unseen domains with 10-tasks and 5-tasks protocols on the DomainNet [29] and OfficeHome [34] datasets.



(a) Clip (b) Paint (c) Real (d) Sketch

Figure 4. The average accuracy numbers of tasks on the unseen domains (Clip, Paint, Real and Sketch) of the DomainNet dataset [29] using the 10-tasks protocol.



(a) Art (b) Product (c) Clipart (d) Real

Figure 5. The average accuracy numbers of tasks on the unseen domains (Art, Product, Clipart and Real) of the OfficeHome dataset [34] using the 10-tasks protocol.

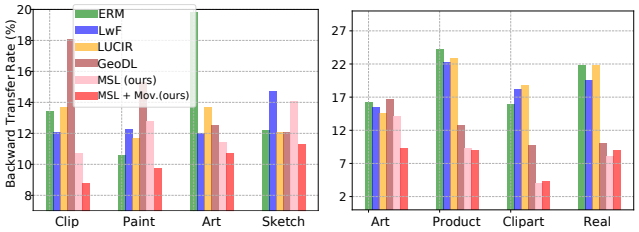
the highest accuracy. To mention one instance, when generalizing to DomainNet-Sketch as an unseen domain, our MSL + Mov obtains 56.4%, which is 2.3% higher compared to GeoDL's performance of 54.1% for the 10-tasks protocol. This improvement trend also appears on the seen domains.

Furthermore, in Fig. 4 and Fig. 5, we show the average classification accuracy numbers of tasks obtained by our methods in comparison to prior CL methods on the DomainNet and OfficeHome datasets for the 10-tasks protocol. Overall, our methods outperform the baselines, with more significant performance gaps on the OfficeHome dataset. In addition, we evaluate on the PACS and the NICO-Animal datasets. The results are shown in Table 2. Here, the average accuracy of our methods is superior to other competitors by at least 2% margin. MSL + Mov clearly shows the benefits of knowledge distillation to a history of models, with a gap of 2.1% over MSL in the best case.

Overall, we observe that standard CL algorithms largely fail to prevent catastrophic forgetting in the cross-domain setting, indicated by high backward transfer rates shown in

Method	NICO-Animal (% ↑)				PACS (% ↑)			
	Eating	Ground	Water	Grass	Art	Cartoon	Photo	Sketch
ERM [33]	88.0	86.5	82.3	84.3	76.3	82.9	84.7	61.9
LwF [23]	88.2	86.2	83.3	84.3	76.4	82.4	85.5	62.4
LUCIR [16]	88.1	86.6	83.3	84.3	76.5	82.1	84.2	62.2
GeoDL [30]	87.9	86.1	82.5	83.3	72.8	83.6	85.4	60.5
ARM [38]	86.2	83.5	80.8	83.5	65.1	83.3	84.9	64.9
MixStyle [40]	86.1	84.0	81.0	83.4	73.3	82.6	81.1	63.5
MixStyle + LUCIR	88.0	82.6	81.9	83.0	70.3	82.7	83.6	63.6
MSL (ours)	89.9	86.2	84.4	85.2	**77.3**	82.1	87.0	62.8
MSL + Mov (ours)	**91.3**	**87.9**	**85.0**	**87.2**	77.2	**84.1**	**89.0**	**64.9**

Table 2. Domain generalization test for 2-tasks supervised learning on NICO-Animal [14] and PACS [21] datasets.



(a) DomainNet (% ↓) (b) OfficeHome (% ↓)

Figure 6. The backward transfer performance (absolute values) on DomainNet [29] and OfficeHome [34] datasets (the lower the better) for the 10-tasks protocol.

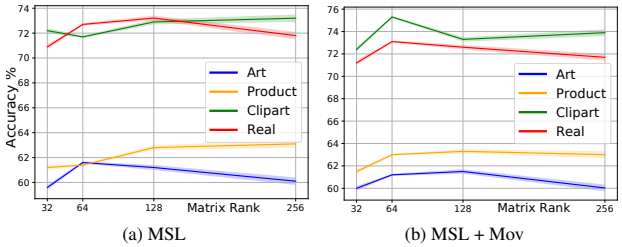
Fig. 6. In contrast, on average, our method with exponential moving average, MSL + Mov, can achieve the lowest backward transfer rate of 10.1% on the DomainNet and 7.8% on the OfficeHome datasets for the 10-tasks protocol. The closest competitors to ours report 12.8% (by LUCIR) and 12.3% (by GeoDL), respectively. Note that the method with the lower backward transfer rate is better.

4.2. Ablation Studies and Analyses

We investigate how hyperparameters impact the performance of our proposal. Below, we show the impact of matrix rank for the Mahalanobis similarity learning and how the memory size affects the performance.

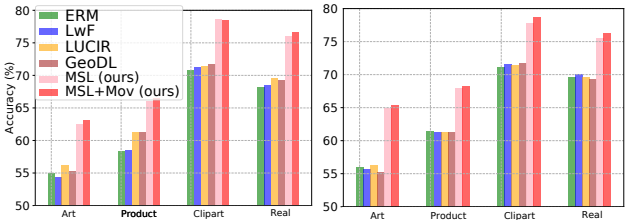
Impact of varying the maximum rank. Our approach employs a low-rank strategy for the metric matrices. We investigate the performances when four different r values (*e.g.*, 32, 64, 128 and 256) are used. We observe in Fig. 7 that varying r would change the accuracy by less than 1.5%. The plots also show that setting a large value for r, which translates to having more parameters, does not directly increase the performance. As a rule of thumb, the matrix maximum rank value that we use for all experiments is set to 64.

Impact of varying the memory size. Below, we investigate how increasing the memory size impacts the average accuracy over the 10-tasks protocol with 10 and 20 exemplars. Fig. 8 shows that having more exemplars in the memory



(a) MSL (b) MSL + Mov

Figure 7. The impact of varying r in learning the Mahalanobis metric matrices on the average accuracy using the unseen domains of the Officehome dataset [34] and the 10-tasks protocol.



(a) 10 memory size (% ↑) (b) 20 memory size (% ↑)

Figure 8. The impact of varying memory size on the average accuracy using the unseen domains of the Officehome dataset [34] and the 10-tasks protocol.

attains higher accuracy for all methods. Our proposed approach can still benefit from more exemplars and outperform baselines. We observe that our method leads by over 5% for both 10 and 20 exemplar sizes in the memory.

5. Limitations and Conclusions

Our setup and method can be limited from two aspects. First, like many other continual learning setups, one underlying assumption of our setup is that some exemplars are stored for replay. This might restrict its use in some applications with strict privacy regulations. Second, the number of parameters in our methods grow linearly following the number of tasks. Though, we have proposed a memory efficient alternative that scales well to many applications, this might still limit the practical use when very large-scale applications with severe memory constraints are desired.

We propose an approach to generalize across training domains while mitigating the so-called *catastrophic forgetting* issue via Mahalanobis similarity learning and knowledge distillation with exponential moving average update. In our evaluation, we follow the so-called leave-one-domain-out protocol where a test domain is not seen during training. As our experimental evaluations indicate, our methods comfortably outperform the existing methods in both class continual learning and domain generalization on challenging datasets, namely DomainNet, OfficeHome, PACS and NICO-Animal. The ablation studies also show that our method consistently improves over the baselines in various conditions and has low sensitivity to the choice of hyperparameters.

References

[1] Adrien Bardes, Jean Ponce, and Yann LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. *arXiv preprint arXiv:2105.04906*, 2021. 3

[2] Eden Belouadah and Adrian Popescu. Il2m: Class incremental learning with dual memory. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 583–592, 2019. 1

[3] Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. Dark experience for general continual learning: a strong, simple baseline, 2020. 2, 5

[4] Francisco M Castro, Manuel J Marín-Jiménez, Nicolás Guil, Cordelia Schmid, and Karteek Alahari. End-to-end incremental learning. In *Proceedings of the European conference on computer vision (ECCV)*, pages 233–248, 2018. 1

[5] Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with a-gem. In *ICLR*, 2019. 2

[6] Qi Dou, Daniel Coelho de Castro, Konstantinos Kamnitsas, and Ben Glocker. Domain generalization via model-agnostic learning of semantic features. In *NIPS*, pages 6450–6461, 2019. 3

[7] Masoud Faraki, Mehrtash T Harandi, and Fatih Porikli. Large-scale metric learning: A voyage from shallow to deep. *IEEE transactions on neural networks and learning systems*, 29(9):4339–4346, 2017. 4

[8] Masoud Faraki, Mehrtash T Harandi, and Fatih Porikli. A comprehensive look at coding techniques on riemannian manifolds. *IEEE transactions on neural networks and learning systems*, 29(11):5701–5712, 2018. 4

[9] Masoud Faraki, Xiang Yu, Yi-Hsuan Tsai, Yumin Suh, and Manmohan Chandraker. Cross-domain similarity learning for face recognition in unseen domains. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15292–15301, 2021. 3, 4

[10] Geoffrey French, Michal Mackiewicz, and Mark Fisher. Self-ensembling for visual domain adaptation. In *ICLR*, 2018. 5

[11] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 1180–1189, 2015. 3

[12] Muhammad Ghifary, David Balduzzi, W. Kleijn, and Mengjie Zhang. Scatter component analysis: A unified framework for domain adaptation and domain generalization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39:1414–1430, 2017. 1

[13] Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. In *International Conference on Learning Representations*, 2021. 1, 3

[14] Yue He, Zheyan Shen, and Peng Cui. Towards non-iid image classification: A dataset and baselines. *Pattern Recognition*, page 107383, 2020. 2, 6, 8

[15] Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. In *NIPS Deep Learning and Representation Learning Workshop*, 2015. 2, 6

[16] Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Learning a unified classifier incrementally via rebalancing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2, 4, 5, 6, 7, 8

[17] Guoliang Kang, Lu Jiang, Yi Yang, and Alexander G Hauptmann. Contrastive adaptation network for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4893–4902, 2019. 3

[18] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017. 1

[19] Jogendra Nath Kundu, Rahul Mysore Venkatesh, Naveen Venkat, Ambareesh Revanur, and R Venkatesh Babu. Class-incremental domain adaptation. In *ECCV*, pages 53–69, 2020. 2

[20] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy Hospedales. Learning to generalize: Meta-learning for domain generalization. In *AAAI Conference on Artificial Intelligence*, 2018. 1

[21] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M. Hospedales. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017. 1, 2, 6, 8

[22] Wenbin Li, Jinglin Xu, Jing Huo, Lei Wang, Yang Gao, and Jiebo Luo. Distribution consistency based covariance metric networks for few-shot learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8642–8649, 2019. 3

[23] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947, 2017. 2, 4, 5, 6, 7, 8

[24] Yaoyao Liu, Yuting Su, An-An Liu, Bernt Schiele, and Qianru Sun. Mnemonics training: Multi-class incremental learning without forgetting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12245–12254, 2020. 2

[25] David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. In *NIPS*, 2017. 1, 2, 6

[26] Marc Masana, Xialei Liu, Bartlomiej Twardowski, Mikel Menta, Andrew D. Bagdanov, and Joost van de Weijer. Class-incremental learning: survey and performance evaluation. *CoRR*, 2020. 1

[27] 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 International Conference on Computer Vision*, pages 1406–1415, 2019. 2

[28] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and

representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 2001–2010, 2017. 1

[29] Kuniaki Saito, Donghyun Kim, Stan Sclaroff, Trevor Darrell, and Kate Saenko. Semi-supervised domain adaptation via minimax entropy. In *ICCV*, 2019. 6, 7, 8

[30] Christian Simon, Piotr Koniusz, and Mehrtash Harandi. On learning the geodesic path for incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1591–1600, 2021. 2, 4, 5, 6, 7, 8

[31] Christian Simon, Piotr Koniusz, Richard Nock, and Mehrtash Harandi. Adaptive subspaces for few-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4136–4145, 2020. 3

[32] Gido M Van de Ven and Andreas S Tolias. Three scenarios for continual learning. *arXiv preprint arXiv:1904.07734*, 2019. 1

[33] Vladimir N. Vapnik. *Statistical Learning Theory Wiley*. 1998. 3, 5, 6, 7, 8

[34] 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. 2, 6, 7, 8

[35] Riccardo Volpi, Diane Larlus, and Grégory Rogez. Continual adaptation of visual representations via domain randomization and meta-learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4443–4453, 2021. 2

[36] Max Welling. Herding dynamic weights for partially observed random field models. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pages 599–606, 2009. 2

[37] Garrett Wilson and Diane J. Cook. Adversarial transfer learning. *CoRR*, abs/1812.02849, 2018. 3

[38] Marvin Zhang, Henrik Marklund, Abhishek Gupta, Sergey Levine, and Chelsea Finn. Adaptive risk minimization: A meta-learning approach for tackling group shift. *CoRR*, abs/2007.02931, 2020. 2, 3, 6, 7, 8

[39] An Zhao, Mingyu Ding, Jiechao Guan, Zhiwu Lu, Tao Xiang, and Ji-Rong Wen. Domain-invariant projection learning for zero-shot recognition. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, page 1027–1038, 2018. 3

[40] Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. Domain generalization with mixstyle. In *International Conference on Learning Representations*, 2021. 2, 3, 6, 7, 8