Weight Averaging Improves Knowledge Distillation under Domain Shift

Valeriy Berezovskiy
HSE University
vsberezovskiy@edu.hse.ru

Nikita Morozov
HSE University
nvmorozov@hse.ru

Abstract

Knowledge distillation (KD) is a powerful model compression technique broadly used in practical deep learning applications. It is focused on training a small student network to mimic a larger teacher network. While it is widely known that KD can offer an improvement to student generalization in i.i.d setting, its performance under domain shift, i.e. the performance of student networks on data from domains unseen during training, has received little attention in the literature. In this paper we make a step towards bridging the research fields of knowledge distillation and domain generalization. We show that weight averaging techniques proposed in domain generalization literature, such as SWAD and SMA, also improve the performance of knowledge distillation under domain shift. In addition, we propose a simplistic weight averaging strategy that does not require evaluation on validation data during training and show that it performs on par with SWAD and SMA when applied to KD. We name our final distillation approach Weight-Averaged Knowledge Distillation (WAKD).

1. Introduction

Large scale vision models over the years have shown impressive results in various computer vision tasks [12, 34, 16, 9, 22]. However, the use of such models in practice is often limited by available computational resources. Knowledge distillation [13] is a popular compression technique that can be used to improve both memory and computational time efficiency of large neural networks or neural network ensembles. KD transfers the knowledge from a cumbersome teacher model to a more lightweight student model by training the student to match the teacher behavior on the available data instead of predicting the true labels.

Domain generalization (DG) [5] focuses on another challenge that often arises in machine learning applications: domain shift. How well do our models perform when training and test data distributions differ? An example can be a vision model encountering a weather condition or a time of day unseen during training. In DG benchmarks, models are trained on a number of source domains and their performance is evaluated on a different target domain that is unavailable during training to emulate the domain shift setting.

The focus of this paper lies at the intersection of these two fields. How well does the teacher's ability to generalize to unseen domains is transferred to the student during KD? This question naturally arises in the situation when we have a large model that performs well under domain shift and want to compress it into a smaller model, preserving the ability to generalize to unseen domains. Thus the scenario of our interest is when both the teacher training and the distillation are done on source domains, while the performance metric is calculated on the target domain. Despite the large amount of existing research on KD and DG, this setting has received little attention in the existing literature.

In this work, we apply weight averaging techniques to improve knowledge distillation performance under distribution shift. Averaging model weights was proposed to improve neural network training in SWA [14], where weights of multiple models from a single training trajectory are averaged to construct a better generalizing model (not to be confused with ensembling, where predictions of multiple models are averaged). A number of SWA modifications were created and successfully applied to domain generalization, such as SWAD [8] and SMA [2], showing that weight averaging improves out-of-distribution generalization as well. We apply SWAD and SMA to knowledge distillation, showing that they improve KD performance across two domain generalization datasets (PACS [18] and Office-Home [38]) and two teacher-student neural network architectures (ResNet and ViT). In addition, we introduce a simplistic weight averaging strategy by modifying SMA to include all the network weights up to the end of the training trajectory and show that it performs on par with its counterparts. The advantage of this strategy over SWAD and SMA is that it does not need to calculate the validation set performance of the student to choose the averaging segment, thus reducing the computational cost of the whole procedure. We coin such knowledge distillation approach Weight-Averaged Knowledge Distillation (WAKD), constructing a

simple yet powerful baseline for future research of knowledge distillation under domain shift. Our source code is available at .

2. Related Work

Domain Generalization: The problem of domain generalization was formally introduced in [5]. Over the years, a plethora of domain generalization methods have been proposed in the literature, including techniques based on domain alignment [24, 23, 20], meta learning [19, 3, 49], self-supervised learning [7, 40, 1, 6] and data augmentation [39, 30, 43]. See [50] for a comprehensive survey. An important work in the field [11] showed that model selection can be crucial, and when correctly taken into account during comparison of the approaches, empirical risk minimization (ERM [37]) baseline showed strong performance in comparison to DG approaches of that time. The main focus of our work is on approaches based on weight averaging. SWAD [8] argues that finding flat minima leads to better performance of the model under domain shift and proposes a dense weight averaging strategy for this purpose. One drawback of SWAD is the fact that it introduces three additional hyperparameters to the training process that are used to find the segment of weights for averaging. This drawback is answered in [2], where, in addition to other contributions, authors present SMA: an alternative hyperparameter-free weight averaging strategy.

Knowledge Distillation: Knowledge distillation was proposed in [13] as a model compression technique. The original method was to minimize KL-divergence between the teacher's and the student's predictive distributions, thus training the student to match the predictions of the teacher. A number of variations and modifications of the approach were proposed after the original work [28, 45, 35, 47, 42, 48]. In contrast to developing new distillation objectives, [4] showed that the original approach remains a powerful and competitive method in computer vision tasks when taking several design choices into account, such as longer training and passing the same augmented images to the teacher and the student.

There are several works that employ ideas from KD literature to construct novel methods for domain generalization [41, 17, 32, 31], however these works do not use knowledge distillation itself as a model compression technique. The authors of [10] consider a setting that resembles ours, with the difference that the goal of the student is to match the teacher's performance on the domain where the teacher was trained, without access to any data from this domain. Among the only works that study the same setting as we do, [51] shows that data augmentation techniques such as CutMix [44] and Mixup [46] improve KD performance under domain shift. The authors of [25] observe high agree-ment between the predictions of the teacher and the student on domains unseen by both of them, which can indicate that KD does in fact transfer knowledge about unseen domains. In another related line of work, [21] studies out-of-distribution generalization when distilling vision-language models (e.g. CLIP [27]).

3. Methodology

3.1. Domain Generalization

The formal setting of domain generalization is as follows. Let $\mathcal{X}$ be the input space and $\mathcal{Y}$ the target space. We have access to data from K source domains S_k, each associated with a joint distribution P_{XY}^k on pairs from $\mathcal{X} \times \mathcal{Y}$. The goal of DG is to learn a predictive model $f : \mathcal{X} \rightarrow \mathcal{Y}$ using only the source domains data such that the prediction error on an unseen target domain T is minimized. The corresponding target domain joint distribution is denoted as P_{XY}^T, and it is assumed that $P_{XY}^T \neq P_{XY}^k, \forall k \in \{1, \ldots, K\}$.

DG datasets generally contain data from multiple labeled domains. To evaluate the performance of a training approach, all possible splits are considered where one domain is chosen as target and the other ones are chosen as source.

It is important to note that the standard DG setting forbids any use of target domain data during both training and model selection, meaning that both training and validation data come from source domains. However, [11] showed that picking a model with the best source validation data performance is a viable strategy for model selection in DG.

3.2. Knowledge Distillation

In our experiments, we consider image classification problems and employ the broadly used and well-studied distillation objective proposed in the original work [13]: KL-divergence between the teacher's and the student's predicted class probability vectors.

Let $\mathbf{z}_t$ and $\mathbf{z}_s$ be the output class logits predicted by the teacher and the student networks respectively on the same data point. Then the distillation loss is

$$\mathcal{L}_{\mathrm{KD}}\left(\mathbf{z}_s, \mathbf{z}_t\right) := -\tau^2 \sum_{j=1}^{C} \sigma_j\left(\frac{\mathbf{z}_t}{\tau}\right) \log \sigma_j\left(\frac{\mathbf{z}_s}{\tau}\right), \quad (1)$$

where $\sigma_i(\mathbf{z}) := \exp\left(z_i\right) / \sum_j \exp\left(z_j\right)$ is the softmax function and C is the number of classes. The temperature parameter τ is introduced to adjust the entropy of the predicted softmax-probability distributions before they are used in the loss computation. The presented loss does not contain the entropy of the teacher since it does not depend on the predictions of the student.

During distillation, the weights of the teacher network are frozen, while the student network minimizes $\mathcal{L}_{\mathrm{KD}}$, thus

learning to match the outputs of the teacher rather than predicting true hard labels. The whole pipeline in our study consists of training the teacher network on the set of available source domains and then distilling it into the student network on the same data. The metric of our interest is the accuracy of the student network on the target domain. To be more precise, it is the improvement of the target domain accuracy when using the distillation objective in comparison to training the same network to predict hard labels.

3.3. Weight Averaging Strategies

Weight averaging approaches follow the idea of picking a subset of models from a training trajectory instead of a single model and averaging their weights to create a final better generalizing model.

SWAD evaluates the current model during training with some frequency, picking a segment of training iterations for weight averaging based on validation loss values of the evaluated models. The start of the segment is chosen to be the first iteration where the loss value is no longer decreased during N_s iterations. The end of the segment is chosen to be the first iteration where the loss value exceeds the loss at the beginning of the segment time tolerance r during N_e iterations. Then, all models for every training iteration inside the segment are averaged. N_s, N_e, and r are hyperparameters of the method. We refer the reader to [8] for a more detailed description of the algorithm.

SMA offers a simpler alternative to SWAD. It also averages all models inside some segment of training iterations, but always takes the start of the segment to be some fixed iteration close to the beginning of training. Validation accuracy is computed for models already averaged from the beginning of the segment up to the current iteration instead of individual models. The end of the segment is taken to maximize the validation accuracy of the averaged model.

In the next section, we perform knowledge distillation with SWAD and SMA, showing that they produce students with better target domain accuracy in comparison to simply picking an individual model with the best validation performance from a training trajectory.

Observing that SWAD with distillation loss often behaves in such a way that the end of the segment is chosen to be close to the end of training, we offer an even simpler strategy in comparison to both. We start averaging from some fixed iteration close to the beginning of training (10% in our experiments) akin to SMA, but average all the weights up to the end of a training trajectory. The advantage of this strategy over SWAD and SMA is that it does not require any model validation during training. **Thus our proposed WAKD procedure consists of running a regular knowledge distillation and then averaging weights of all models from the training trajectory except for the first 10%.**

4. Experiments

4.1. Experimental Details

We carry out the experimental evaluation on two domain generalization datasets: PACS [18] (9,991 images, 7 classes, and 4 domains) and OfficeHome [38] (15,588 images, 65 classes, and 4 domains).

We explore two teacher-student architectures: ResNet-50 to ResNet-18 [12] and DeiT-Small to DeiT-Tiny [36]. It is important to note that the authors of DeiT presented their own novel distillation approach [36], while we use the original KD objective (see Eq. 1) to create a unified experimental setup.

There is no particular necessity in using the best possible performing teacher in our experimental setting since the interest lies mainly in the performance of KD, thus we chose SWAD to train the teachers as an approach well-established in domain generalization literature.

We train teacher models using SWAD for 5,000 iterations (the same amount as in SWAD paper), as well as baselines for student models, where a model of student architecture is trained to predict hard labels independently of the teacher using SWAD. Following observations from [4], we increase the amount to 50,000 iterations for all distillation experiments. For both SWAD and SMA, we use the same validation frequency of 100 iterations for a fair comparison. For SMA and WAKD, we fix the starting iteration for averaging at 5,000 (10% of the training duration). All models are initialized with ImageNet-1k [29] pre-trained weights at the beginning of the training.

In addition to weight-averaged distillation students, we present the performance of a single model from the iteration with the best validation accuracy (denoted as ERM in the results), acting as a baseline for weight averaging strategies.

All models are trained using Adam [15] with learning rate of 5e-5 and batch size 64. We set temperature parameter $\tau = 5$ for KD. For SWAD we use standard hyperparameter values from the paper: $N_s = 3$, $N_e = 6$ and $r = 1.3$.

We use a standard set of data augmentations described in [11], which is widely utilized in domain generalization literature. The only difference is that in the case of distillation we use more aggressive "inception-style" random crops [33] as proposed in [4].

For each experiment, we present mean and standard deviation values computed across three random seeds. For each seed, we split source domains data into 80% and 20% train/val splits as described in [11], then train the teacher and distill it into the student. To allow fair comparison, all averaging strategies are compared on the same training trajectories, meaning that for each seed (and for each choice of target domain) both teacher training and distillation are done only once. Overall, running the presented experiments took 75 GPU days on NVIDIA V100.

	PACS					OFFICEHOME				
MODEL	A	C	P	S	AVG.	A	C	P	R	AVG.
RESNET-50 (SWAD)	91.6±0.8	85.1±1.0	98.5±0.2	82.5±0.4	89.4	70.8±0.6	57.5±1.1	81.3±0.3	82.5±0.2	73.0
RESNET-18 (SWAD)	82.9±0.2	76.7±1.9	94.9±0.2	77.5±1.6	83.0	54.2±0.3	50.6±1.0	70.6±0.6	72.2±0.3	61.9
RESNET-18 KD (ERM)	85.8±0.8	81.3±1.3	94.6±0.4	82.8±0.3	85.9	57.8±0.4	54.6±0.5	74.4±0.4	76.2±0.2	65.8
RESNET-18 KD (SWAD)	87.4±0.7	81.9±0.9	95.2±0.1	82.1±0.4	86.7	59.7±0.1	55.1±0.8	74.9±0.4	76.8±0.3	66.6
RESNET-18 KD (SMA)	86.8±0.7	81.6±1.1	95.1±0.1	81.2±0.9	86.2	59.7±0.1	55.0±0.9	74.9±0.4	76.9±0.3	66.6
RESNET-18 KD (OURS)	87.3±0.2	81.9±0.2	95.1±0.2	82.1±0.1	86.6	59.8±0.2	55.2±0.9	74.9±0.4	76.9±0.4	66.7

Table 1. Knowledge distillation with ResNet models. The teacher is ResNet-50 and the students are ResNet-18. The presented metric is target domain accuracy. ResNet-50 shows better performance here in comparison to SWAD paper [8] due to the use of a newer version of pre-trained weights in PyTorch [26].

	PACS					OFFICEHOME				
MODEL	A	C	P	S	AVG.	A	C	P	R	AVG.
DEIT-SMALL (SWAD)	91.9±0.4	85.1±0.6	98.9±0.1	82.4±1.9	89.6	72.8±0.7	61.6±1.1	81.8±0.8	83.7±0.1	75.0
DEIT-TINY (SWAD)	85.2±1.3	79.0±0.8	96.9±0.2	79.8±0.6	85.2	61.8±0.8	51.6±1.0	74.8±0.5	76.6±0.3	66.2
DEIT-TINY KD (ERM)	85.8±0.4	81.2±1.6	95.5±0.3	80.1±1.3	85.7	63.2±0.5	56.2±0.9	78.1±0.6	79.3±0.5	69.2
DEIT-TINY KD (SWAD)	88.2±0.3	82.2±0.3	97.0±0.2	83.1±0.3	87.6	66.4±0.4	56.6±0.6	79.0±0.5	80.2±0.2	70.5
DEIT-TINY KD (SMA)	88.2±0.4	82.2±0.4	96.8±0.1	82.3±1.2	87.4	65.8±0.1	56.2±0.2	78.9±0.8	80.3±0.1	70.3
DEIT-TINY KD (OURS)	88.0±0.2	82.3±0.3	96.9±0.2	83.1±0.4	87.6	66.1±0.4	56.8±0.7	78.9±0.6	80.3±0.1	70.5

Table 2. Knowledge distillation with DeiT models. The teacher is DeiT-Small and the students are DeiT-Tiny. The presented metric is target domain accuracy.

4.2. Results

The results of the experiments are presented in Table 1 and Table 2. The distillation without weight averaging (ERM) outperforms the baseline of independently trained student networks across both datasets and both architectures, indicating that the teacher's ability to generalize out-of-domain is transferred to the student to some extent and KD can perform well under domain shift. On top of that, adding our weight averaging strategy to KD leads to extra target domain accuracy improvement of +0.8pp on average in the case of ResNets and +1.6pp in the case of DeiTs. When compared to the other two strategies, WAKD shows similar performance to distillation with SWAD and slightly outperforms distillation with SMA, while being simpler and lacking the need to compute validation performance during training. However, in all cases there is still a significant gap between the target domain performance of the teacher and the student, which indicates that there is room for improvement.

5. Conclusion

In this paper, we studied the setting of performing knowledge distillation under domain shift. We showed that weight averaging techniques from the domain generalization literature, namely SWAD and SMA, can improve the performance of distillation students on unseen domains. We proposed a novel simplified weight averaging approach that does not require evaluation on the validation set during training and showed its effectiveness when applied to KD under domain shift. Better understanding the reasons behind the said improvements and their connection to flat minima, as well as exploring other distillation objectives are some of the possible future work directions.

Acknowledgements

References

[1] Isabela Albuquerque, Nikhil Naik, Junnan Li, Nitish Keskar, and Richard Socher. Improving out-of-distribution generalization via multi-task self-supervised pretraining. *arXiv preprint arXiv:2003.13525*, 2020. 2

[2] Devansh Arpit, Huan Wang, Yingbo Zhou, and Caiming Xiong. Ensemble of averages: Improving model selection and boosting performance in domain generalization. *Advances in Neural Information Processing Systems*, 35:8265–8277, 2022. 1, 2

[3] Yogesh Balaji, Swami Sankaranarayanan, and Rama Chellappa. Metareg: Towards domain generalization using meta-regularization. *Advances in neural information processing systems*, 31, 2018. 2

[4] Lucas Beyer, Xiaohua Zhai, Amélie Royer, Larisa Markeeva, Rohan Anil, and Alexander Kolesnikov. Knowledge distillation: A good teacher is patient and consistent. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10925–10934, 2022. 2, 3

[5] Gilles Blanchard, Gyemin Lee, and Clayton D. Scott. Generalizing from several related classification tasks to a new unlabeled sample. In *NIPS*, 2011. 1, 2

[6] Silvia Bucci, Antonio D'Innocente, Yujun Liao, Fabio Maria Carlucci, Barbara Caputo, and Tatiana Tommasi. Self-supervised learning across domains. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 2

[7] Fabio M Carlucci, Antonio D'Innocente, Silvia Bucci, Barbara Caputo, and Tatiana Tommasi. Domain generalization by solving jigsaw puzzles. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2229–2238, 2019. 2

[8] Junbum Cha, Sanghyuk Chun, Kyungjae Lee, Han-Cheol Cho, Seunghyun Park, Yunsung Lee, and Sungrae Park. Swad: Domain generalization by seeking flat minima. *Advances in Neural Information Processing Systems*, 34:22405–22418, 2021. 1, 2, 3, 4

[9] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 1

[10] Gongfan Fang, Yifan Bao, Jie Song, Xinchao Wang, Donglin Xie, Chengchao Shen, and Mingli Song. Mosaicking to distill: Knowledge distillation from out-of-domain data. *Advances in Neural Information Processing Systems*, 34:11920–11932, 2021. 2

[11] Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. In *International Conference on Learning Representations*, 2021. 2, 3

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1, 3

[13] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 1, 2

[14] P Izmailov, AG Wilson, D Podoprikhin, D Vetrov, and T Garipov. Averaging weights leads to wider optima and better generalization. In *34th Conference on Uncertainty in Artificial Intelligence 2018, UAI 2018*, pages 876–885, 2018. 1

[15] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 3

[16] Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Joan Puigcerver, Jessica Yung, Sylvain Gelly, and Neil Houlsby. Big transfer (bit): General visual representation learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16*, pages 491–507. Springer, 2020. 1

[17] Kyungmoon Lee, Sungyeon Kim, and Suha Kwak. Cross-domain ensemble distillation for domain generalization. In

[18] 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*, pages 5542–5550, 2017. 1, 3

[19] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Learning to generalize: Meta-learning for domain generalization. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018. 2

[20] Haoliang Li, Sinno Jialin Pan, Shiqi Wang, and Alex C Kot. Domain generalization with adversarial feature learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5400–5409, 2018. 2

[21] Xuanlin Li, Yunhao Fang, Minghua Liu, Zhan Ling, Zhuowen Tu, and Hao Su. Distilling large vision-language model with out-of-distribution generalizability. *arXiv preprint arXiv:2307.03135*, 2023. 2

[22] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021. 1

[23] Saeid Motiian, Marco Piccirilli, Donald A Adjeroh, and Gianfranco Doretto. Unified deep supervised domain adaptation and generalization. In *Proceedings of the IEEE international conference on computer vision*, pages 5715–5725, 2017. 2

[24] Krikamol Muandet, David Balduzzi, and Bernhard Schölkopf. Domain generalization via invariant feature representation. In *International Conference on Machine Learning*, pages 10–18. PMLR, 2013. 2

[25] Utkarsh Ojha, Yuheng Li, and Yong Jae Lee. What knowledge gets distilled in knowledge distillation? *arXiv preprint arXiv:2205.16004*, 2022. 2

[26] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019. 4

[27] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 2

[28] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. *arXiv preprint arXiv:1412.6550*, 2014. 2

[29] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115:211–252, 2015. 3

European Conference on Computer Vision, pages 1–20. Springer, 2022. 2

[30] Yichun Shi, Xiang Yu, Kihyuk Sohn, Manmohan Chandraker, and Anil K Jain. Towards universal representation learning for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6817–6826, 2020. 2

[31] Ankur Singh and Senthilnath Jayavelu. Robust representation learning with self-distillation for domain generalization. *ArXiv*, abs/2302.06874, 2023. 2

[32] Maryam Sultana, Muzammal Naseer, Muhammad Haris Khan, Salman Khan, and Fahad Shahbaz Khan. Self-distilled vision transformer for domain generalization. In *Proceedings of the Asian Conference on Computer Vision*, pages 3068–3085, 2022. 2

[33] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015. 3

[34] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR, 2019. 1

[35] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive representation distillation. In *International Conference on Learning Representations*, 2019. 2

[36] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International conference on machine learning*, pages 10347–10357. PMLR, 2021. 3

[37] Vladimir Vapnik and Vlamimir Vapnik. Statistical learning theory wiley. *New York*, 1(624):2, 1998. 2

[38] 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. 1, 3

[39] Riccardo Volpi and Vittorio Murino. Addressing model vulnerability to distributional shifts over image transformation sets. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7980–7989, 2019. 2

[40] Shujun Wang, Lequan Yu, Caizi Li, Chi-Wing Fu, and Pheng-Ann Heng. Learning from extrinsic and intrinsic supervisions for domain generalization. In *European Conference on Computer Vision*, pages 159–176. Springer, 2020. 2

[41] Yufei Wang, Haoliang Li, Lap-pui Chau, and Alex C Kot. Embracing the dark knowledge: Domain generalization using regularized knowledge distillation. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 2595–2604, 2021. 2

[42] Kunran Xu, Lai Rui, Yishi Li, and Lin Gu. Feature normalized knowledge distillation for image classification. In *European conference on computer vision*, pages 664–680. Springer, 2020. 2

[43] Zhenlin Xu, Deyi Liu, Junlin Yang, Colin Raffel, and Marc Niethammer. Robust and generalizable visual representation learning via random convolutions. *arXiv preprint arXiv:2007.13003*, 2020. 2

[44] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6023–6032, 2019. 2

[45] Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. In *International Conference on Learning Representations*, 2016. 2

[46] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017. 2

[47] Linfeng Zhang, Yukang Shi, Zuoqiang Shi, Kaisheng Ma, and Chenglong Bao. Task-oriented feature distillation. *Advances in Neural Information Processing Systems*, 33:14759–14771, 2020. 2

[48] Borui Zhao, Quan Cui, Renjie Song, Yiyu Qiu, and Jiajun Liang. Decoupled knowledge distillation. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, pages 11953–11962, 2022. 2

[49] Yuyang Zhao, Zhun Zhong, Fengxiang Yang, Zhiming Luo, Yaojin Lin, Shaozi Li, and Nicu Sebe. Learning to generalize unseen domains via memory-based multi-source meta-learning for person re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6277–6286, 2021. 2

[50] Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. Domain generalization: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 2

[51] Kaiyang Zhou, Yuanhan Zhang, Yuhang Zang, Jingkang Yang, Chen Change Loy, and Ziwei Liu. On-device domain generalization. *arXiv preprint arXiv:2209.07521*, 2022. 2