AUGMENTATION-DRIVEN CHECKPOINT AVERAGING FOR DOMAIN-AGNOSTIC INCREMENTAL SOUND CLASSIFICATION

Technical Report

Jiajun Sun *Zhe Gao*†

Shanghai Normal University

{ora942878@gmail.com, zgao0911@shnu.edu.cn}

ABSTRACT

We design a system for DCASE 2026 Task 7, the domain-agnostic incremental learning for audio classification, which requires a classifier to seperate sequential audios from both previous observed and newly released domains, and no class labels from previous domains are allowed in new domains. We propose our method by introducing an augmentation-driven checkpoint averaging strategy into the official CNN14-based architecture. We first use the original checkpoints from the given official CNN14 model, then apply data augmentation with randomly generated seeds for each newly released audio domain, and fine-tune the model for each seed to obtain a new checkpoint. Next, we update the checkpoint by a weighted-average method between the previous checkpoint and the newly generated one via the optimal scale parameter settled after ablation study. After the checkpoint is updated, for each audio in the new domain, five 3-second audio crops are extracted from different positions along the time axis, and the prediction via the updated checkpoint with the highest softmax confidence is the final output for this audio from the new domain. In Task 7, the checkpoints are updated twice from the initial model to domain D2 and D3. With the final checkpoint after D3, we test our method and achieve class-wise accuracies on domain D2 and D3 of 63.53% and 67.08%, respectively, or 65.31% on average. All the results are better than the baseline results, demonstrating the superiority and domain adaptivity of our method.

***Index Terms*—** domain-agnostic incremental learning, audio recognition and classification, data augmentation, checkpoints, softmax confidence

1. INTRODUCTION

DCASE 2026 Task 7 studies domain-agnostic incremental learning for audio classification [1]. In this task, an official CNN14-based classifier with initial checkpoints is given, and an updated model is required to classify audio samples from newly released domains. Different from conventional domain adaptation tasks [2], in Task 7, the class labels for the training audio samples that determined the initial checkpoints are unknown. Moreover, at each stage, the required model must learn the new domain using only data from that domain, while no access to data from earlier domains is allowed. Therefore, the model should not only classify audio samples from different domains consistently, but should also balance flexibility for the newly released domain and stability for retaining strong performance on previous domains.

To achieve this, we design a new method by introducing an augmentation-driven checkpoint averaging strategy into the official CNN14-style architecture. This method can be separated into three stages: data augmentation, checkpoint averaging, and audio cropping. First, for each newly released audio domain, we apply a stochastic data augmentation policy with different random seeds to increase the audio quantity from the same domain, and then fine-tune the model to obtain a new checkpoint for each seed. Next, we update the checkpoint with a weighted average between the checkpoint from the previous model and the mean of the fine-tuned checkpoints. An optimal weight β for the new checkpoint is searched as 0.8 via ablation study. During checkpoint updating, the multi-BN structure in the official CNN14 model is kept the same. After the checkpoint is updated, five 3-second crops are extracted from different positions along the time axis for each audio sample from the new domain. The prediction with the highest softmax confidence via the updated checkpoint is determined as the final output of this audio sample.

Inference performance of our method is measured by class-wise accuracies, the percentage of newly released audio samples from the new domain that are correctly recognized per class. By adopting the initial checkpoint from the official CNN14 model with the audio samples in Domain 2, we obtain the first updated checkpoint. With the audio samples in Domain 3 and the updated checkpoint, we obtain the second updated checkpoint, which is the final checkpoint of our proposed method. We test our method with the final checkpoint on domain D2 and D3 separately and obtain class-wise accuracies of 63.53% and 67.08%, respectively, or an arithmetic mean of 65.31%. These results are all better than those from the baseline method and demonstrate the superiority and domain adaptivity of our method.

2. BACKGROUND

2.1. Task Protocol and Metric

Task 7 in DCASE 2026 requires a domain-agnostic incremental learning (DIL) audio classifier which can learn sound events from different domains sequentially over time without significantly forgetting the knowledge from previously learned domains [1, 3]. At each sequential learning step, this classifier is only permitted to train the data from the domain in this step but may classify sound events from arbitrary domains during testing while the testing domains may be unknown. Therefore, the proposed DIL classifier should keep a performance balance between flexibility for future domains while remaining stable for domains learned in history.

Class-wise accuracies over domains with equal weight are used for performance measurement to ensure equal treatment for imbalanced datasets. Checkpoints, i.e., final system parameters, are required for program duplication and performance evaluations.

2.2. Backbone and related approaches

The official model is a modified CNN14-style convolutional neural network with multiple BN branches that reflect data distributions of different domains. The modified CNN14 is related to the family of pretrained audio neural networks (PANNs) [4], which are widely used as baselines in audio classification and recognition tasks. We keep this modified CNN14 structure as the backbone of our DIL system.

Existing methods for DIL systems usually follow three directions. The first direction focuses on reducing knowledge forgetting, with representative methods including iCaRL [5], EWC [6], and LwF [7]. The second direction improves system robustness through data or feature augmentation, such as SpecAugment [8], mixup [9], AugMix [10], and MixStyle [11]. The third direction smooths or combines model parameters, including Mean Teacher [12], stochastic weight averaging [13], model soups [14], and WiSE-FT [15].

Most of the above methods are general incremental-learning tools, while Task 7 requires audio-domain-aware sequential adaptation; therefore, we design our method around the modified CNN14 backbone and audio-specific training and inference strategies.

3. METHOD

The proposed audio-DIL system consists of three main sequential steps: stochastic data augmentation, weighted checkpoint averaging, and multi-crop inference. We first introduce the parameters used in our model and then discuss the three steps in detail.

3.1. Parameters

Given three domains D1, D2, and D3, which contain audio data with class labels, θ_t and B_t are defined as the shared non-BN parameters and BN parameters in our audio-DIL system during sequential learning stage t, respectively. The checkpoints in these stages are represented by (θ_1, B_1), (θ_2, B_2), and (θ_3, B_3). As shown in Figure 1, our audio-DIL system constructs the final checkpoint through a learning procedure from D1 to D2 and then from D2 to D3. In learning stage 1, checkpoint (θ_1, B_1) is given by the modified CNN14 baseline model trained on D1. In stages 2 and 3, checkpoints (θ_2, B_2) and (θ_3, B_3) are obtained using D2 and D3, respectively. The model with (θ_3, B_3) is our final submitted audio-DIL system for inference.

3.2. Stochastic Data Augmentation

Short-time Fourier transform (STFT) is applied to obtain the spectrogram for each audio stream in D2 and D3, and then the spectrogram is chopped into 4-second chunks. Next, each chunk goes through the stochastic data augmentation pipeline with three independent operations: audio concatenation, temporal shift, and gain perturbation.

Audio concatenation: With 50% chance, an audio chunk is selected for audio concatenation. Any selected chunk is truncated to the first entire sound event and then concatenated with other events from other chunks in the same class to form a new audio chunk with length around 4 seconds. If the chunks from the same class are insufficient, the first sound event from the selected chunk will be repeated to fulfill the required 4-second length.

Temporal shift: With 50% chance, a temporal shift following a uniform distribution is applied to an audio chunk to generate different sound event positions.

Gain perturbation: With 50% chance, a random gain perturbation following a uniform distribution in dB is applied. An augmented audio spectrogram manipulated through the three steps is displayed in Figure 2.

Inverse-frequency sampling is applied to make augmented samples more class-balanced, where each class is sampled according to the inverse square root of its sample count.

With the 50% augmentation probabilities, five augmented training datasets are independently generated for each newly released domain using different random seeds, as shown by set 1 to set 5 in Figure 1. Each dataset is used to fine-tune one model path, and the resulting fine-tuned checkpoints are averaged for checkpoint updating. This produces multiple seed-specific adaptation paths from the same domain and improves the stability of incremental learning.

3.3. Weighted Checkpoint Averaging Method

Given five augmented datasets, for a newly released domain in stage t, the non-BN parameters obtained via dataset k are denoted as $\theta_{t,k}$, and the updated non-BN parameters on all augmented data in stage t are denoted as $\frac{1}{K}\sum_{k=1}^{K}\theta_{t,k}$. The non-BN parameters calculated in the previous stage $t-1$ are denoted as θ^*_{t-1}, and the non-BN parameters in the current stage t are calculated as:

$$\theta^*_t = (1-\beta)\theta^*_{t-1} + \beta\frac{1}{K}\sum_{k=1}^{K}\theta_{t,k}, \tag{1}$$

where β is the weighted checkpoint averaging parameter that controls the tradeoff between stability, the performance retained from the previous domain, and flexibility, the performance improved for new domains. After heuristic searching in the ablation study, the optimal weight β^* is 0.8, which leverages 80% on the newly released domain. In Task 7, the optimal weight keeps the same during incremental learning from D1 to D2 and from D2 to D3.

Since non-BN parameters θ and BN parameters B converge and update independently but parallelly in a similar way in every incremental learning stage, the optimal weight β^* keeps the same value for BN parameters B. Thus, for the whole checkpoint, we have:

$$(\theta_\beta, B_\beta) = (1-\beta)(\theta_{\text{old}}, B_{\text{old}}) + \beta(\theta_{\text{ft}}, B_{\text{ft}}), \tag{2}$$

where $(\theta_{\text{old}}, B_{\text{old}})$ is the old checkpoint from the previous domain, $(\theta_{\text{ft}}, B_{\text{ft}})$ is the fine-tuned checkpoint given the newly released domain, and (θ_β, B_β) is the checkpoint after incremental learning.

3.4. Confidence-Selected Multi-Crop Inference

The multi-crop inference strategy is designed specifically for testing data after the final checkpoint is constructed. Since a discriminative sound event may only occupy a small section of an audio stream, it is hard to localize this event with a single audio crop. Given a test audio stream, five 3-second audio crops are used, and their centers are located at the 15%, 30%, 50%, 70%, and 85% positions of the total length of this test stream, respectively, as displayed with distinct colors in Figure 3. Our proposed audio-DIL system predicts these five crops independently with the final checkpoint. The class label for the testing audio stream is determined from the crop that has the highest softmax confidence.

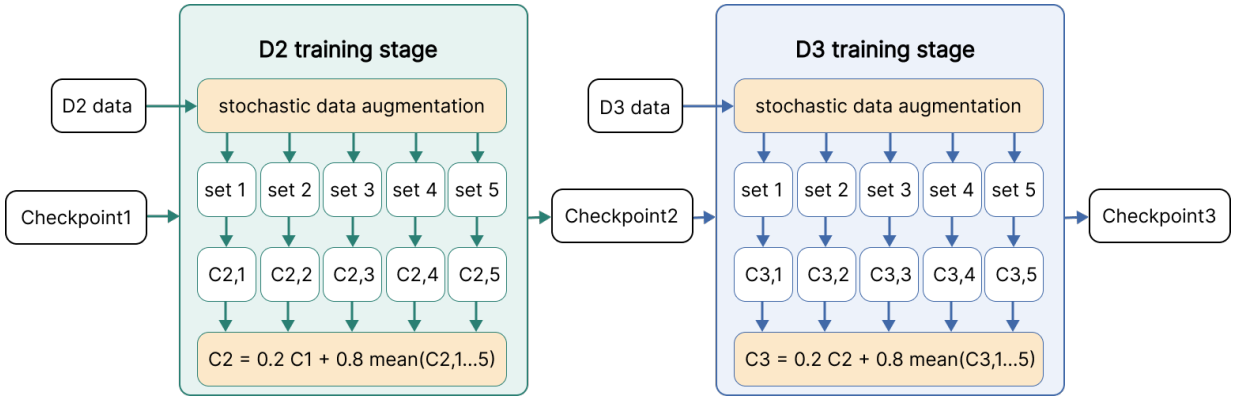

Figure 1: Serial checkpoint construction in the proposed audio-DIL system. For each new domain, five seed-specific fine-tuned checkpoints are averaged and then combined with the previous checkpoint using $\beta = 0.8$.

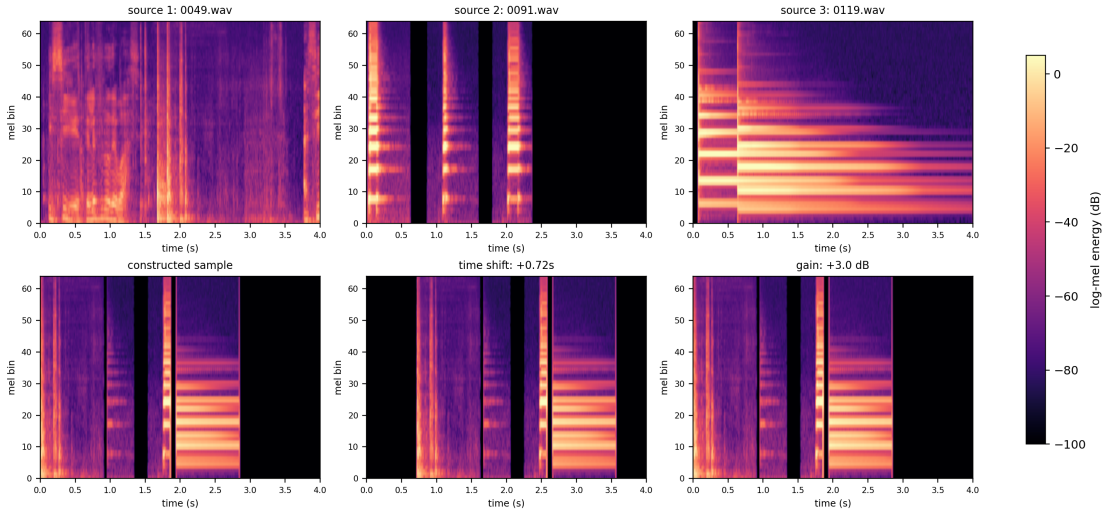

Figure 2: Example of the stochastic audio augmentation pipeline. Same-class sound events are composed into a four-second training sample, followed by temporal shift and gain perturbation.

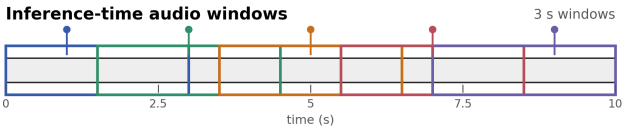

Figure 3: Confidence-selected multi-crop inference. Five 3-second crops are evaluated independently, and the crop with the highest softmax confidence is selected as the clip-level output.

4. EXPERIMENTAL SETUP

We follow the official Task 7 rules and use the official development data for evaluation. All audio streams in D2 and D3 are preprocessed into 4-second chunks following the official baseline setup. For each newly released domain, five fine-tuning paths are trained with different random seeds. Each path is trained for 120 epochs using Adam with learning rate 10^{-5}, cosine annealing to 10^{-6}, batch size 128, and class-balanced cross-entropy loss.

All initializations and model structures are kept the same except for data augmentation in the ablation study. Rather than the sequential learning process from D1 to D2 to D3, domains D2 and D3 are combined and learned together to enlarge the training dataset, while the training epochs are reduced to 40. In this way, the ablation result for data augmentation is highlighted.

5. ABLATION STUDY AND RESULTS

5.1. Evaluation Metric

The class-wise accuracy is used to evaluate the model performance. For domain d with class quantity of C_d, this accuracy is calculated as:

$$\text{Avg.}(d) = \frac{1}{|C_d|} \sum_{c \in C_d} \frac{\text{TP}_c}{\text{TP}_c + \text{FN}_c}, \tag{3}$$

where TP_c and FN_c are quantities of true positive and false negative samples in the selected class $c \in C_d$.

Table 1: Ablation study results on data augmentation.

Augmentation	D2 Avg.	D3 Avg.	Avg.
Plain	74.60	62.97	68.78
Self concat	74.10	65.82	69.96
Same-class concat	75.55	64.85	70.20
Self + shift + gain	74.47	66.11	70.29
Same-class + shift + gain	76.20	65.65	**70.93**

5.2. Data Augmentation

Table 1 reports the ablation study for data augmentation. The method "Plain" represents that no augmentation policy is adopted and only original audios are used. Both "Self concat" and "Same-class concat" denote that sound events are truncated and the concatenation operation is undertaken. "Same-class concat" represents that the truncated events are combined among the same classes to the length of 4 seconds, while "Self concat" represents that truncated events repeat themselves to the required length. "Shift" and "Gain" represent that operations of temporal shift and gain perturbation are applied, respectively. With all three augmentation operations under consideration, five augmentation combinations of "Plain", "Self concat", "Same-class concat", "Self concat+Shift+Gain", and "Same-class concat+Shift+Gain" are adopted and compared.

Displayed in Table 1, the combination of "Same-class concat" with "Shift" and "Gain" returns the best class-wise accuracy of 76.20% on D2, 65.65% on D3, and 70.93% on average. This result validates our proposed design of data augmentation with the three operations.

5.3. Optimal Beta Selection

Since domain D1 is not given, we search the optimal β^* with the official checkpoint2 (θ_2', B_2'). According to Eq. 1, after data augmentation and the five-path average fine-tuning, we can sequentially obtain a trained checkpoint3 (θ_3', B_3') by weighted averaging the official checkpoint2 (θ_2', B_2') and the fine-tuned checkpoint on domain D3. A greedy searching policy is designed for the optimal β^*. We gradually increase β from 0.60 to 1.00 with an incremental step of 0.05, assigning larger weights to the fine-tuned checkpoint from D3.

Then, the obtained checkpoint3 is tested on both domain D2 and D3 via the optimal data augmentation combination determined in Section 5.2 and the five-path average fine-tuning policy over the five augmented datasets. Next, the testing results are compared with the results on D2 and D3 via the official checkpoint2 without data augmentation, and the results via the official checkpoint2 with data augmentation but only the best fine-tuning path is applied.

The comparison results are listed in Table 2. With the increase of β, the trained checkpoint3 performs better on D3 while performance decreases on D2, since larger weights are assigned to the fine-tuned checkpoint from D3. The average class-wise accuracy over D2 and D3 is applied to determine the optimal β^*. The optimal performance occurs when β is 0.8. All the results from our weighted checkpoint averaging method perform better than the competitors of no checkpoint update with or without data augmentation, or the five-path average fine-tuning policy.

Table 2: Ablation study results on checkpoint averaging weight β.

Method	D2 Avg.	D3 Avg.	Avg.
Previous checkpoint	70.06	40.08	55.07
Single path (best in five)	50.96	68.32	59.64
Five-path update, $\beta = 0.60$	60.32	58.77	59.54
Five-path update, $\beta = 0.70$	58.95	62.70	60.83
Five-path update, $\beta = 0.75$	58.74	64.97	61.86
Five-path update, $\beta = 0.80$	57.93	66.23	**62.08**
Five-path update, $\beta = 0.85$	56.36	66.63	61.50
Five-path update, $\beta = 0.90$	55.79	67.63	61.71
Five-path update, $\beta = 1.00$	52.39	69.13	60.76

Table 3: Ablation study results on inference window.

Inference window	D2 Avg.	D3 Avg.	Avg.
3 s, 7 crops, max-conf	62.87	67.03	64.95
3 s, 5 crops, max-conf	63.53	67.08	**65.31**
4 s, 7 crops, max-conf	62.68	64.88	63.78
4 s, 5 crops, max-conf	62.43	65.09	63.76
5 s, 7 crops, max-conf	61.72	65.36	63.54
5 s, 5 crops, max-conf	61.54	65.06	63.30
Full clip	61.15	65.17	63.16

5.4. Confidence-Selected Multi-Crop Inference

Table 3 summarizes the design and selection of our multi-crop method for inference. According to our sequential learning process of D1-D2-D3, the final checkpoint (θ_3, B_3) is used, and domains D2 and D3 are applied as the testing datasets. The crop setting that returns the best average class-wise accuracy over D2 and D3 is determined as the best design. We compare critical parameters of crop quantities between 5 and 7 and crop sizes of 3, 4, and 5 seconds. The design with the best inference results is the one with 5 crops and 3-second crop length.

5.5. Final System and Results

After the ablation studies, we determine the final configuration as same-class sound event concatenation with temporal shift and gain perturbation, checkpoint averaging with $\beta^* = 0.8$, and confidence-selected inference using five 3-second crops. Checkpoint2 achieves 73.41% on D2 and 43.92% on D3, with an average score of 58.66%. After the D3 update, checkpoint3 achieves 63.53% on D2 and 67.08% on D3, improving the average score to 65.31%. Compared with the official baseline average of 52.55%, checkpoint3 provides better overall performance and balance between D2 and D3.

6. CONCLUSION

In this report, we presented a properly designed audio domain-agnostic incremental learning system for DCASE 2026 Task 7. With properly designed sequential learning steps of data augmentation, weighted checkpoint averaging, and confidence-selected multi-crop inference, this system is able to achieve posterior learning flexibility for new domains while keeping the prior learned knowledge from previous domains. After proper modification, we believe our system can be applied in other domains for more challenging incremental learning problems.

7. REFERENCES

[1] R. Casciotti, M. Mulimani, M. Harju, J. R. Jensen, and A. Mesaros, "Domain-agnostic incremental learning for sound classification. a DCASE 2026 challenge task," 2026. [Online]. Available: https://arxiv.org/abs/2606.02173

[2] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, "Domain-adversarial training of neural networks," *Journal of Machine Learning Research*, vol. 17, no. 59, pp. 1–35, 2016. [Online]. Available: https://jmlr.org/papers/v17/15-239.html

[3] DCASE Community, "DCASE 2026 challenge task 7: Domain-agnostic incremental learning for audio classification," 2026. [Online]. Available: https://dcase.community/challenge2026/task-domain-agnostic-incremental-learning-for-audio-classification

[4] Q. Kong, Y. Cao, T. Iqbal, Y. Wang, W. Wang, and M. D. Plumbley, "PANNs: Large-scale pretrained audio neural networks for audio pattern recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2880–2894, 2020.

[5] S.-A. Rebuffi, A. Kolesnikov, G. Sperl, and C. H. Lampert, "iCaRL: Incremental classifier and representation learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2001–2010. [Online]. Available: https://arxiv.org/abs/1611.07725

[6] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, D. Hassabis, C. Clopath, D. Kumaran, and R. Hadsell, "Overcoming catastrophic forgetting in neural networks," *Proceedings of the National Academy of Sciences*, vol. 114, no. 13, pp. 3521–3526, 2017.

[7] Z. Li and D. Hoiem, "Learning without forgetting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 12, pp. 2935–2947, 2018.

[8] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "SpecAugment: A simple data augmentation method for automatic speech recognition," in *Proceedings of Interspeech*, 2019, pp. 2613–2617.

[9] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," in *Proceedings of the International Conference on Learning Representations*, 2018.

[10] D. Hendrycks, N. Mu, E. D. Cubuk, B. Zoph, J. Gilmer, and B. Lakshminarayanan, "AugMix: A simple data processing method to improve robustness and uncertainty," in *Proceedings of the International Conference on Learning Representations*, 2020.

[11] K. Zhou, Y. Yang, Y. Qiao, and T. Xiang, "Domain generalization with MixStyle," in *Proceedings of the International Conference on Learning Representations*, 2021.

[12] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," in *Advances in Neural Information Processing Systems*, 2017.

[13] P. Izmailov, D. Podoprikhin, T. Garipov, D. Vetrov, and A. G. Wilson, "Averaging weights leads to wider optima and better generalization," in *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, 2018.

[14] M. Wortsman, G. Ilharco, S. Y. Gadre, R. Roelofs, R. Gontijo-Lopes, A. S. Morcos, H. Namkoong, A. Farhadi, Y. Carmon, S. Kornblith, and L. Schmidt, "Model soups: Averaging weights of multiple fine-tuned models improves accuracy without increasing inference time," in *Proceedings of the International Conference on Machine Learning*, 2022.

[15] M. Wortsman, G. Ilharco, J. W. Kim, M. Li, S. Kornblith, R. Roelofs, R. Gontijo-Lopes, H. Hajishirzi, A. Farhadi, H. Namkoong, and L. Schmidt, "Robust fine-tuning of zero-shot models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. [Online]. Available: https://arxiv.org/abs/2109.01903