# SimpleDG: Simple Domain Generalization Baseline without Bells and Whistles

Zhi Lv, Bo Lin, Siyuan Liang, Lihua Wang, Mochen Yu, Yao Tang, and Jiajun Liang

MEGVII Technology {lvzhi,linbo,liangsiyuan,wanglihua, yumochen,tangyao02,liangjiajun}@megvii.com

Abstract. We present a simple domain generalization baseline, which wins second place in both the common context generalization track and the hybrid context generalization track respectively in NICO CHAL-LENGE 2022. We verify the founding in recent literature, domainbed, that ERM is a strong baseline compared to recent state-of-the-art domain generalization methods and propose SimpleDG which includes several simple yet effective designs that further boost generalization performance. Code is available at https://github.com/megvii-research/SimpleDG.

Keywords: Domain Generalization, Domainbed, NICO++

### 1 Introduction

#### 1.1 Domain Generalization

Deep learning models have achieved tremendous success in many vision and language tasks, and even beyond human performance in well-defined and constrained tasks. However, deep learning models often fail to generalize to outof-distribution(OOD) data, which hinders greater usage and brings potential security issues in practice. For example, a self-driving car system could fail when encountering unseen signs, and a medical diagnosis system might misdiagnose with the new imaging system.

Aware of this problem, the research community has spent much effort in domain generalization(DG) where the source training data and the target test data are from different distributions. Datasets like PACS [1], VLCS [2], Office-Home [3], DomainNet [4] have been released to evaluate the generalization ability of the algorithms. Many methods like MMD [5], IRM [6], MixStyle [7], SWAD [8] have been proposed to tackle the problem.

However, We find that traditional CNN architecture with simple technologies like augmentation and ensemble, when carefully implemented, is still a strong baseline for domain generalization problem. We call our method SimpleDG which is briefly introduced in Fiugre1.



**Fig. 1:** An overview of four key designs of our method. (a) ERM is a strong baseline when carefully implemented. Many modern DG methods fail to outperform it; (b) ViTs suffer from overfitting on the small dataset without pretraining. CNN due to its proper inductive bias has a much smaller train-test accuracy gap than ViTs; (c) Stronger augmentations help generalize better. The source domain's distribution is extended by strong augmentations and gets more overlap between different domains which is of benefit to optimizing for the target domain. (d) Models ensemble improves generalization performance as output probability distributions from models compensate for each other and results in more reasonable predictions.

#### 1.2 DomainBed

DomainBed [9] is a testbed for domain generalization including seven multidomain datasets, nine baseline algorithms, and three model selection criteria. The author suggests that a domain generalization algorithm should also be responsible for specifying a model selection method. Since the purpose of DG is to evaluate the generalization ability for unseen out-of-distribution data, the test domain data should also not be used in the model selection phase. Under this circumstance, the author found that Empirical Risk Minimization(ERM) [10] results on the above datasets are comparable with many state-of-the-art DG methods when carefully implemented with proper augmentations and hyperparameter searching.

### 1.3 NICO++

NICO++ [11] is a new DG dataset recently released in the NICO challenge 2022. The goal of the NICO Challenge is to promote research on discarding spurious



Fig. 2: An overview of the NICO++ dataset. Identical contexts are shared across all different categories in the common contexts track, while contexts are mutually exclusive in the unique contexts track. Some categories might have a single context only and therefore are more likely to suffer from overfitting problems.

correlations and finding the causality in vision. The advantages of the NICO++ dataset compared with popular DG datasets are as follows: 1) more realistic and natural context semantics. All the training data is collected from the real world and categorized carefully with specific concepts; 2) more diversity is captured in the dataset, which makes generalization not trivial; 3) more challenging settings. Except for the classic DG setting, NICO++ also includes the unique contexts track where the overfitting problem is more severe.

The NICO Challenge contains two main tracks: 1) common context generalization track; 2) hybrid context generalization track. The difference between these two tracks is whether the context of all the categories is aligned and whether the domain label is available. Same as the classic DG setting, identical contexts are shared across all categories in both training and test data in the common context generalization track. However, contexts are mutually exclusive in the hybrid context generalization track as shown in Figure 2. Context labels are available for the common context generalization track, but not for the hybrid context generalization track.

One main challenge of this competition comes from the unique context. Some of the samples have unique contexts, which might cause the model to overfit the unrelated background information. Another challenge comes from the small 4 Zhi Lv, Bo Lin et al.

object and multi-object samples. As we observe, some samples in the dataset contain extremely small target objects. It's very likely to crop the background part when generating the training data which may cause training noise and introduce bias. There are also some samples that have more than one target category. This may cause the model to be confused and overfit noise. The rule of preventing using extra data also makes the task harder since large-scale pretrained models are not permitted.

# 2 SimpleDG

In this section, we first introduce the evaluation metric of our experiments and then discuss four major design choices of our method named SimpleDG, including

- Why ERM is chosen as a baseline over other methods
- Why CNN is favored over ViT in this challenge
- How does augmentation help in generalization
- How to scale up the models to further improve performance

#### 2.1 Evaluation Metric

To evaluate the OOD generalization performance internally, we use 4 domains, i.e. dim, grass, rock, and water, as the training domains(in-distribution) and 2 domains, i.e. autumn and outdoor, as the test domains(out-of-distribution). For model selection, we split 20% of the training data of each domain as the validation dataset and select the model with the highest validation top-1 accuracy. All numbers are reported using top-1 accuracy on unseen test domains.

For submission, we retrain the models using all domains with a lower validation percentage(5%) for both track1 and track2. Because we find that the more data we use, the higher accuracy we got in the public test dataset.

### 2.2 Key Designs of SimpleDG

### I. ERM is a simple yet strong baseline

A recent literature [9] argues that many DG methods fail to outperform simple ERM when carefully implemented on datasets like PACS and Office-Home, and proposes a code base, called domainbed, including proper augmentations, hyperparameter searching and relatively fair model selection strategies.

We conduct experiments on NICO dataset with this code base and extend the algorithms and augmentations in domainbed. Equipped with our augmentations, we compare ERM with recent state-of-the-art DG algorithms. We find the same conclusion that most of them have no clear advantage over ERM as shown in Table1.

algorithm	test acc
GroupDRO [12]	69.0
MMD [5]	69.4
MixStyle [7]	68.3
SelfReg [13]	69.5
CORAL [14]	68.6
SD [15]	69.3
RSC [16]	69.0
ERM [10]	70.1

 Table 1: Many DG methods fail to outperform simple ERM

### II. ViTs suffer from overfitting on small training sets without pretraining

ViT [17] has shown growing popularity these years, and we first compare the performance of ViT with popular CNN in track1. We choose one CNN model, ResNet18, and two vision transformer model, ViT-B/32 and CLIP [18]. CNN outperforms ViT significantly when trained from scratch with no pre-trained weights. ViT achieves higher training accuracy but fails to generalize well on unseen test domains. We tried ViT training tricks such as LayerScale [19] and stochastic depth [20]. The test accuracy improves, but there is still a huge gap compared with CNN as shown in Table2. On the contrary, the ViTs outperform CNN when using pre-trained weights and finetuning on NICO dataset.

We surmise that ViTs need more amount of training to generalize than CNNs as no strong inductive biases are included. So we decide not to use them since one of the NICO challenge rules is that no external data (including ImageNet) can be used and the model should be trained from scratch.

Table 2: Test domain accuracy of CNN and ViTs on NICO track1

	$\operatorname{ResNet18}$	$\rm ViT\text{-}B/32$	CLIP
w/ pretrain	81	87	90
w/o pretrain	64	30	

### III. More and stronger augmentation help generalize better

Both track1 and track2 suffer from overfitting since large train-validation accuracy gaps are clearly observed. Track2 has mutually exclusive contexts across categories and therefore suffers more from overfitting. With relatively weak augmentations, the training and test accuracy saturate quickly due to the overfitting problem. Generalization performance improves by adding more and stronger augmentations and applying them with a higher probability.



Fig. 3: Visualization of Fourier Domain Adaptation. The low-frequency spectrum of the content image and the style image is swapped to generate a new-style image.

Following the standard ImageNet training method, we crop a random portion of the image and resize it to 224x224. We adopt timm [21]'s RandAugment which sequentially applies N(default value 2) operations randomly chosen from a candidate operation list, including auto-contrast, rotate, shear, etc, with magnitude M(default value 10) to generate various augmented images. Test domain accuracy gets higher when more candidate operations(color jittering, grayscale and gaussian blur, etc.) are applied, and larger M and N are used.

Mixup [22] and FMix [23] are simple and effective augmentations to improve generalization. Default Mixup typically samples mixing ratio  $\lambda \approx 0$  or 1, making one of the two interpolating images dominate the interpolated one. RegMixup [24] proposes a simple method to improve generalization by enforcing mixing ratio distribution concentrate more around  $\lambda \approx 0.5$ , by simply increase the hyperparameter  $\alpha$  in mixing ratio distribution  $\lambda \sim Beta(\alpha, \alpha)$ . We apply RegMix to both Mixup and Fmix to generate augmented images with more variety. With these stronger augmentations, we mitigate the saturation problem and benefit from a longer training schedule.

For domain adaption augmentation, we adopt Fourier Domain Adaptation [25] proposed by Yang et al. FDA uses Fourier transform to do analogous "style transfer". FDA requires two input images, reference and target images, it can generate the image with the "style" of the reference image while keeping the semantic "content" of the target image as shown in Figure 3. The breakdown effect for each augmentation is shown in Table 3.

**Table 3:** The breakdown effect for augmentation, high resolution finetuning and ensemble inference for Top-1 accuracy (%) of NICO challenge track 1 training on ResNet-101.

Method	Top-1 accuracy (%)
Vanilla	81.22
+ RandAugment	82.85
+ Large alpha Mixup series	84.58
+ Fourier Domain Adaptation	85.61
+ High Resolution Fine-tune	86.01
+ Ensemble inference	87.86

### IV. Over-parameterized models saturate quickly, and ensemble models continue to help

Big models are continuously refreshing the best results on many vision and language tasks. We investigate the influence of model capacity on NICO with the ResNet [26] series. When we test ResNet18, ResNet50, and ResNet101, the accuracy improves as the model size increases. But when we continue to increase the model size as large as ResNet152, the performance gain seems to be saturated. The capacity of a single model might not be the major bottleneck for improving generalization when only the small-scale dataset is available.

To further scale up the model, we adopt the ensemble method which averages the outputs of different models. When we average ResNet50 and ResNet101 as an ensemble model whose total flops is close to ResNet152, the performance gets higher than ResNet152. When further averaging different combinations of ResNet50, ResNet101, and ResNet152, the test accuracy get up to 2% improvement. The ensemble method results are shown in Figure 4.

To figure out how ensemble helps, we conduct the following experiments. We first study ensemble models of best epochs from different train runs with the same backbone such as ResNet101. There is nearly no performance improvement even with a large ensemble number. The variety of candidate models should be essential for the ensemble method to improve performance. We launch experiments with different settings including different augmentations and different random seeds which influence the training/validation data split while still keeping the backbone architecture the same, i.e. ResNet101, among all experiments. This time, the ensemble models of these ResNet101s get higher test accuracy. We conclude that model variety not only comes from backbone architecture but also can be influenced by experiment settings that might lead to significantly different local minimums.

#### 2.3 More Implementation Detail

**Distributed training.** We re-implemented the training config using PyTorch's Distributed Data Parallels framework [27]. We can train ResNet101 with 512 batch-size in 10 hours with 8 GPUs(2080ti).



Fig. 4: Test domain accuracy with different model size

**Training from scratch.** All models use MSRA initialization [28]. We set the gamma of all batch normalization layers that sit at the end of a residual block to zero, SGD optimizer with 0.9 momentum, linear scaling learning rate, 1e-5 weight decay. We use 224x224 resized input images, 512 batch size, learning rate warmup for 10 epochs, and cosine annealing in overall 300 epochs. All experiments are trained with 8 GPUs.

Fine-tune in high-resolution. We fine-tune all models in 448x448 resolution and 128 batch size for 100 epochs, this can further boost the model performance. Ensemble inference. In the inference phase, we use the ensemble method to reduce the variance. We average the features before softmax layer from multiple models, which is better than logits averaging after softmax layer.

### 2.4 Public Results

The top-10 public test dataset, which is available during the competition, results in track1 and track2 are shown in the Table 4.

### 2.5 Private Results

The NICO official reproduced our method and tested it on the private test dataset, which is unavailable during the competition, and the results are shown in Table5.

Our method is quite stable between public dataset and private dataset, the ranking stays the same in track1 and becomes better in track2 while other methods undergo ranking turnover.

### SimpleDG 9

Table 4: Top-10 Teams' Public Dataset Results of Track1 and Track2

Rank	Track1		Track2	
	Team	Top1-Acc	Team	Top1-Acc
1	$detectors_218$	88.15704	PingPingPang	84.62421
2	megvii_is_fp(Ours)	87.85504	vtddggg	84.05049
3	mcislab840	87.53865	timmy11hu	81.99656
4	ShitongShao	86.83397	megvii_is_fp( <b>Ours</b> )	81.49168
5	MiaoMiao	85.83447	Wentian	79.91968
6	Wentian	85.75538	czyczyyzc	79.35743
7	test404	85.54685	wangyuqing	78.81813
8	peisen	85.46775	Jarvis-Tencent-KAUST	78.78371
9	HuanranChen	84.92126	Wild	78.41652
10	wangyuqing	84.6624	peisen	77.80838

 Table 5: Top-5 Teams' Private Dataset Results of Track1 and Track2

	Team	Phase 1 R	ank Phase 2 Score	Phase 2 Rank
Track1	MCPRL-TeamSpirit	1	0.7565	1
	megvii-biometrics(Ours)	2	0.7468	2
	DCD404	6	0.7407	3
	mcislab840	3	0.7392	4
	MiaoMiao	4	0.7166	5
Track2	vtddggg	2	0.8123	1
	megvii-biometrics(Ours)	4	0.788	2
	PingPingPangPangBangBangBang	1	0.7631	3
	jarvis-Tencent-KAUST	5	0.7442	4
	PoER	8	0.6724	5

## 3 Conclusion

In this report, we proposed SimpleDG which wins both the second place in the common context generalization track and the hybrid context generalization track of NICO CHALLENGE 2022. With proper augmentations and a longer training scheduler, the ERM baseline could generalize well on unseen domains. Many existing DG methods failed to continue to increase the generalization from this baseline. Based on ERM, both augmentations and model ensembles played an important role in further improving generalization.

After participating in the NICO challenge, we found that simple techniques such as augmentation and model ensemble are still the most effective ways to improve generalization. General and effective domain generalization methods are in demand, but there is still a long way to go. 10 Zhi Lv, Bo Lin et al.

### References

- 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.
- Chen Fang, Ye Xu, and Daniel N Rockmore. Unbiased metric learning: On the utilization of multiple datasets and web images for softening bias. In *Proceedings of* the IEEE International Conference on Computer Vision, pages 1657–1664, 2013.
- 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.
- 4. Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1406–1415, 2019.
- 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.
- Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. arXiv preprint arXiv:1907.02893, 2019.
- Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. Domain generalization with mixstyle. arXiv preprint arXiv:2104.02008, 2021.
- 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.
- Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. arXiv preprint arXiv:2007.01434, 2020.
- Vladimir N Vapnik. An overview of statistical learning theory. *IEEE transactions* on neural networks, 10(5):988–999, 1999.
- Xingxuan Zhang, Linjun Zhou, Renzhe Xu, Peng Cui, Zheyan Shen, and Haoxin Liu. Nico++: Towards better benchmarking for domain generalization. ArXiv, abs/2204.08040, 2022.
- 12. Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. *arXiv preprint arXiv:1911.08731*, 2019.
- Daehee Kim, Youngjun Yoo, Seunghyun Park, Jinkyu Kim, and Jaekoo Lee. Selfreg: Self-supervised contrastive regularization for domain generalization. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 9619–9628, 2021.
- Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *European conference on computer vision*, pages 443–450. Springer, 2016.
- 15. Mohammad Pezeshki, Oumar Kaba, Yoshua Bengio, Aaron C Courville, Doina Precup, and Guillaume Lajoie. Gradient starvation: A learning proclivity in neural networks. *Advances in Neural Information Processing Systems*, 34:1256–1272, 2021.
- Zeyi Huang, Haohan Wang, Eric P Xing, and Dong Huang. Self-challenging improves cross-domain generalization. In *European Conference on Computer Vision*, pages 124–140. Springer, 2020.

- 17. 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.
- 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.
- Hugo Touvron, Matthieu Cord, Alexandre Sablayrolles, Gabriel Synnaeve, and Hervé Jégou. Going deeper with image transformers. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 32–42, 2021.
- Gao Huang, Yu Sun, Zhuang Liu, Daniel Sedra, and Kilian Q Weinberger. Deep networks with stochastic depth. In *European conference on computer vision*, pages 646–661. Springer, 2016.
- Ross Wightman. Pytorch image models. https://github.com/rwightman/ pytorch-image-models, 2019.
- 22. Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.
- Ethan Harris, Antonia Marcu, Matthew Painter, Mahesan Niranjan, Adam Prügel-Bennett, and Jonathon Hare. Fmix: Enhancing mixed sample data augmentation. arXiv preprint arXiv:2002.12047, 2020.
- Francesco Pinto, Harry Yang, Ser-Nam Lim, Philip HS Torr, and Puneet K Dokania. Regmixup: Mixup as a regularizer can surprisingly improve accuracy and out distribution robustness. arXiv preprint arXiv:2206.14502, 2022.
- Yanchao Yang and Stefano Soatto. Fda: Fourier domain adaptation for semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 4085–4095, 2020.
- 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.
- Shen Li, Yanli Zhao, Rohan Varma, Omkar Salpekar, Pieter Noordhuis, Teng Li, Adam Paszke, Jeff Smith, Brian Vaughan, Pritam Damania, et al. Pytorch distributed: Experiences on accelerating data parallel training. arXiv preprint arXiv:2006.15704, 2020.
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In Proceedings of the IEEE international conference on computer vision, pages 1026–1034, 2015.