Delving Deep into the Generalization of Vision Transformers under Distribution Shifts

Chongzhi Zhang[1*], **Mingyuan Zhang**[2*], **Shanghang Zhang**[*], **Daisheng Jin**[1], **Qiang Zhou**[3], **Zhongang Cai**[2], **Haiyu Zhao**[24], **Shuai Yi**[24], **Xianglong Liu**[1], **Ziwei Liu**[2]

[1]Beihang University
[2]S-Lab, Nanyang Technological University
[3]Shandong University
[4]Shanghai AI Laboratory
chongzhizhang@buaa.edu.cn, shzhang.pku@gmail.com, ziwei.liu@ntu.edu.sg

Abstract

Recently, Vision Transformers (ViTs) have achieved impressive results on various vision tasks. Yet, their generalization ability under different distribution shifts is rarely understood. In this work, we provide a comprehensive study on the out-of-distribution generalization of Vision Transformers. To support a systematic investigation, we first present a taxonomy of distribution shifts by categorizing them into five conceptual groups: corruption shift, background shift, texture shift, destruction shift, and style shift. Then we perform extensive evaluations of ViT variants under different groups of distribution shifts and compare their generalization ability with Convolutional Neural Network (CNN) models. Several important observations are obtained: **1)** ViTs generalize better than CNNs under multiple distribution shifts. With the same or less amount of parameters, ViTs are ahead of corresponding CNNs by more than 5% in top-1 accuracy under most types of distribution shift. **2)** Larger ViTs gradually narrow the in-distribution (ID) and out-of-distribution (OOD) performance gap. To further improve the generalization of ViTs, we design the Generalization-Enhanced Vision Transformers by integrating adversarial learning, information theory, and self-supervised learning. By investigating these three types of generalization-enhanced Transformers, we observe the gradient-sensitivity of Vision Transformers and design a smoother learning strategy to achieve a stable training process. With modified training schemes, we achieve improvements on performance towards out-of-distribution data by 4% from vanilla ViTs. We comprehensively compare these three types of generalization-enhanced Vision Transformers with their corresponding CNN models, and observe that: **1)** For the enhanced model, larger ViTs still benefit more for the out-of-distribution generalization. **2)** generalization-enhanced Vision Transformers are more sensitive to the hyper-parameters than their corresponding CNN models. We hope our comprehensive study could shed light on the design of more generalizable learning architectures. Our code is available at https://github.com/Phoenix1153/ViT_OOD_generalization.

1 Introduction

Since its debut, transformer has made remarkable achievements in natural language processing [26, 5, 21]. Recently, researchers have successfully adopted transformers to computer vision, *e.g.* image

[*]These authors contributed equally to this work.

Preprint. Under review.

classification [4, 6, 25], object detection [2, 33] and image processing [3]. Despite the encouraging results achieved on standard benchmarks, Vision Transformers remain dubious in their deployment in the wild. In controlled laboratory environments, it is generally assumed that the test data for model evaluation are independent identically distributed (IID) with sampled training data. However, this assumption does not always hold in real-world applications. Thus, models are desired to generalize to out-of-distribution (OOD) data, *i.e.* data under distribution shifts. There are abundant researches in analysing model OOD generalizations on CNN architectures [10–12] while the investigation in ViTs remains scarce. Therefore, in this paper, we mainly focus on delving deep into the OOD generalization of Vision Transformers under distribution shifts.

To comprehensively study the OOD generalization ability of Vision Transformers, we first define a categorization of commonly appearing distribution shifts based on the modified semantic concepts in images. Generally, an image for classification contains a foreground object and the background information. The foreground object consists of hierarchical semantic concepts including pixel-level elements, object textures, and shapes, object parts, and object itself [32]. A distribution shift usually causes variance on one or more semantics and we thus present a taxonomy of distribution shifts into five conceptual groups: background shifts, corruption shifts, texture shifts, destruction shifts, and style shifts.

With the taxonomy of distribution shifts, we investigate the OOD generalization of Vision Transformers by comparison with state-of-the-art CNNs in each case. We use DeiT models for investigation due to their data efficiency. We obtain the following observations from our extensive evaluation: **1)** Vision Transformers generalize better than CNNs in most cases. Specifically, Vision Transformer not only achieves better performance on out-of-distribution data but also has smaller generalization gaps between in-distribution and out-of-distribution datasets. **2)** As the model scale increases, Vision Transformers gradually narrow the in-distribution and out-of-distribution generalization gaps, especially in the case of corruption shifts and background shifts. Thus, we can conclude that larger ViTs are better at diminishing the effect of local changes. **3)** Vision Transformers trained with larger patch size deal with texture shifts better, yet are inferior in other cases. The results of this part are shown in Fig. 1 (a).

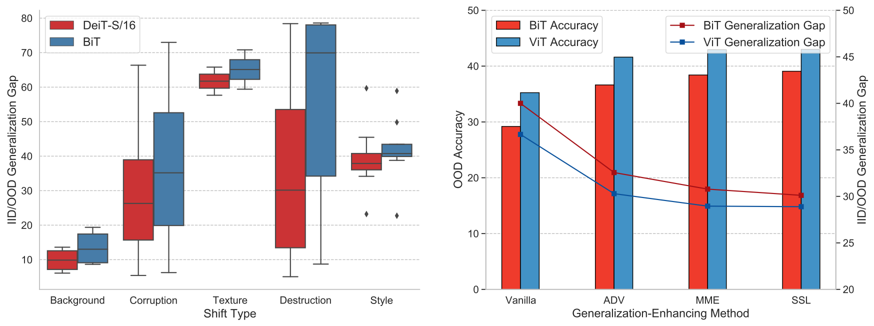
After validating the superiority of ViTs in dealing with out-of-distribution data, we focus on further improving their generalization capacity. Specifically, we design Generalization-Enhanced Vision Transformers by integrating adversarial training [8], information theory [23] and self-supervised learning [31]. The designed three types of enhanced Vision Transformers are named T-ADV, T-MME, and T-SSL, respectively. Equipped with Generalization-Enhanced ViTs, we achieve significant performance boosts towards out-of-distribution data by 4% from vanilla ViTs. By performing an in-depth investigation on different models, we draw the following conclusions: **1)** For the enhanced transformer models, larger ViTs still benefit more for the out-of-distribution generalization. **2)** Generalization-Enhanced Vision Transformers are more sensitive to the hyper-parameters than their corresponding CNN models. The results of this part are shown in Fig. 1 (b).

2 Taxonomy of Distribution Shifts and Evaluation Protocols

2.1 Taxonomy of Distribution Shifts

To make an extensive study on out-of-distribution generalization, we build the taxonomy of distribution shifts upon what kinds of semantic concepts are modified from the original image. In the object recognition task, an image usually consists of a foreground object and the background. Previous works generally suppose a hierarchy on the semantic concepts emerging in the image [32]. These semantic concepts can be listed from low-level to high-level: pixel-level elements, object textures and shapes, object parts, objects. Therefore, we divide the distribution shifts into five cases: background shifts, corruption shifts, texture shifts, destruction shifts and style shifts, as shown in Fig. 2.

- **Background Shifts.** Image backgrounds are usually regarded as auxiliary cues in assigning images to corresponding labels in the image classification task. However, previous works have demonstrated that backgrounds may dominate in prediction [22, 1], which is undesirable to us. We focus on the model's invariance towards background change and thus define the background shifts. *ImageNet-9* [29] is adopted for background shifts.



(a) (b)

Figure 1: **A quick glance of our investigation observations.** (a) Investigation of IID/OOD Generalization Gap implies that ViTs generalize better than CNNs under most types of distribution shifts. (b) Combined with generalization-enhancing methods, we achieve significant performance boosts on the OOD data by 4% compared with vanilla ViTs, and consistently outperform the corresponding CNN models. The enhanced ViTs also have smaller IID/OOD Generalization Gap than the ehhanced BiT models.

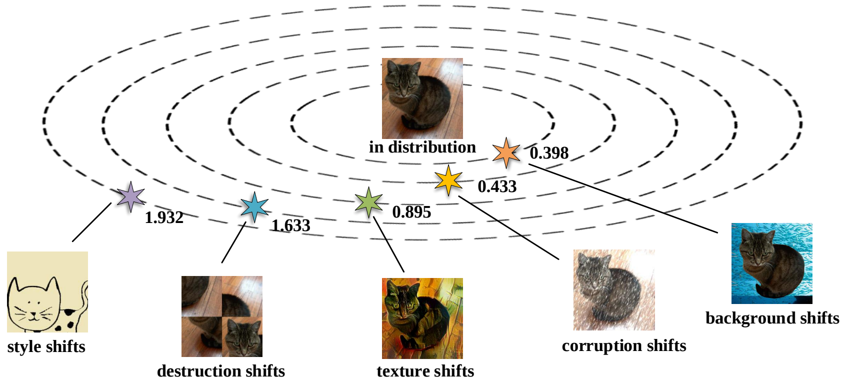


Figure 2: **Illustration of our taxonomy of distribution shifts.** We build the taxonomy upon what kinds of semantic concepts are modified from the original image. We divide the distribution shifts into five cases: background shifts, corruption shifts, texture shifts, destruction shifts, and style shifts. We apply the proxy $\mathcal{A}$-distance (PAD) [9] as an empirical measurement of distribution shifts. We select a representative sample of each distribution shift type and rank them by their PAD values (illustrated nearby the stars), respectively. Please refer to Appendix C for details.

- **Corruption Shifts.** The concept of corruption was proposed in Hendrycks and Dietterich [11], which stands for those naturally occurring vicinal impurities mixed in images. These corruptions either come from environmental influence during the shooting stage or from the image processing stage. We define these cases as corruption shifts, which only impact on object pixel-level elements while can still cause models obvious performance decrease. *ImageNet-C* [11] is used to examine generalization ability under corruption shifts.
- **Texture Shifts.** Generally, the texture gives us information about the spatial arrangement of the colors or intensities in an image, which is critical for the classifiers in obtaining a correct prediction. Thus, a replacement of object textures can influence model prediction. We define these variations as texture shifts. *Cue Conflict Stimuli* and *Stylized-ImageNet* [10] are used to investigate generalization under texture shifts.
- **Destruction Shifts.** The destruction shifts correspond to breaking the whole object into pieces. For example, the random patch-shuffling process splits the image into several square patches and random shuffles the position of these patches. *Random patch-shuffling* is utilized for destruction shifts to destruct images into random patches. This process can destroy long-range object information and the severity increases as the split numbers grow. In addition, we make a variant by further

divide each patch into two right triangles and respectively shuffle two types of triangles. We name the process *triangular patch-shuffling*.

- **Style Shifts.** Typically, style is a complicated concept determined by the characteristics that describe the artwork, such as the form, color, composition, etc. The variance of style often reflects in multiple concept levels, including texture, shape and object part, etc. For instance, comparing a stick figure with a corresponding photo, we could observe the difference by textures and colors, as well as the ignorance of some unimportant object parts in the stick figure. *ImageNet-R* [12] and *DomainNet* [20] are used for the case of style shifts.

We list the detailed descriptions of the used datasets in Appendix B.1.

2.2 Model Zoo

- **Vision Transformer.** We follow the implementation in DeiT [25] and choose a range of models with different scales for experiments. The ViT architecture takes as input a grid of non-overlapping contiguous image patches of resolution $N \times N$. In this paper we typically use $N = 16$ ("/16") or $N = 32$ ("/32"). Besides the official DeiT models, we also utilize the data-efficient training scheme to train ViT-L/16 and ViT-B/32 and rename them DeiT-L/16 and DeiT-B/32.
- **Big Transfer.** Big Transfer models [14] are build on ResNet-V2 models. We select BiT-S-R50X1 based on a ResNet-50 backbone. Besides the official implementation, we also train a version using identical data augmentation strategy from DeiTs for comparison. We respectively name them BiT and BiT$_{da}$.

The configurations of the used networks are summarized in Table 2 in Appendix.

2.3 Evaluation Protocols

In image classification tasks, a model generally consists of a feature encoder F and a classifier C. Suppose the model is trained on a training set $\mathscr{D}_{train} = \{(\mathrm{x}_i, y_i)\}_{i=1}^{N_{train}}$. We respectively introduce a set of independent identically distributed (IID) validation data $\mathscr{D}_{iid} = \{(\mathrm{x}_i, y_i)\}_{i=1}^{N_{iid}}$ and a set of out-of-distributed (OOD) data $\mathscr{D}_{ood} = \{(\mathrm{x}_i, y_i)\}_{i=1}^{N_{ood}}$ in the same semantic space. $N_{train}, N_{iid}, N_{ood}$ represent the number of data in $\mathscr{D}_{train}, \mathscr{D}_{iid}, \mathscr{D}_{ood}$ respectively. Then we use the following evaluations.

- **Accuracy on OOD Data.** A direct measurement is to calculate the accuracy on the OOD dataset:

$$Acc(F, C; \mathscr{D}_{ood}) = \frac{1}{|\mathscr{D}_{ood}|} \sum_{(\mathrm{x},y)\in\mathscr{D}_{ood}} \mathbb{1}(C(F(\mathrm{x})) = y), \tag{1}$$

  where $\mathbb{1}$ is the indicator function.
- **IID/OOD Generalization Gap.** In this paper, we also focus on how well a model could behave towards the OOD data compared with the IID data. Hence, we use the IID/OOD generalization gap to measure the performance difference caused by the distribution shift:

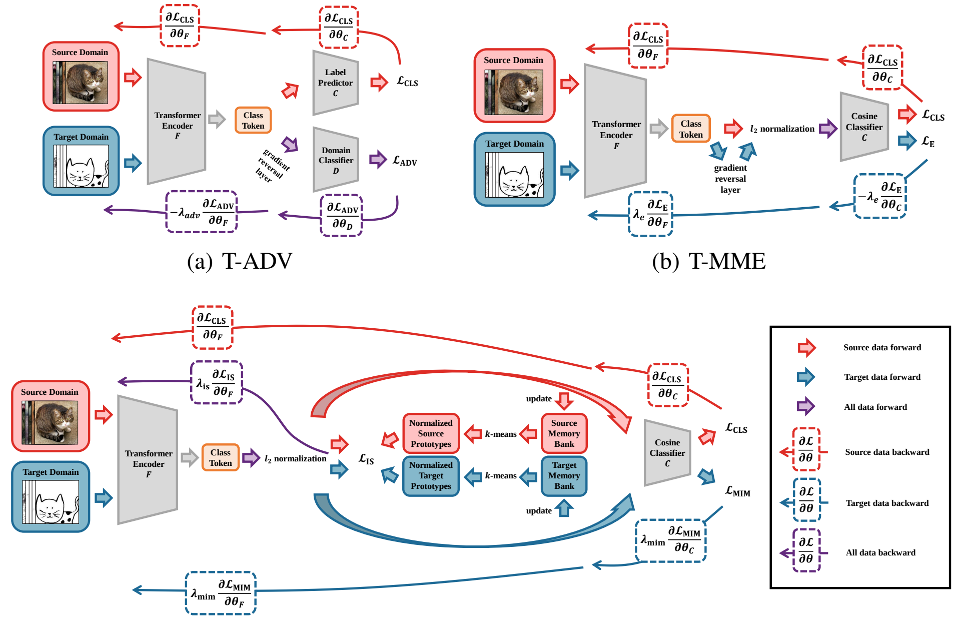
$$Gap(F, C; \mathscr{D}_{iid}, \mathscr{D}_{ood}) = Acc(F, C; \mathscr{D}_{iid}) - Acc(F, C; \mathscr{D}_{ood}). \tag{2}$$

3 Generalization-Enhanced Vision Transformers

After investigating the OOD generalization properties of ViTs, it is natural to figure out strategies to further improve them. There are recent popular designs on basis of adversarial training, information theory and self-supervised learning. We further borrow these ideas and propose three frameworks. The designed three types of enhanced Vision Transformers are named as T-ADV, T-MME and T-SSL.

3.1 Domain Adversarial Learning

To learn domain-invariant representations, we introduce a domain discriminator [8] to promote the backbone to produce domain-confused features by adversarial training. Specifically, as shown in Fig


Source Domain
Target Domain
Transformer Encoder F
Class Token
Label Predictor C
Domain Classifier D
gradient reversal layer
$\mathcal{L}_{CLS}$
$\mathcal{L}_{ADV}$
$\frac{\partial \mathcal{L}_{CLS}}{\partial \theta_F}$
$\frac{\partial \mathcal{L}_{CLS}}{\partial \theta_C}$
$-\lambda_{adv}\frac{\partial \mathcal{L}_{ADV}}{\partial \theta_F}$
$\frac{\partial \mathcal{L}_{ADV}}{\partial \theta_D}$
l_2 normalization
Cosine Classifier C
$\mathcal{L}_E$
$\lambda_e\frac{\partial \mathcal{L}_E}{\partial \theta_F}$
$-\lambda_e\frac{\partial \mathcal{L}_E}{\partial \theta_C}$
Normalized Source Prototypes
Normalized Target Prototypes
k-means
Source Memory Bank
Target Memory Bank
update
$\mathcal{L}_{IS}$
$\mathcal{L}_{MIM}$
$\lambda_{is}\frac{\partial \mathcal{L}_{IS}}{\partial \theta_F}$
$\lambda_{mim}\frac{\partial \mathcal{L}_{MIM}}{\partial \theta_C}$
$\lambda_{mim}\frac{\partial \mathcal{L}_{MIM}}{\partial \theta_F}$
Source data forward
Target data forward
All data forward
Source data backward
Target data backward
All data backward

(a) T-ADV

(b) T-MME

(c) T-SSL

Figure 3: **A framework overview of the three designed generalization-enhanced ViTs.** All networks use a Vision Transformer F as feature encoder and a label prediction head C. Under this setting, the inputs to the models have labeled source examples and unlabeled target examples. **a) T-ADV** promotes the network to learn domain-invariant representations by introducing a domain classifier D for domain adversarial training. **b) T-MME** leverage the minimax process on the conditional entropy of target data to reduce the distribution gap while learning discriminative features for the task. The network uses a cosine similarity-based classifier architecture C to produce class prototypes. **c) T-SSL** is an end-to-end prototype-based self-supervised learning framework. The architecture uses two memory banks V^s and V^t to calculate cluster centroids. A cosine classifier C is used for classification in this framework.

3 (a), the network consists of a shared feature encoder F, a label predictor C and a domain classifier D. The feature encoder aims at minimizing the domain confusion loss $\mathcal{L}_{\mathrm{ADV}}$ for all samples and label prediction loss $\mathcal{L}_{\mathrm{CLS}}$ for labeled source samples while the domain classifier focus on maximizing the domain confusion loss $\mathcal{L}_{\mathrm{ADV}}$. The overall objectives are:

$$\begin{aligned} &\mathcal{L}_{\mathrm{CLS}} = \textstyle\sum_{(\mathrm{x},y)\in\mathscr{D}_s} \mathcal{H}(\sigma(C(F(\mathrm{x}))), y), && \mathcal{L}_{\mathrm{ADV}} = \textstyle\sum_{(\mathrm{x},y_d)\in\mathscr{D}_s,\mathscr{D}_t} \mathcal{H}(\sigma(D(F(\mathrm{x}))), y_d), \\ &(\hat{\theta}_F, \hat{\theta}_C) = \arg\min_{\theta_F,\theta_C} \mathcal{L}_{\mathrm{CLS}} + \lambda_{adv}\mathcal{L}_{\mathrm{ADV}}, && \hat{\theta}_D = \arg\max_{\theta_D} \mathcal{L}_{\mathrm{ADV}}, \end{aligned} \tag{3}$$

where y and y_d denote the class label and binary domain label respectively. $\sigma(\cdot)$ stands for the Softmax function and $\mathcal{H}(\cdot,\cdot)$ returns the cross-entropy of two input distributions. λ_{adv} is an adaptive coefficient that gradually changed from 0 to 1 by the schedule proposed in [8]. Furthermore, to facilitate the training procedure, a gradient reversal layer (GRL) is applied to implement the opposite objective of two parts.

3.2 Minimax Entropy

We leverage the minimax process on the conditional entropy of target data [23] to reduce the distribution gap while learning discriminative features for the task. As the pipeline is shown n Fig 3 (b), a cosine similarity-based classifier architecture C is exploited to produce class prototypes. The cosine classifier C consists of weight vectors $\mathrm{W} = [\mathrm{w}_1, ..., \mathrm{w}_{n_c}]$, where n_c denotes the total number of classes, and a temperature T. C takes ℓ_2 normalized $\frac{F(\mathrm{x})}{\|F(\mathrm{x})\|}$ as an input and output $\frac{1}{T}\frac{\mathrm{W}^{\mathrm{T}}F(\mathrm{x})}{\|F(\mathrm{x})\|}$. The key idea is to minimize the distance between the class prototypes and neighboring unlabeled target samples, thus extracting discriminative target features. To overcome the dominant impact of labeled source data on prototypes, prototypes are moved towards the target by maximizing the entropy $\mathcal{L}_{\mathrm{E}}$ of unlabeled target examples. Meanwhile, the feature extractor aims at minimizing the entropy of the unlabeled examples, to make them better clustered around the prototypes. Therefore, a minimax process is formulated between the weight vectors and the feature extractor. Additionally, the label prediction loss $\mathcal{L}_{\mathrm{CLS}}$ is also utilized on source samples. The overall objectives are:

$$\begin{array}{ll} \mathcal{L}_{\mathrm{CLS}} = \sum_{(\mathrm{x},y)\in\mathscr{D}_s} \mathcal{H}(\sigma(C(F(\mathrm{x}))), y), & \mathcal{L}_{\mathrm{E}} = \sum_{\mathrm{x}\in\mathscr{D}_t} \mathcal{H}(\sigma(C(F(\mathrm{x})))), \\ \hat{\theta}_F = \arg\min_{\theta_F} \mathcal{L}_{\mathrm{CLS}} + \lambda_e \mathcal{L}_{\mathrm{E}}, & \hat{\theta}_C = \arg\min_{\theta_C} \mathcal{L}_{\mathrm{CLS}} - \lambda_e \mathcal{L}_{\mathrm{E}}, \end{array} \tag{4}$$

where $\mathcal{H}(\cdot,\cdot)$ returns the cross-entropy of two input distributions and $\mathcal{H}(\cdot)$ returns the entropy. λ_e is a coefficient to balance two loss terms.

3.3 Self-Supervised Learning

We integrate an end-to-end prototypical self-supervised learning framework [31] into vision transformer. As shown in Fig. 3 (c), the framework also uses a cosine classifier C as introduced in Section 3.2. It first encodes semantic structure of data into the embedding space. ProtoNCE [16] is respectively applied in source and target domains. Specifically, two memory banks V^s and V^t are maintained to store feature vectors of every sample from source and target. These vectors are updated with momentum after each batch. k-means clustering is performed on memory banks to generate normalized prototypes $\{\mu_j^s\}_{j=1}^k$ and $\{\mu_j^t\}_{j=1}^k$. Then the similarity distribution vector between ℓ_2 normalized source feature vectors $f_i^s = \frac{F(\mathrm{x}_i^s)}{\|F(\mathrm{x}_i^s)\|}$ from current batch and normalized source prototypes $\{\mu_j^s\}_{j=1}^k$ as $P_i^s = [P_{i,1}^s, ..., P_{i,k}^s]$ with $P_{i,j}^s = \frac{\exp(\mu_j^s \cdot f_i^s/\phi)}{\sum_{r=1}^k \exp(\mu_r^s \cdot f_i^s/\phi)}$, where ϕ is a temperature value. Then the in-domain prototypical self-supervision loss is formed as: $\mathcal{L}_{\mathrm{IS}} = \sum_{i=1}^{|\mathscr{D}_s|} \mathcal{H}(P_i^s, c_s(i)) + \sum_{i=1}^{|\mathscr{D}_t|} \mathcal{H}(P_i^t, c_t(i))$, where $c_s(\cdot)$ and $c_t(\cdot)$ return the cluster index of the sample, and $|\cdot|$ returns the cardinal of the set. $\mathcal{H}(\cdot,\cdot)$ returns the cross-entropy of two input distributions.

In addition, since a network is desired to have high-confident and diversified predictions, an objective is set for maximizing the mutual information between the input image and the network prediction. This objective is split into two terms: entropy maximization of expected network prediction and entropy minimization on the network output. Therefore, the objective is formulated as: $\mathcal{L}_{\mathrm{MIM}} = \mathbb{E}_{\mathrm{x}}[\mathcal{H}(p(y|\mathrm{x};\theta)] - \mathcal{H}(\mathbb{E}_{\mathrm{x}\in\mathscr{D}_s\cup\mathscr{D}_t}[p(y|\mathrm{x};\theta])$. The last term of training objective is the supervision loss on source domain measured by cross-entropy: $\mathcal{L}_{\mathrm{CLS}} = \sum_{(\mathrm{x},y)\in\mathscr{D}_s} \mathcal{H}(\sigma(C(F(\mathrm{x}))), y)$.

Finally, the overall learning objective is formulated as:

$$(\hat{\theta}_F, \hat{\theta}_C) = \arg\min_{\theta_F,\theta_C} \mathcal{L}_{\mathrm{CLS}} + \lambda_{\mathrm{is}} \mathcal{L}_{\mathrm{IS}} + \lambda_{\mathrm{mim}} \mathcal{L}_{\mathrm{MIM}}, \tag{5}$$

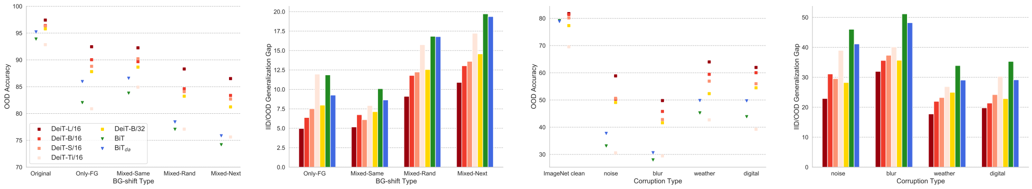
where λ_{is} and λ_{mim} denotes the coefficients of corresponding loss terms.

4 Experiments

4.1 Systematic Study on the Generalization of ViTs

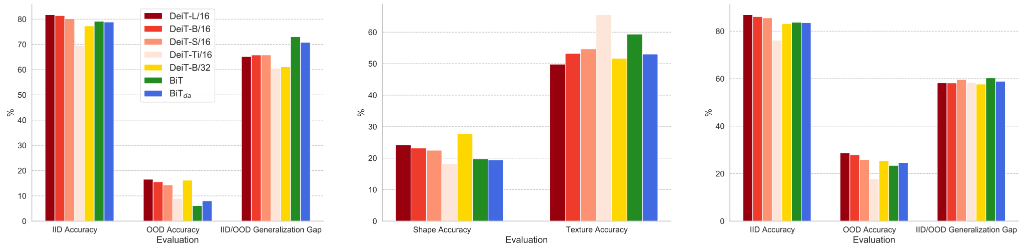
In-Distribution Generalization. We first examine the in-distribution generalization of different models on the ImageNet benchmark. As results are shown in Fig. 4 (c) column 1, we have the following observations. **1)** With the data-efficient training scheme, DeiT models tend to perform better as scales increase from *tiny* to *large*, but the gain of scale growth gradually dwindles. **2)** Having almost the same parameters and both trained without external data, DeiT-S/16 could beat BiT and BiT$_{da}$.

Background Shifts Generalization. The OOD accuracy and IID/OOD gap results of four varieties of background shifts are illustrated in Fig. 4 (a) and (b) respectively. From the results, we have the following observations. **1)** All Transformers are ahead of CNNs in both OOD accuracy and IID/OOD Gaps except for DeiT-Ti/16 due to its compactness. **2)** A larger Transformer architecture contributes to a better OOD performance as well as a smaller IID/OOD gap. Even DeiT-L/16 could further narrow the gap by about 2% from DeiT-B/16, while they achieve almost the same in distribution accuracy results. **3)** Comparing two BiT models, BiT$_{da}$ outperforms normal BiT in OOD accuracy. However, their IID/OOD gaps only differ in 'Only-FG' and 'Mixed-Same', where the backgrounds are either purely black or replaced by the same class. When the backgrounds are selected from the



(a) (b) (c) (d)

Figure 4: **Results on ImageNet-9 and ImageNet-C.** (a)-(b) and (c)-(d) respectively illustrate the OOD Accuracy and IID/OOD Generalization Gap for different models on ImageNet-9 and ImageNet-C datasets. From (a) and (b), we conclude that **1)** ViTs are less likely to assign a background to labels, and this special quality of ViTs is not obtained by complicated augmentations in the training procedure. **2)** A larger ViT tends to focus more attention on the foreground and learn a more background-irrelevant representation. From (c) and (d), we draw the conclusions that **1)** a larger ViT generalizes better in this case, **2)** patch size for training has little influence on generalization ability from IID data to OOD data but only acts on the model in distribution generalization.



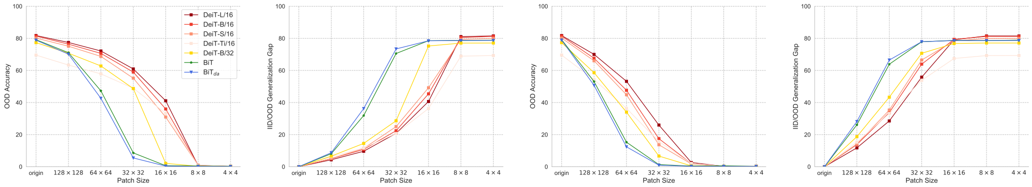
(a) Stylized-ImageNet (b) Cue Conflict Stimuli (c) ImageNet-R

Figure 5: **Results on Stylized-ImageNet, Cue Conflict Stimuli and ImageNet-R.** (a), (b) and (c) respectively illustrate the OOD Accuracy and IID/OOD Generalization Gap for different models on Stylized-ImageNet, Cue Conflict Stimuli, and ImageNet-R data sets. From (a) and (b) we could draw the following conclusions that **1)** Vision Transformers deal with the texture shifts better, **2)** a larger Transformer contributes to better leveraging global semantic features (such as shape and object parts) and less affected by local changes, **3)** Vision Transformers with larger patch size rely less on local texture features and focus more on global high-level features. From (c) we observe that most ViTs beat BiTs in OOD accuracy while having little difference in the IID/OOD generalization gap.

other classes ('Mixed-Rand' and 'Mixed-Next'), there is little difference. Based on these observations, we could conclude that **1)** ViTs are less likely to assign a background to labels, and this special quality of ViTs is not obtained by complicated augmentations in the training procedure. **2)** A larger Transformer tends to focus more attention on the foreground and learn a more background-irrelevant representation.

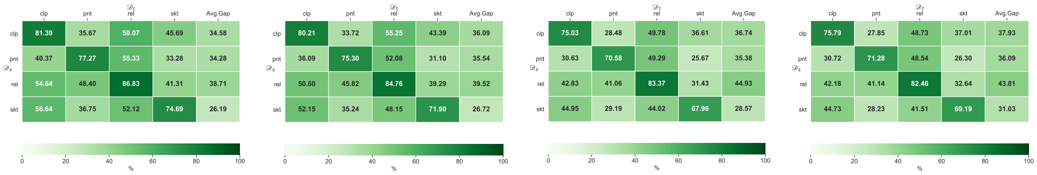
Corruption Shifts Generalization. The corruption results of 4 categories averaged over all subclasses and all severities, are shown in Fig. 4 (c) and (d). From the results, we first observe some similar phenomena with the background shifts results. **1)** Most Transformers lead the BiT models to a large extent under both evaluations in all situations. **2)** A larger Transformer architecture achieves a better OOD performance and narrows the IID/OOD generalization gap. we also have different observations. **1)** Compared with BiT, BiT$_{da}$ constantly achieves about 4% better in OOD performances and IID/OOD gaps. **2)** Though DeiT-B/16 leads DeiT-B/32 in OOD accuracy, their gaps are similar. From these phenomena we could conclude that **1)** a larger Transformer generalizes better in this case, **2)** models shall benefit from diverse augmentation when facing such vicinal impurities, **3)** patch size for training has little influence on generalization ability from IID data to OOD data but only act on the model in distribution generalization.

Texture Shifts Generalization. The results on Stylized-ImageNet is shown in Fig. 5 (a). We could observe that **1)** Vision Transformers lead BiT models under both evaluations. **2)** A larger Transformer architecture achieves a better OOD performance. **3)** DeiT-B/32 behaves better than DeiT-B/16 in OOD accuracy and IID/OOD generalization gap, which is opposite to their performances on ImageNet. These phenomena reappear in results on Cue Conflict Stimuli shown in Fig. 5 (b) that **1)**



(a) (b) (c) (d)

Figure 6: **Results on patch-shuffled and triangular patch-shuffled ImageNet.** (a)-(b) and (c)-(d) respectively illustrate the OOD Accuracy and IID/OOD Generalization Gap for different models on patch-shuffled and triangular patch-shuffled ImageNet with different splitting numbers. The image resolution we use for experiments is 384 × 384. The generalization of ViTs towards such distribution shifts all collapse when the split patch size gets smaller than the one used for their training. We can thus conclude that ViTs rely less on spatial information (provided by position embedding) than CNNs if the inner-patch information is intact, while cannot stand the inner-patch destroy.



	clp	pnt	rel	skt	Avg.Gap
clp	81.39	35.67	59.07	45.69	34.58
pnt	40.37	77.27	55.33	33.28	34.28
rel	54.64	48.40	86.83	41.31	38.71
skt	56.64	36.75	52.12	74.69	26.19

(a) DeiT-B/16

	clp	pnt	rel	skt	Avg.Gap
clp	80.21	33.72	55.25	43.39	36.09
pnt	36.09	75.30	52.08	31.10	35.54
rel	50.60	45.82	84.76	39.29	39.52
skt	52.15	35.24	48.15	71.90	26.72

(b) DeiT-S/16

	clp	pnt	rel	skt	Avg.Gap
clp	75.03	28.48	49.78	36.61	36.74
pnt	30.63	70.58	49.29	25.67	35.38
rel	42.83	41.06	83.37	31.43	44.93
skt	44.95	29.19	44.02	67.96	28.57

(c) BiT

	clp	pnt	rel	skt	Avg.Gap
clp	75.79	27.85	48.73	37.01	37.93
pnt	30.72	71.28	48.54	26.30	36.09
rel	42.18	41.14	82.46	32.64	43.81
skt	44.73	28.23	41.51	69.19	31.03

(d) BiT$_{da}$

Figure 7: **Results on DomainNet.** From the results, we can conclude that **1)** DeiT-S/16 performs better on the small-scale datasets in IID conditions. Thus, the model easily outperforms BiTs in OOD accuracy, **2)** when inspecting the IID/OOD generalization gap, the results differ a lot. When models are trained on clipart and painting, there is no obvious difference of gap between DeiT-S/16 and BiTs.

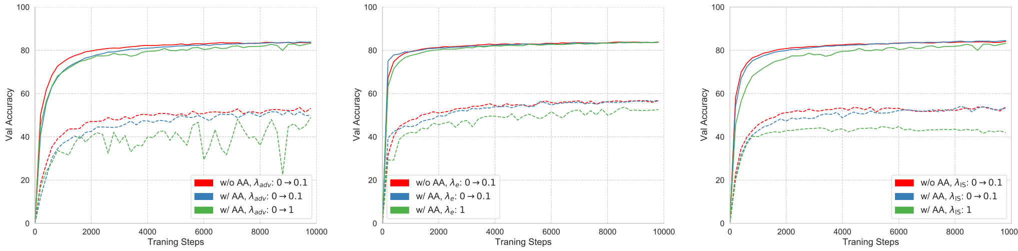
most Vision Transformers achieve higher shape accuracy and lower texture accuracy than BiTs. **2)** there exists an uptrend of shape accuracy and a downtrend of texture accuracy as the Transformer size increases **3)** DeiT-B/32 is less affected by the misleading texture than DeiT-B/16, resulting in a higher shape accuracy. We could draw the following conclusions that **1)** Vision Transformers deal with the texture shifts better, **2)** a larger Transformer contributes to better leveraging global semantic features (such as shape and object parts) and less affected by local changes, **3)** Vision Transformers with larger patch size rely less on local texture features and focus more on global high-level features.

Destruction Shifts Generalization. The patch-shuffling results are illustrated in Fig. 6 (a) and (b), which show some interesting phenomena. **1)** During the demultiplication of split patch size from 128 × 128 to 32 × 32, BiT models gradually come to collapse while ViTs are insensitive towards the distribution shift, i.e. the IID/OOD gap increases very slowly. **2)** However, generalization of ViTs towards such distribution shift all collapse when the split patch size gets smaller than the one used for their training. When inspecting the results of triangular patch-shuffling in Fig. 6 (c) and (d), we observe that the performance gaps between ViTs and BiTs decline faster since the triangular patches destruct the context in square-shape patches for Transformers. These findings indicate that Vision Transformers rely less on spatial information (provided by position embedding) than CNNs if the inner-patch information is intact, while cannot stand the inner-patch destroy.

Style Shifts Generalization. The results on ImageNet-R is shown in Fig. 5 (c). Since ImageNet-R only contains 200 classes of ImageNet, we follow [12] to record model accuracy on the corresponding ImageNet subset named ImageNet-200 and regard it as the IID result. When focusing on the accuracy on ImageNet-R, we could observe that **1)** Most ViTs beat BiTs in OOD accuracy, while having little difference in the IID/OOD generalization gap. **2)** DeiT-B/16 has better accuracy than DeiT-B/32. For DomainNet, we mainly compare the models with the same scale, i.e. DeiT-S/16 and BiTs. We observe that **1)** DeiT-S/16 performs better on the small-scale datasets in IID conditions. Thus, the model easily outperforms BiTs in OOD accuracy. **2)** When inspecting the IID/OOD generalization gap, the results differ a lot. When models are trained on clipart and painting, there is no obvious difference of gap between DeiT-S/16 and BiTs. **3)** But in the case of real, DeiT-S/16 leads BiTs more than 4%, which could be explained as the Transformers utilize the knowledge from pre-train data better if the pre-train data and downstream data are from similar distribution.

Table 1: **Results of Generalization-enhanced methods.** Specifically, we compare three types of Generalization-enhanced ViTs with their corresponding CNNs. From the results we could conclude that 1) equipped with Generalization-Enhanced ViTs, we achieve significant performance boosts towards out-of-distribution data by 4% from vanilla ViTs. 2) three generalization-enhanced ViTs have almost the same improvement from vanilla models on OOD accuracy. 3) for the enhanced transformer models, larger ViTs still benefit more for the out-of-distribution generalization.

Model	Method	R to C	R to P	P to C	C to S	S to P	R to S	P to R	Avg.
DeiT-B/16	-	54.64	48.40	40.37	45.69	36.75	41.31	55.33	46.07
	T-ADV	58.19	50.85	41.91	51.18	46.12	47.47	55.65	50.20
	T-MME	**60.59**	**51.98**	42.30	50.32	45.79	**47.92**	54.87	50.54
	T-SSL	56.80	49.06	**45.96**	**51.79**	**46.95**	45.95	**60.98**	**51.07**
DeiT-S/16	-	50.60	45.82	36.09	43.39	35.24	39.29	52.08	43.22
	T-ADV	53.60	47.84	37.99	47.10	41.61	41.94	52.82	46.13
	T-MME	**56.86**	**49.15**	38.97	46.48	42.95	**42.07**	52.49	47.00
	T-SSL	53.86	46.71	**42.79**	**47.25**	**43.01**	40.94	**57.07**	**47.37**
BiT	-	42.18	41.14	30.72	37.01	28.23	32.64	48.54	36.78
	DANN [8]	45.20	42.86	32.96	40.44	36.63	35.26	49.25	40.37
	MME [23]	50.21	**44.61**	34.75	40.27	38.41	37.83	47.58	41.95
	SSL [31]	**52.55**	42.80	**39.03**	**45.72**	**39.08**	**39.65**	**56.07**	**44.98**
VGG-16	-	39.39	37.32	26.36	32.96	25.55	27.79	45.70	33.58
	DANN [8]	43.26	40.09	28.68	36.22	31.63	**35.45**	44.73	37.15
	MME [23]	42.65	**42.46**	27.41	**36.93**	33.94	32.58	**45.87**	37.41
	SSL [31]	**43.79**	41.88	**32.19**	35.73	**36.99**	31.05	55.18	**39.54**



(a) T-ADV (b) T-MME (c) T-SSL

Figure 8: **Investigation of Generalization-enhanced methods with different training strategies.** (a)-(c) show training curves on both source domain and target domain. From the results, we can conclude that classical training strategies(the green lines) on CNNs are not suitable for ViTs, which need smoother strategies (the red lines) to align features in both domains.

4.2 Effectiveness of Generalization-Enhanced ViTs

Setup. We use DomainNet [20] for the following experiments. Following [23], we focus on the 7 scenarios listed in Tab. 1. To make a full comparison, we implement these enhancing techniques on two representative CNNs VGG-16 and BiT, and two Vision Transformers including DeiT-S/16 and DeiT-B/16. We explore their performance on both the vanilla version and the generalization-enhanced version. Implementation details can be found in supplementary materials.

Performance Analysis. The results of three generalization-enhanced ViTs comparing with CNNs are shown in Tab. 1. From the results we have the following observations: **1)** equipped with Generalization-Enhanced ViTs, we achieve significant performance boosts towards out-of-distribution data by 4% from vanilla ViTs. **2)** Three generalization-enhanced ViTs have almost the same improvement from vanilla models on OOD accuracy. In contrast, CNNs benefit more from the self-supervised learning method than the others. **3)** DeiT-B/16 has a larger gain on those enhancing methods than DeiT-S/16. Therefore, we conclude that **1)** ViTs and CNNs share many characteristics, and both can be beneficial from the generalization-enhancement methods. **2)** For the enhanced transformer models, larger ViTs still benefit more for the out-of-distribution generalization.

Smooth Feature Alignment. Fig. 8 shows the performance of our generalization-enhanced ViTs with different training strategies. The green line represents the same training strategies used in CNNs.

The other two lines use smoother strategies. From the comparison of these strategies, we observe that **1)** the generally used automated augmentation schemes shall cause performance degradation on T-ADV while has little influence on T-MME and T-SSL. **2)** smoother learning strategies are significant for ViT convergence, especially in adversarial training mode. As for T-MME and T-SSL, smoothness of auxiliary losses also significantly improve the performance. Based on these observations, we draw the conclusion that Generalization-Enhanced Vision Transformers are more sensitive to the hyper-parameters than their corresponding CNN models.

5 Conclusion

In this work, we provide a comprehensive study on the out-of-distribution generalization of Vision Transformers, with the following contributions: **1)** We define a taxonomy on data distribution shifts according to the modified semantic concepts in images. **2)** We perform a comprehensive study on out-of-distribution generalization properties of Vision Transformers under the five categorized distribution shifts. Several valuable observations are obtained. **3)** We further improve the out-of-distribution generalization of Vision Transformers by designing Generalization-Enhanced ViTs through adversarial learning, information theory, and self-supervised learning with smoother training strategies. Our work serves as an early attempt, thus there is plenty of room for developing more powerful generalization-enhanced ViTs.

References

[1] Andrei Barbu, David Mayo, Julian Alverio, William Luo, Christopher Wang, Dan Gutfreund, Josh Tenenbaum, and Boris Katz. Objectnet: A large-scale bias-controlled dataset for pushing the limits of object recognition models. *Advances in neural information processing systems*, 32: 9453–9463, 2019. 2

[2] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part I*, volume 12346 of *Lecture Notes in Computer Science*, pages 213–229. Springer, 2020. 2, 12

[3] Hanting Chen, Yunhe Wang, Tianyu Guo, Chang Xu, Yiping Deng, Zhenhua Liu, Siwei Ma, Chunjing Xu, Chao Xu, and Wen Gao. Pre-trained image processing transformer. *arXiv preprint arXiv:2012.00364*, 2020. 2, 12

[4] Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In *International Conference on Machine Learning*, pages 1691–1703. PMLR, 2020. 2, 12

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics, 2019. 1, 12

[6] 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. 2, 12, 14

[7] Qi Dou, Daniel C Castro, Konstantinos Kamnitsas, and Ben Glocker. Domain generalization via model-agnostic learning of semantic features. *arXiv preprint arXiv:1910.13580*, 2019. 13

[8] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *International conference on machine learning*, pages 1180–1189. PMLR, 2015. 2, 4, 5, 9, 13, 14, 16, 17

[9] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The journal of machine learning research*, 17(1):2096–2030, 2016. 3, 15

[10] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. Imagenet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In *International Conference on Learning Representations*, 2019. 2, 3, 13

[11] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2019. 3, 13

[12] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. *arXiv preprint arXiv:2006.16241*, 2020. 2, 4, 8, 13

[13] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1501–1510, 2017. 13

[14] Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Joan Puigcerver, Jessica Yung, Sylvain Gelly, and Neil Houlsby. Big transfer (bit): General visual representation learning. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part V*, volume 12350 of *Lecture Notes in Computer Science*, pages 491–507. Springer, 2020. 4, 14

[15] 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. 13

[16] Junnan Li, Pan Zhou, Caiming Xiong, Richard Socher, and Steven CH Hoi. Prototypical contrastive learning of unsupervised representations. *arXiv preprint arXiv:2005.04966*, 2020. 6

[17] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael Jordan. Learning transferable features with deep adaptation networks. In *International conference on machine learning*, pages 97–105. PMLR, 2015. 13

[18] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 14

[19] Ilya Loshchilov and Frank Hutter. Fixing weight decay regularization in adam. 2018. 14

[20] 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. 4, 9, 13

[21] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019. 1, 12

[22] Amir Rosenfeld, Richard Zemel, and John K Tsotsos. The elephant in the room. *arXiv preprint arXiv:1808.03305*, 2018. 2

[23] Kuniaki Saito, Donghyun Kim, Stan Sclaroff, Trevor Darrell, and Kate Saenko. Semi-supervised domain adaptation via minimax entropy. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8050–8058, 2019. 2, 5, 9, 14, 16, 17

[24] Yu Sun, Eric Tzeng, Trevor Darrell, and Alexei A Efros. Unsupervised domain adaptation through self-supervision. *arXiv preprint arXiv:1909.11825*, 2019. 13

[25] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. *arXiv preprint arXiv:2012.12877*, 2020. 2, 4, 12, 13, 14

[26] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008, 2017. 1, 12

[27] Riccardo Volpi, Hongseok Namkoong, Ozan Sener, John Duchi, Vittorio Murino, and Silvio Savarese. Generalizing to unseen domains via adversarial data augmentation. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 5339–5349, 2018. 13

[28] Yuqing Wang, Zhaoliang Xu, Xinlong Wang, Chunhua Shen, Baoshan Cheng, Hao Shen, and Huaxia Xia. End-to-end video instance segmentation with transformers. *arXiv preprint arXiv:2011.14503*, 2020. 12

[29] Kai Xiao, Logan Engstrom, Andrew Ilyas, and Aleksander Madry. Noise or signal: The role of image backgrounds in object recognition. *arXiv preprint arXiv:2006.09994*, 2020. 2, 13

[30] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems*, volume 32, 2019. 12

[31] Xiangyu Yue, Zangwei Zheng, Shanghang Zhang, Yang Gao, Trevor Darrell, Kurt Keutzer, and Alberto Sangiovanni Vincentelli. Prototypical cross-domain self-supervised learning for few-shot unsupervised domain adaptation. *arXiv preprint arXiv:2103.16765*, 2021. 2, 6, 9, 16, 17

[32] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014. 2

[33] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable {detr}: Deformable transformers for end-to-end object detection. In *International Conference on Learning Representations*, 2021. 2, 12

A Detailed Related Work

A.1 Transformers

A.1.1 Transformer Architectures

Benefiting from the self-attention mechanism to capture global dependencies among context, Transformer architectures have been widely applied in natural language modeling since proposed by Vaswani et al. [26]. With sufficient data for pre-training, Transformers are able to achieve remarkable performance on a wide range of tasks [5, 21, 30].

A Transformer model is made up by stacking Transformer blocks. Each Transformer block consists of a multi-head self-attention layer (MSA) and a token-wise feed-forward layer. Layer-norm (LN) is applied before each layer and residual connections in both the self-attention layer and the feed-forward layer. The standard Transformer receives a sequence of token embeddings as input. The self-attention layer updates these embeddings by computing pairwise dot product attention between them.

A.1.2 Transformers in Vision Tasks

Recent researchers have explored to apply Transformers to various vision tasks including image classification [4, 6, 25], object detection [2, 33], segmentation [28] and image processing [3]. Among them, the Vision Transformer (ViT) [6] is the first fully-transformer model applied for image classification and competitive with state-of-the-art convolutional neural networks (CNNs). ViT uses the standard Transformer architecture with a few modifications. To handle 2D images, an image is first split into a sequence of flattened patches with the same resolution and then processed by a trainable linear projection. Besides, the authors also append a learnable embedding to the sequence of embedded patches as the CLS token, whose representation serves for final classification. Though as a milestone for Transformers in vision task, ViT still heavily relies on large-scale datasets such as ImageNet-21k and JFT-300M (publically unavailable) for model pre-training, requiring huge computation resources. Later, Touvron et al. [25] propose the Data-efficient image Transformer (DeiT), which achieves competitive results against the state-of-the-art CNNs on ImageNet without external data by simply changing training strategies from ViT. Since its efficiency, we use this family of models for investigating generalizations of Vision Transformers under distribution shifts in this paper.

A.2 Out-of-distribution Generalization

Attracting much attention recently, various works have been proposed for the problem of out-of-distribution (OOD) generalization under different settings. The domain adaptation literature focuses on methods aimed at promoting the model's performance under the distribution shift. For instance, Unsupervised Domain Adaptation methods are designed with the assumption that unlabeled data from the target domain are available during training [8, 17, 24]. Another general setting for OOD generalization concentrates on learning representations without attaching them to any target data. This is commonly referred as domain generalization [15, 27, 7]. In addition, some recent works characterize model OOD generalization on their newly-built testing benchmarks. Hendrycks and Dietterich [11] analyze image models and show that they are sensitive to various simulated image corruptions (e.g., noise, blur, weather, JPEG compression, etc.) from their ImageNet-C benchmark. [10] provide a benchmark Stylized-ImageNet by using neural style transfer. This dataset resembles ImageNet in appearance but varies in texture. ImageNet-R [12] is a dataset with different renditions of semantic information from ImageNet classes. These renditions, such as paintings or embroidery, differ from the ImageNet images in local textures and global appearance.

B Experimental Setup

B.1 Dataset Zoo

- **ImageNet-9** [29] is adopted for background shifts. ImageNet-9 is a variety of 9-class datasets with different foreground-background recombination plans, which helps disentangle the impacts of foreground and background signals on classification. In our case, we use the four varieties of generated background with foreground unchanged, including 'Only-FG', 'Mixed-Same', 'Mixed-Rand' and 'Mixed-Next'. The 'Original' data set is used to represent in-distribution data.
- **ImageNet-C** [11] is used to examine generalization ability under corruption shifts. ImageNet-C includes 15 types of algorithmically generated corruptions, grouped into 4 categories: 'noise', 'blur', 'weather', and 'digital'. Each corruption type has five levels of severity, resulting in 75 distinct corruptions.
- **Cue Conflict Stimuli** and **Stylized-ImageNet** are used to investigate generalization under texture shifts. Utilizing style transfer, Geirhos et al. [10] generated **Cue Conflict Stimuli** benchmark with conflicting shape and texture information, that is, the image texture is replaced by another class with other object semantics preserved. In this case, we respectively report the shape and texture accuracy of classifiers for analysis. Meanwhile, **Stylized-ImageNet** is also produced in [10] by replacing textures with the style of randomly selected paintings through AdaIN style transfer [13].
- **Random patch-shuffling** is utilized for destruction shifts to destruct images into random patches. This process can destroy long-range object information and the severity increases as the split numbers grow. In addition, we make a variant by further divide each patch into two right triangles and respectively shuffle two types of triangles. We name the process **triangular patch-shuffling**.
- **ImageNet-R** [12] and **DomainNet** [20] are used for the case of style shifts. ImageNet-R [12] contains 30000 images with various artistic renditions of 200 classes of the original ImageNet validation data set. The renditions in ImageNet-R are real-world, naturally occurring variations, such as paintings or embroidery, with textures and local image statistics which differ from those of ImageNet images. DomainNet [20] is a recent benchmark dataset for large-scale domain adaptation that consists of 345 classes and 6 domains. As labels of some domains are very noisy,

we follow the 7 distribution shift scenarios in [23] with 4 domains (Real, Clipart, Painting, Sketch) picked.

B.2 Implementation Details

B.2.1 Vanilla Model Implementation

- **DeiT.** For Vision Transformers, we pre-train the DeiT models on the ImageNet dataset with the AdamW optimizer [19], a batch size of 1024 and the resolution of 224×224. The learning rate is linearly ramped up during the first 5 epochs to its base value determined with the following linear scaling rule: lr = 0.0005*batch size/512. After the warmup, we decay the learning rate with a cosine schedule [18] with the weight decay = 0.05 and train for 300 epochs. We follow the data augmentations scheme in the official paper [25]. For the downstream fine-tuning, we scale up the resolution to 384×384 by adopting perform 2D interpolation of the pre-trained position embeddings proposed in [6]. We train models with learning rate lr = 5e-6, weight decay = 1e-8 for 75000 iters and the other settings identical to pre-training stage.
- **BiT.** BiT models use the augmentation schemes mentioned in [14]. We train them upstream using SGD with momentum. We use an initial learning rate of 0.03, weight decay 0.0001, and momentum 0.9. We train for 90 epochs and decay the learning rate by a factor of 10 at 30, 60, and 80 epochs. We use a global batch size of 4096 and multiply the learning rate by batch size/256 with the resolution 224×224. For downstream fine-tuning, we use SGD with an initial learning rate of 0.003, momentum 0.9, and batch size 512 with the resolution 384×384.
- **BiT$_{da}$.** BiT$_{da}$ models use identical data augmentation strategy from DeiTs. The other training setups are consistent with BiT models.

B.2.2 Generalization-Enhanced Model Implementation

- **T-ADV.** T-ADV consists of a feature encoder, a label predictor, and a domain classifier. The feature encoder is implemented by the DeiT backbone and the label predictor is a linear layer projecting CLS token to logit. The domain classifier is a three-layer MLP with hidden dimension 1024, aiming at predicting domain labels. We implement T-ADV on downstream tasks using ImageNet pre-trained DeiT backbones. We train models with learning rate lr = 5e-5 and weight decay = 1e-7 for 10000 iters except that the domain classifier use a learning rate lr = 2.5e-4. Furthermore, we make some adjustments to the training scheme due to the special architectures of Vision Transformers. Based on our practice, we restrict the magnitude of gradient reverse coefficient λ_{adv} by further multiplying 0.1 from the original setting, to avoid great fluctuation during training. These adjustments contribute to a stable adversarial training process.
- **T-MME.** T-MME consists of a feature encoder and a cosine similarity-based classifier. The classifier is implemented by a three-layer MLP with a hidden size of 1024. The learning rate of this part is scaled up 10 times, which is consistent with the original setting. We make further adjustments on DeiT architectures by introducing the adaptive update scheme in [8] on the coefficient λ_e from 0 to 0.1, instead of the original constant setting. The other training setups are consistent with T-ADV.
- **T-SSL.** T-SSL consists of a feature encoder and a cosine similarity-based classifier. The output of the feature encoder is linearly embedded into 512-d and then ℓ_2-normalized. The normalized vectors are used for k-means clustering and label prediction. We train models with learning rate lr = 5e-5 and weight decay = 1e-7 for 10000 iters. The balancing coefficient λ_{mim} is constantly assigned to 0.5 and λ_{is} is adaptively updated from 0 to 0.1 using the scheme in [8].
- **BiT-DANN** BiT-DANN consists of a feature encoder, a label predictor, and a domain classifier. The domain classifier has the same architecture as the one in T-ADV. Similarly, we implement BiT-DANN on downstream tasks using ImageNet pre-trained BiT backbones. We train models with learning rate lr = 3e-3 and weight decay = 5e-4 for 10000 iters except that the domain classifier use a learning rate lr = 0.015.
- **BiT-MME** BiT-MME consists of a feature encoder and a cosine similarity-based classifier. The classifier has the same architecture as the one in T-ADV. The learning rate of this part is also scaled up 10 times from the base set. The coefficient λ_e is a constant value of 0.1. The other training setups are consistent with BiT-DANN.

Table 2: **Configurations of the used model architectures.**

Model	patch size	embedding dimension	#heads	#layers	#params
DeiT-Ti/16	16×16	192	3	12	5M
DeiT-S/16	16×16	384	6	12	22M
DeiT-B/16	16×16	768	12	12	86M
DeiT-B/32	32×32	768	12	12	86M
DeiT-L/16	16×16	1024	16	24	307M
BiT-S-R50X1	-	-	-	-	23M

- **BiT-SSL** The architecture of BiT-SSL is consistent with T-SSL. except the backbone is replaced by a BiT model. We train models with learning rate lr = 0.01 for 10000 iters. The balancing coefficient λ_{mim} and λ_{is} is respectively assigned to 0.5 and 1.0.

C Measure Distribution Shifts via Proxy $\mathcal{A}$-distance

Here we introduce the **Proxy $\mathcal{A}$-distance (PAD)** used by Ganin et al. [9]. Given two distributions $\mathscr{D}_a$ and $\mathscr{D}_b$, a learning algorithm is introduced to discriminate between examples from two distributions. A new data set is constructed as

$$\mathscr{U} = \{(\mathrm{x}_i^a, 0)\}_{i=1}^{N_a} \cup \{(\mathrm{x}_i^b, 1)\}_{i=1}^{N_b}, \tag{6}$$

where samples x_i^a from $\mathscr{D}_a$ are labeled 0 and samples x_i^b from $\mathscr{D}_b$ are labeled 1. N_a and N_b represent the number of samples from $\mathscr{D}_a$ and $\mathscr{D}_b$ respectively. Then, with the generalization error ϵ of the classifier trained on $\mathscr{U}$, we could calculate the Proxy $\mathcal{A}$-distance by

$$\hat{d}_{\mathcal{A}} = 2(1 - 2\epsilon). \tag{7}$$

In practice, we first train a simple CNN model on a subset of $\mathscr{U}$. This CNN model only consists of four 3x3 convolution layers and four max-pooling layers. Then we use the obtained classifier error on the other subset as the value of ϵ in Eq. 7.

D More Experiment Results

D.1 Ablation Study on Self-supervised Generalization Enhancing Method

Since the self-supervised learning method T-SSL and BiT-SSL contains two major loss terms $\mathcal{L}_{\mathrm{IS}}$ and $\mathcal{L}_{\mathrm{MIM}}$ in enhancing out-of-distribution generalization, we separately test their effectiveness. As the results are shown in Tab. 3, we could observe that 1) $\mathcal{L}_{\mathrm{MIM}}$ works the best for VGG-16, 2) but the combination of two parts perform the best on average for larger models including DeiT-S/16 and BiT. Thus, we could conclude that there exists a mutual promotion between the in-domain self-supervision and mutual information maximization towards large models.

D.2 Generalization-Enhanced ViTs results under multiple shifts

We examine the effectiveness of the generalization-enhanced methods under multiple shifts, including corruption shifts, texture shifts, and style shifts. We respectively use ImageNet-C, Stylized-ImageNet, and ImageNet-R for experiments. Because of the lack of training sets, we make a 2:1 split on these benchmarks and the ImageNet validation set for training and testing. Specifically, we use severity 3 of corruptions for use. The results are shown in Tab. 4, 5, 6 and 7. From the results we could observe that 1) MME dominates the results under corruption shift for both types of models, 2) T-SSL performance the best under background shifts while DANN works the best for BiT models. 3) these generalization-enhancing methods may be harmful to generalization under certain distribution shifts for BiTs, e.g. defocus and brightness, while having little infection on DeiTs.

Table 3: **Ablation study results on self-supervised generalization enhancing method.**

Model	Method	R to C	R to P	P to C	C to S	S to P	R to S	P to R	Avg.
DeiT-S/16	-	50.60	45.82	36.09	43.39	35.24	39.29	52.08	43.22
	T-MIM	53.67	44.80	42.31	47.00	**43.28**	**41.70**	**58.38**	47.30
	T-IS	53.31	**47.19**	40.93	46.71	41.93	38.77	55.62	46.35
	T-SSL	**53.86**	46.71	**42.79**	**47.25**	43.01	40.94	57.07	**47.37**
BiT	-	42.18	41.14	30.72	37.01	28.23	32.64	48.54	36.78
	MIM [31]	**53.02**	41.64	**40.24**	45.10	38.26	**40.17**	54.70	44.73
	IS [31]	48.31	**44.12**	36.32	43.84	38.30	35.81	53.31	42.85
	SSL [31]	52.55	42.80	39.03	**45.72**	**39.08**	39.65	**56.07**	**44.98**
VGG-16	-	39.39	37.32	26.36	32.96	25.55	27.79	45.70	33.58
	MIM [31]	**48.41**	42.18	**36.34**	**43.08**	**38.45**	**37.51**	54.32	**42.89**
	IS [31]	42.05	**42.36**	31.30	38.68	36.59	30.74	51.07	38.97
	SSL [31]	43.79	41.88	32.19	35.73	36.99	31.05	**55.18**	39.54

Table 4: **Results of generalization-enhanced methods under corruption shifts including noises and blurs.**

Model	Method	Noise				Blur				
		Gauss.	Impulse	Shot	Avg.	Defocus	Glass	Motion	Zoom	Avg.
DeiT-S/16	-	55.51	54.70	54.74	54.98	45.11	27.82	51.51	41.51	41.49
	T-ADV	58.96	62.35	60.02	60.44	**55.95**	54.94	62.38	62.07	58.84
	T-MME	**60.40**	**63.85**	**60.96**	**61.74**	55.86	**56.02**	**63.69**	**63.35**	**59.73**
	T-MIM	59.10	62.17	59.37	60.21	53.97	54.39	62.00	61.46	57.96
	T-IS	39.78	41.18	39.67	40.21	32.87	21.48	41.59	34.16	32.52
	T-SSL	56.54	60.26	56.99	57.93	50.24	51.56	59.68	59.04	55.13
BiT	-	37.48	33.46	34.70	35.21	22.62	10.04	30.54	31.44	23.66
	DANN [8]	37.21	41.08	39.85	39.38	**30.55**	28.54	42.70	40.60	**35.60**
	MME [23]	**43.51**	**46.14**	**43.81**	**44.49**	15.07	**30.75**	**47.17**	**42.82**	33.95
	MIM [31]	36.23	36.33	35.81	36.12	23.95	25.59	40.11	34.68	31.08
	IS [31]	18.33	17.24	17.04	17.54	9.77	6.07	15.41	17.33	12.15
	SSL [31]	35.00	35.28	34.95	35.08	25.90	26.19	36.42	20.15	27.16

Table 5: **Results of generalization-enhanced methods under corruption shifts including weathers and digital corruptions.**

Model	Method	Weather					Digital				
		Bright	Fog	Frost	Snow	Avg.	Contrast	Elastic	JPEG	Pixel	Avg.
DeiT-S/16	-	72.11	53.67	50.45	53.79	57.51	66.58	63.10	61.25	60.51	62.86
	T-ADV	71.54	67.07	59.81	66.43	66.21	67.73	69.83	64.43	68.25	67.56
	T-MME	71.93	**67.62**	**60.57**	**67.03**	**66.79**	**68.44**	70.04	**65.65**	**68.92**	**68.26**
	T-MIM	**72.37**	66.98	59.76	66.38	66.37	68.22	**70.19**	64.39	68.56	67.84
	T-IS	66.50	51.91	39.62	51.65	52.42	56.06	58.77	53.53	50.07	54.60
	T-SSL	70.85	65.42	57.04	64.80	64.53	66.38	68.83	63.71	66.43	66.34
BiT	-	**65.96**	49.08	32.23	34.68	45.49	54.57	44.61	52.74	48.27	50.04
	DANN [8]	60.88	54.67	37.95	48.10	50.40	55.75	56.12	48.52	54.57	53.74
	MME [23]	61.94	**57.05**	**41.45**	**52.43**	**53.22**	**57.46**	**57.93**	**51.28**	**56.61**	**55.82**
	MIM [31]	60.38	56.37	34.66	48.92	50.08	56.29	57.17	50.55	55.92	54.98
	IS [31]	49.45	34.50	17.77	22.01	30.93	35.03	32.27	32.19	28.44	31.98
	SSL [31]	54.78	48.88	34.25	42.20	45.03	49.26	49.54	44.67	15.68	39.79

Table 6: **Results of generalization-enhanced methods under background shifts.**

Model	Method	Background				
		Only-FG	Mixed-Same	Mixed-Rand	Mixed-Next	Avg.
DeiT-S/16	-	88.80	90.21	84.08	82.70	86.44
	T-ADV	94.26	96.54	92.79	92.86	94.11
	T-MME	93.89	96.39	92.13	91.69	93.52
	T-MIM	**95.51**	96.83	92.72	92.42	84.37
	T-IS	95.36	96.17	88.67	87.57	91.94
	T-SSL	94.92	**96.98**	**93.01**	**93.16**	**94.52**
BiT	-	82.04	83.81	77.07	74.19	79.27
	DANN [8]	**94.19**	**95.44**	**91.25**	**91.76**	**93.16**
	MME [23]	87.57	92.75	83.72	87.27	87.83
	MIM [31]	94.04	95.29	90.07	90.73	92.53
	IS [31]	85.07	92.42	82.50	83.82	85.95
	SSL [31]	93.67	95.14	88.82	90.22	91.96

Table 7: **Results of generalization-enhanced methods under texture shifts and style shifts.**

Model	Method	Texture	Style
		Stylized ImageNet	ImageNet-R
DeiT-S/16	-	13.11	25.91
	T-ADV	**27.16**	**48.97**
	T-MME	17.90	47.71
	T-MIM	15.79	46.60
	T-IS	9.49	37.08
	T-SSL	13.78	47.33
BiT	-	**6.05**	23.46
	DANN [8]	4.49	35.33
	MME [23]	5.33	31.84
	MIM [31]	3.99	37.79
	IS [31]	3.46	28.36
	SSL [31]	4.15	**38.29**