Domain Generalization by Mutual-Information Regularization with Pre-trained Models – Appendix

Junbum Cha[1], Kyungjae Lee[2], Sungrae Park[3], and Sanghyuk Chun[4]

[1] Kakao Brain [2] Chung-Ang University
[3] Upstage AI Research [4] NAVER AI Lab
junbum.cha@kakaobrain.com, kyungjae.lee@ai.cau.ac.kr,
sungrae.park@upstage.ai, sanghyuk.c@navercorp.com

A Derivation of Lower Bound

Assumption 1 *The variational distribution $q(\cdot|z)$ satisfies the regularity condition such that, for any $\mathbb{P}_{X|z} \in \{\mathbb{P}'_{X|z} \mid \mathbb{E}_{X|z}[|X|^2] < \infty\}$,*

$$\mathbb{E}_{X|z}\left[(\nabla_x \log q(x|z)|_{x=X})^\intercal \nabla_x \log q(x|z)|_{x=X}\right] < \infty, \tag{7}$$

where $\mathbb{E}_{X|z}$ is a conditional expectation of X given z.

Remark 1. Note that the Gaussian distribution used in our implementation satisfies the regularity condition. To check the regularity condition of Gaussian distribution, we first compute the gradient as follows,

$$\nabla_x \log q(x|z)|_{x=X} \tag{8}$$

$$= \nabla_x \left(C + \frac{1}{2}\log|\Sigma(z)| + \frac{1}{2}(x-\mu(z))^\intercal \Sigma(z)^{-1}(x-\mu(z)) \right)|_{x=X} \tag{9}$$

$$= \Sigma(z)^{-1}(X-\mu(z)). \tag{10}$$

Hence, we get,

$$\mathbb{E}_{X|z}\left[(\nabla_x \log q(x|z)|_{x=X})^\intercal \nabla_x \log q(x|z)|_{x=X}\right] \tag{11}$$

$$= \mathbb{E}_{X|z}\left[(X-\mu(z))^\intercal \Sigma(z)^{-2}(X-\mu(z))\right] < \infty. \tag{12}$$

since $\mu(z)$ and $\Sigma(z)$ are finite and $\mathbb{E}_{X|z}[|X|^2]$ is bounded. Hence, the Gaussian distribution satisfies the regularity condition.

Under the assumption of q, we derive the lower bound.

Proof (Derivation of the Lower Bound). Based on the regularity condition, we derive the lower bound of the term, $\mathbb{E}_{Z_{f^*},Z_f}[\log q(Z_{f^*} \mid Z_f)]$. Before starting the derivation, let us define $d_{2,\infty}(f,g) := \sup_x \|f(x)-g(x)\|_2$. Then, the derivation starts from Taylor's theorem for a differentiable multivariate function. From

Taylor's theorem, there exists a point c such that $c = tx + (1-t)x_0$ for some $t \in [0,1]$ and the following equality holds,

$$\log q(x \mid y) = \log q(x_0 \mid y) + \nabla_x \log q(x \mid y)|_{x=c}^{\intercal}(x - x_0). \tag{13}$$

Then, we can derive the following upper bound as follows,

$$\begin{aligned}
\log q(x \mid y) &= \log q(x_0 \mid y) + \nabla_x \log q(x \mid y)|_{x=c}^{\intercal}(x - x_0) && (14)\\
&\leq \log q(x_0 \mid y) + |\nabla_x \log q(x \mid y)|_{x=c}^{\intercal}(x - x_0)| && (15)\\
&\leq \log q(x_0 \mid y) + \|\nabla_x \log q(x \mid y)|_{x=c}\|_2 \, \|x - x_0\|_2 && (16)
\end{aligned}$$

By using this bound, we can derive the following lower bound,

$$\begin{aligned}
&\mathbb{E}_{Z_{f^*},Z_f}[\log q(Z_{f^*} \mid Z_f)] = \mathbb{E}_{X,X'}\left[\log q(f^*(X) \mid f(X'))\right] && (17)\\
&\quad \geq \mathbb{E}_{X,X'}\left[\log q(f^0(X) \mid f(X'))\right]\\
&\qquad - \mathbb{E}_{X,X'}\left[\|\nabla \log q(c(X) \mid f(X'))\|_2 \left\|f^0(X) - f^*(X)\right\|_2\right] && (18)\\
&\quad \geq \mathbb{E}_{X,X'}\left[\log q(f^0(X) \mid f(X'))\right]\\
&\qquad - \mathbb{E}_{X,X'}\left[\|\nabla \log q(c(X) \mid f(X'))\|_2\right] d_{2,\infty}(f^*, f^0) && (19)\\
&\quad \geq \mathbb{E}_{Z_{f^0},Z_f}\left[\log q(Z_{f^0} \mid Z_f)\right] - C d_{2,\infty}(f^*, f^0), && (20)
\end{aligned}$$

where $c(x)$ is the function between f^0 and f^*, which selects the point satisfying Taylor's theorem, and C is a constant derived from the regularity condition.

B Additional Implementation Details

B.1 Hyperparameter tuning

We split the hyperparameters (HPs) into two groups: algorithm-specific HPs and algorithm-agnostic HPs. The algorithm-agnostic HPs consist of batch size, learning rate, dropout, and weight decay, and MIRO has only one algorithm-specific HP, λ. To reduce the computational cost, we tune the algorithm-specific HPs and algorithm-agnostic HPs independently. We first search algorithm-specific HPs with default algorithm-agnostic HPs, then search algorithm-agnostic HPs with the tuned algorithm-specific HPs. That is, the λ is searched in [1.0, 0.1, 0.01, 0.001] with the batch size of 32, the learning rate of 5e-5, no dropout, and no weight decay. Then, we search algorithm-agnostic HPs with the searched λ following Cha *et al.* [3]. They propose reduced HP search space for efficiency compared to DomainBed [5]. The protocol searches the learning rate in [1e-5, 3e-5, 5e-5], dropout in [0.0, 0.1, 0.5], and weight decay in [1e-4, 1e-6]. The batch size per domain is fixed to 32. Since MIRO is a regularization method, we add a case of no weight decay.

Even though we use the efficient HP search protocol, it still requires heavy computational resources. Therefore, we tune λ only for the non-main experiments, including combination with SWAD, combination with various pre-trained backbones, and the case study on `Camelyon17`. Also, we use the batch size of 16 for SWAG [10] due to the GPU memory limitation. Note that there is room for further performance improvement by intensive HP tuning and additional usage of GPU memory, considering the simplified HP search protocol and limited computational resources.

B.2 Implementation details

The variance encoder is initialized to estimate the variance of 0.1. It is chosen by observing the convergence point of the variance. Softplus function is employed to ensure non-negativity of the variance. Also, we empirically apply the 10 times larger learning rate for the mean and variance encoders than the feature extractor and the classifier.

B.3 Mutual information estimation

In Section 2.2, we estimate the mutual information using Mutual Information Neural Estimator (MINE) [1]. The mutual information is estimated by MINE as follows:

$$\widehat{I(Z_{f^*}; Z_f)} = \sup_{\theta \in \Theta} \mathbb{E}_{\mathbb{P}_{Z_{f^*} Z_f}} [T_\theta] - \log \left(\mathbb{E}_{\mathbb{P}_{Z_{f^*}} \otimes \mathbb{P}_{Z_f}} \left[e^{T_\theta} \right] \right). \tag{21}$$

For the features Z_{f^*} and Z_f, the features after global average pooling are uniformly collected by domains. The statistics network, T_θ, consists of two hidden linear layers with 512 dimensions and ELU activation functions, following [1]. In the case of fine-tuning, such as ERM−, ERM+, and MIRO, the models are trained as many as the number of target domains. Therefore, we estimate the mutual information for each model and report their average value.

C Additional Analysis and Discussion

C.1 Variations on the assumptions of domain generalization

Table 5: **Performances of class-conditional MIRO.**

Algorithm	`PACS`	`VLCS`	`OfficeHome`	`TerraInc`	**Avg.**
ERM	84.2±0.1	77.3±0.1	67.6±0.2	47.8±0.6	69.2
MIRO	**85.4**±0.4	**79.0**±0.0	70.5±0.4	**50.4**±1.1	**71.3**
C-MIRO	85.3±0.5	78.5±0.5	**70.8**±0.3	49.4±0.3	71.0

In general, domain generalization (DG) assumes that there are multiple source domains, source domain labels are available, and the same input has the same label between source and target domains. Here, we can make the variations on the problem settings by changing the assumption. Single-source DG does not assume the multiple source domains [3, 4]. Several studies try to solve DG problem without domain labels [2, 3, 8]. Heterogeneous DG deals with the label set shift, *i.e.*, the same input can have different labels between source and target domains [7, 11]. In this task, it is assumed that a classifier is learnable in the target domain and the methods focus on the feature extractor. The proposed method exploits pre-trained models instead of assuming available multiple source domains or source domain labels, and focuses on the feature extractor instead of the classifier. Therefore, MIRO is directly applicable to single-source DG, DG without domain labels, and heterogeneous DG problems. On the other hand, we can consider a more specific type of distribution shift. In this case, we may need a different mutual information (MI) strategy. For example, we can employ class-conditional MI for class-conditional distribution shift (C-MIRO) by using $I(Z_{f^*}; Z_f|Y)$ instead of $I(Z_{f^*}; Z_f)$. In Table 5, C-MIRO achieves comparable scores with MIRO and outperforms ERM even though the problem setting is not class-conditional. From the results, we believe that MIRO can be adapted to other distribution shifts by choosing the proper MI strategy.

C.2 The relationship between mutual information and domain generalization performance

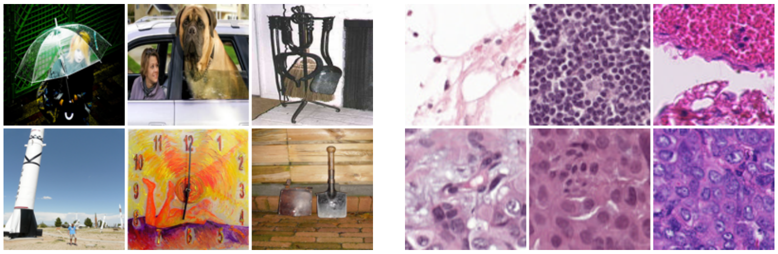
Table 6: **Average accuracies of ERM−, ERM+, and MIRO in PACS.** ERM− and ERM+ indicate ERM without and with pre-trained model, respectively.

Pre-trained model	**ERM−**	**ERM+**	**MIRO**
ResNet-50 (ImageNet)	51.6	84.2	85.4
RegNet-16GF (Instagram-3.6B)	51.5	89.6	97.4

Our method assumes that knowledge of the oracle model helps domain generalization and it is transferable to the target model by maximizing mutual information (MI). These assumptions are quite intuitive, but there is no theoretical guarantee that MI with the oracle model is directly correlated with DG performance. Empirically, we observe that a high MI model shows better DG performance if the empirical loss constraint of Equation (2) holds; the pre-trained model itself has high MI but does not satisfy this constraint. In the main text, Figure 1 shows the rankings of MI for ERM−, ERM+, and MIRO are in order. Table 6 shows that the rankings are the same for accuracies: ERM− (51.6%), ERM+ (84.2%), and MIRO (85.4%) in ImageNet pre-trained ResNet and ERM− (51.5%), ERM+ (89.6%), and MIRO (97.4%) in Instagram-3.6B pre-trained RegNet, respectively.

D Additional Results

D.1 Visual comparison between ImageNet and `Camelyon17`



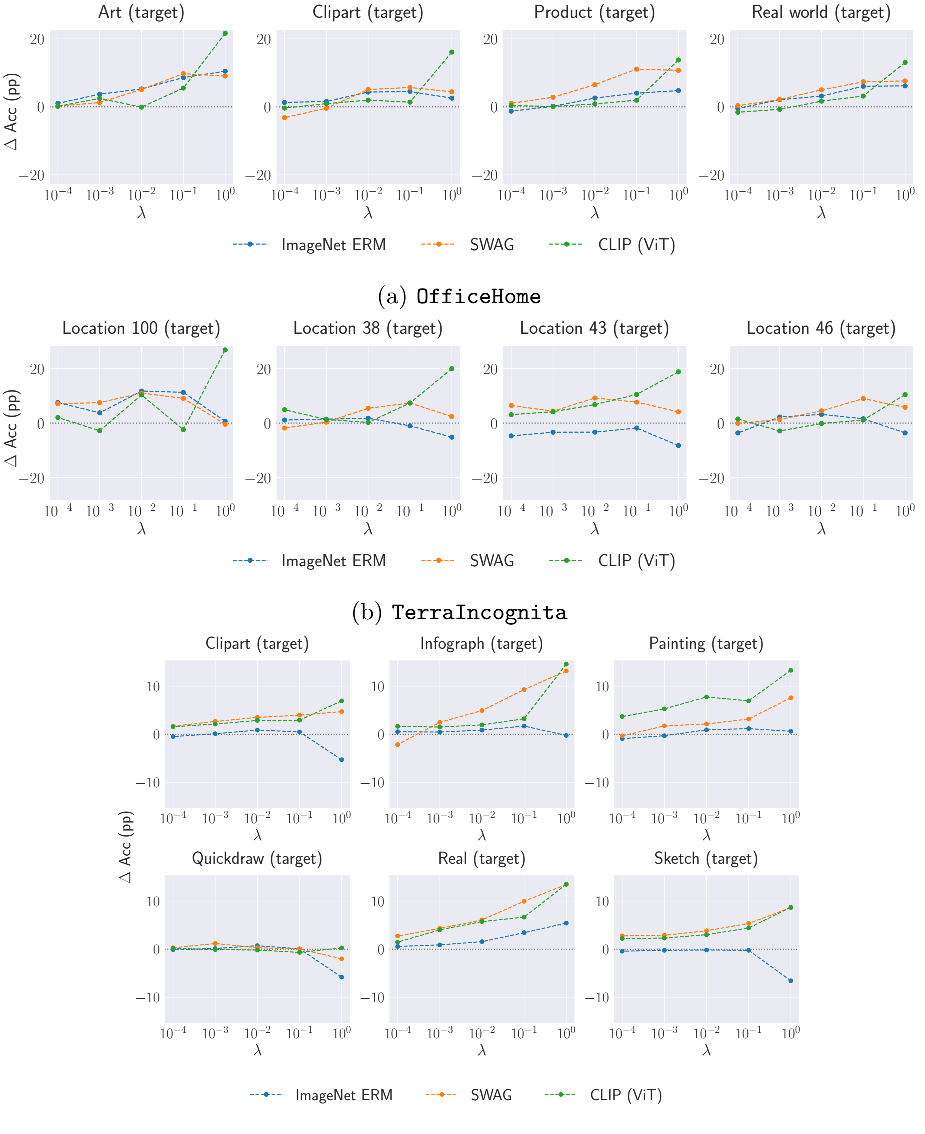
(a) ImageNet (pre-train) (b) `Camelyon17` (fine-tuning)

Fig. 4: **Example images of ImageNet and `Camelyon17`.** Large distribution shift occurs between pre-training (ImageNet) and fine-tuning (`Camelyon17`). ImageNet is a multiclass objective recognition task and `Camelyon17` is a binary classification task for reading whether the image contains tumor tissue. Instagram-3.6B examples are omitted since it is not publicly available.

Figure 4 shows a huge visual gap between pre-training (ImageNet) and fine-tuning (`Camelyon17`) datasets. The tasks are also different; ImageNet is an object recognition task and `Camelyon17` is a binary classification of breast cancer. Despite the large gap between pre-training and fine-tuning distribution, the proposed method shows consistent performance improvement (See Table 4 in the main text).

D.2 Relationship between the pre-training scale and the intensity of the mutual information regularization

In this section, we provide the extended results of Figure 3 in the main text. Figure 5 shows the additional comparison of three pre-trained backbones according to λ about `OfficeHome`, `TerraIncognita`, and `DomainNet`. The comparisons show similar trends with the results in `PACS`. ImageNet pre-trained backbone, such as ResNet-50 pre-trained in ImageNet [6], has a negative correlation between the performance difference and λ in some target domains. Large-scale pre-trained backbones, such as SWAG [10] and CLIP [9], tend to consistently make significant performance improvements at high λ and become less sensitive to the choice of λ.



(a) `OfficeHome`

(b) `TerraIncognita`

(c) `DomainNet`

Fig. 5: **Comparison of three pre-trained models according to λ.** Y-axis indicates the performance difference of MIRO to ERM. λ is the intensity of the mutual information regularization. We compare three models: ResNet-50 pre-trained in ImageNet [6], RegNetY-16GF pre-trained by SWAG [10], and ViT-B pre-trained by CLIP [9].

References

1. Belghazi, M.I., Baratin, A., Rajeshwar, S., Ozair, S., Bengio, Y., Courville, A., Hjelm, D.: Mutual information neural estimation. In: International Conference on Machine Learning (2018)
2. Carlucci, F.M., D'Innocente, A., Bucci, S., Caputo, B., Tommasi, T.: Domain generalization by solving jigsaw puzzles. In: Computer Vision and Pattern Recognition (2019)
3. Cha, J., Chun, S., Lee, K., Cho, H.C., Park, S., Lee, Y., Park, S.: Swad: Domain generalization by seeking flat minima. In: Neural Information Processing Systems (2021)
4. Fan, X., Wang, Q., Ke, J., Yang, F., Gong, B., Zhou, M.: Adversarially adaptive normalization for single domain generalization. In: Computer Vision and Pattern Recognition (2021)
5. Gulrajani, I., Lopez-Paz, D.: In search of lost domain generalization. In: International Conference on Learning Representations (2021)
6. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Computer Vision and Pattern Recognition (2016)
7. Li, Y., Yang, Y., Zhou, W., Hospedales, T.: Feature-critic networks for heterogeneous domain generalization. In: International Conference on Machine Learning (2019)
8. Matsuura, T., Harada, T.: Domain generalization using a mixture of multiple latent domains. In: AAAI Conference on Artificial Intelligence (2020)
9. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International Conference on Machine Learning (2021)
10. Singh, M., Gustafson, L., Adcock, A., de Freitas Reis, V., Gedik, B., Kosaraju, R.P., Mahajan, D., Girshick, R., Dollár, P., van der Maaten, L.: Revisiting weakly supervised pre-training of visual perception models. In: Computer Vision and Pattern Recognition (2022)
11. Wang, Y., Li, H., Kot, A.C.: Heterogeneous domain generalization via domain mixup. In: International Conference on Acoustics, Speech and Signal Processing (2020)