1 Additional details for Section 3

1.1 Proof of Theorem 1.

Definition 1. *(Mutual Information Sufficient Encoder). Let f_{op} be a mutual information sufficient encoder. For positive pairs $(\mathbf{x}_i, \mathbf{x}_j)$, there are $(\mathbf{z}_i^*, \mathbf{z}_j^*)$=$f_{op}((\mathbf{x}_i, \mathbf{x}_j))$*

$$I(\mathbf{x}_i; \mathbf{x}_j) = I(\mathbf{z}_i^*; \mathbf{z}_j^*) \geq I(\mathbf{z}_i; \mathbf{z}_j) \tag{1}$$

Different from the general deep feature encoder, the mutual information sufficient encoder obtains the optimal value by selecting the most suitable positive instance pairs. Namely,

$$I(\mathbf{z}_i^*; \mathbf{x}_i) = I(\mathbf{x}_i; \mathbf{x}_j) = I(\mathbf{z}_i^*; \mathbf{z}_j^*) \tag{2}$$

Since the supervised information of the feature extraction network trained using contrastive learning comes entirely from another positive instances. This definition is built on the assumption that the feature extractor is ideal, namely, the feature extraction process is lossless. The representations $\mathbf{z}_i, \mathbf{z}_j$ are built from positive pair $\mathbf{x}_i, \mathbf{x}_j$ and learned by the contrastive objective with the assumption of mutual information sufficient encoder.

Theorem 1 *Suppose f_{op} is a mutual information sufficient encoder, Y and D are independent (See Section 3.1 for definition). For tasks that generalize to arbitrary given unknown domain, the optimal positive pairs $(\mathbf{x}_i^{k_i,y_i}, \mathbf{x}_j^{k_j,y_j})$ selection are:*

$$(\mathbf{x}_i^*, \mathbf{x}_j^*) = \underset{\mathbf{x}_i,\mathbf{x}_j}{\operatorname{argmin}} -\kappa I(\mathbf{x}_i; \mathbf{x}_j; Y) + I(\mathbf{x}_i; \mathbf{x}_j; D) + I(\mathbf{x}_i; \mathbf{x}_j | Y, D) \tag{3}$$

Proof. According to the information bottleneck theory, we can find the optimal assignment by minimizing the function:

$$\min I(Z; X) - (\kappa + 1) I(Z; Y) \tag{4}$$

where, $(\kappa + 1)$ controls the trade-off, depending on the expression ability of the model. When the model has powerful expression ability to fully represent the Y information in X, $(\kappa + 1)$ is greater than 1. By definition of the mutual information sufficient encoder assumptions, the model can fully represent the Y label information in $(\mathbf{x}_i, \mathbf{x}_i)$. Therefore, we have $\kappa > 0$.

Then, under the condition of f_{op} is a mutual information sufficient encoder and $Y \perp\!\!\!\perp D$ conditioned on $\forall \mathbf{z}_i \in \mathcal{Z}$, for an arbitrary positive pair $(\mathbf{x}_i^{k_i,y_i}, \mathbf{x}_j^{k_j,y_j})$:

$$\begin{aligned}
&\arg\min_{\mathbf{x}_i,\mathbf{x}_j} I(\mathbf{z}_i^*; \mathbf{x}_i) - (\kappa + 1) I(\mathbf{z}_i^*; Y) \\
&= \arg\min_{\mathbf{x}_i,\mathbf{x}_j} I(\mathbf{x}_i; \mathbf{x}_j) - (\kappa + 1) I(\mathbf{x}_i; \mathbf{x}_j; Y) \\
&= \arg\min_{\mathbf{x}_i,\mathbf{x}_j} I(\mathbf{x}_i; \mathbf{x}_j; Y) + I(\mathbf{x}_i; \mathbf{x}_j; D) + I(\mathbf{x}_i; \mathbf{x}_j | Y, D) - (\kappa + 1) I(\mathbf{x}_j; \mathbf{x}_i; Y) \\
&= \arg\min_{\mathbf{x}_i,\mathbf{x}_j} I(\mathbf{x}_i; \mathbf{x}_j; D) + I(\mathbf{x}_i; \mathbf{x}_j | Y, D) - \kappa I(\mathbf{x}_i; \mathbf{x}_j; Y)
\end{aligned} \tag{5}$$

And we find that:

$$I_{(i,j)\in S_{\text{pos+}}} I(\mathbf{x}_i; \mathbf{x}_j; D) > I_{(i,j)\in S_{\text{pos-}}} (\mathbf{x}_i; \mathbf{x}_j; D) \quad \text{almost surely,} \tag{6}$$

by the definition of the domain. Given the domain information D, instances within the same domain are more similar (as we show the supervised learning experiments in the introduction section) and have a higher degree of information association, leading to a larger mutual information value.

Finally, we reach the conclusion of the Theorem 1.

1.2 Proof of Theorem 2.

Theorem 2 *Let N be the batch size, the adjustable constant $C_w = \eta(2N - 2)$. Assuming the number of instances per subdomain is similar. As $\eta, N \to \infty$, with respect to arbitrary anchor $\mathbf{x}_i \in \mathcal{X}$, the expectation of ℓ_{ms} converges to:*

$$\begin{aligned}
\lim_{\eta,N\to\infty} \mathbb{E}[\ell_{ms}] = \log \mathbb{E}_{(i,l^+)\in S_{neg+}} \left[\exp\left(sim(\mathbf{z}_i, \mathbf{z}_{l^+}) \right) \right] \\
-\frac{1}{\tau} \mathbb{E}_{(i,j)\in S_{pos-}} \left[sim(\mathbf{z}_i, \mathbf{z}_j) \right] + \log C_w
\end{aligned} \tag{7}$$

Proof. To begin with, under the condition of the number of instances per subdomain is similar, for $\forall i$, we note that:(As defined in Section 3.1 of the main text, there are C classes and K domains)

$$\mathbb{E}[m_0(i)] = \frac{2N}{C},\ \mathbb{E}[m_1(i)] = \frac{2N(C-1)(K-1)}{C \cdot K},\ \mathbb{E}[m_2(i)] = \frac{2N(C-1)}{C \cdot K} \tag{8}$$

Using random uniform sampling to obtain the batch, for the anchor instance i, we have:

$$\begin{aligned}
&\lim_{\eta,N\to\infty} \mathbb{E}\big[\ell_{\text{ms}}\big] = \lim_{\eta,N\to\infty} \mathbb{E}\Big[-\log\big[\exp(\text{sim}(\mathbf{z}_i,\mathbf{z}_j)/\tau)\big] + \log\big[\sum_{l=1}^{2N}\mathbf{w}(i,l)\exp(\text{sim}(\mathbf{z}_i,\mathbf{z}_l)/\tau)\big]\Big] \\
&\overset{(1)}{=} -\frac{1}{\tau}\mathbb{E}_{(i,j)\in S_{\text{pos-}}}\big[\text{sim}(\mathbf{z}_i,\mathbf{z}_j)\big] + \lim_{\eta,N\to\infty}\log\mathbb{E}\big[\sum_{(i,l')\in S_{\text{neg-}}}\exp(\text{sim}(\mathbf{z}_i,\mathbf{z}_{l'})/\tau) + \sum_{(i,l^+)\in S_{\text{neg+}}}\frac{C_{\text{w}}\text{-}m_1(i)}{m_2(i)}\exp(\text{sim}(\mathbf{z}_i,\mathbf{z}_{l^+})/\tau)\big] \\
&\overset{(2)}{=} -\frac{1}{\tau}\mathbb{E}_{(i,j)\in S_{\text{pos-}}}\big[\text{sim}(\mathbf{z}_i,\mathbf{z}_j)\big] + \lim_{\eta,N\to\infty}\log\big[m_1(i)\mathbb{E}_{(i,l')\in S_{\text{neg-}}}\exp(\text{sim}(\mathbf{z}_i,\mathbf{z}_{l'})/\tau) + \big(C_{\text{w}}\text{-}m_1(i)\big)\mathbb{E}_{(i,l^+)\in S_{\text{neg+}}}\exp(\text{sim}(\mathbf{z}_i,\mathbf{z}_{l^+})/\tau)\big] \\
&\overset{(3)}{=} -\frac{1}{\tau}\mathbb{E}_{(i,j)\in S_{\text{pos-}}}\big[\text{sim}(\mathbf{z}_i,\mathbf{z}_j)\big] + \lim_{\eta,N\to\infty}\log\mathbb{E}_{(i,l^+)\in S_{\text{neg+}}}\exp(\text{sim}(\mathbf{z}_i,\mathbf{z}_{l^+})/\tau) + \log\big[C_{\text{w}}\text{-}m_1(i)\big] \\
&\overset{(4)}{=} -\frac{1}{\tau}\mathbb{E}_{(i,j)\in S_{\text{pos-}}}\big[\text{sim}(\mathbf{z}_i,\mathbf{z}_j)\big] + \log\mathbb{E}_{(i,l^+)\in S_{\text{neg+}}}\big[\exp\big(\text{sim}(\mathbf{z}_i,\mathbf{z}_{l^+})/\tau\big)\big] + \log C_{\text{w}}
\end{aligned} \tag{9}$$

where, (1-2) hold due to the continuous mapping theorem, (3) holds because the law of large numbers, and (4) holds because the property of concave function $\log(x)$. Note that the second term of Eq.(7) is independent of the value of N and η. And in the same way, for conventional contrastive loss ℓ_{cl} there is

$$\mathbf{w}'(i,j) = \mathbf{M}_{\text{neg-}}(i,j) + \mathbf{M}_{\text{neg+}}(i,j) + \mathbf{M}_{\text{pos}}(i,j) \tag{10}$$

The expectation of ℓ_{cl} is:

$$\begin{aligned}
&\lim_{N\to\infty}\mathbb{E}\big[\ell_{\text{cl}}\big] = \lim_{N\to\infty}\mathbb{E}\Big[-\log\big[\exp(\text{sim}(\mathbf{z}_i,\mathbf{z}_j)/\tau)\big] + \log\big[\sum_{l=1}^{2N}\mathbf{w}'(i,l)\exp(\text{sim}(\mathbf{z}_i,\mathbf{z}_l)/\tau)\big]\Big] \\
&= -\frac{1}{\tau}\mathbb{E}_{(i,j)\in S_{\text{pos}}}\big[\text{sim}(\mathbf{z}_i,\mathbf{z}_j)\big] + \mathbb{E}\log\big[\sum_{(i,l')\in S_{\text{neg-}}}\exp(\text{sim}(\mathbf{z}_i,\mathbf{z}_{l'})/\tau) + \sum_{(i,l^+)\in S_{\text{neg+}}}\exp(\text{sim}(\mathbf{z}_i,\mathbf{z}_{l^+})/\tau) + \sum_{(i,l')\in S_{\text{pos}}}\exp(\text{sim}(\mathbf{z}_i,\mathbf{z}_{l'})/\tau)\big] \\
&= -\frac{1}{\tau}\mathbb{E}_{(i,j)\in S_{\text{pos}}}\big[\text{sim}(\mathbf{z}_i,\mathbf{z}_j)\big] + \log\mathbb{E}_{(i,l)\in S_{\text{neg-}}\cup S_{\text{neg+}}\cup S_{\text{pos}}}\big[\exp\big(\text{sim}(\mathbf{z}_i,\mathbf{z}_l)\big)\big] + \log\big[m_1(i) + m_2(i)\big]
\end{aligned} \tag{11}$$

Next, we check the convergence of the function ℓ_{ms}. Using Chebyshev's inequality, we have

$$Pr(S_{\text{neg+}},\epsilon) = P\Big(\Big|\sum_{(i,l^+)\in S_{\text{neg+}}}\exp(\text{sim}(\mathbf{z}_i,\mathbf{z}_{l^+})/\tau) - \mathbb{E}_{(i,l^+)\in S_{\text{neg+}}}\big[\exp\big(\text{sim}(\mathbf{z}_i,\mathbf{z}_{l^+})\big)\big]\Big| \le \epsilon\Big) \ge 1 - \frac{Var_{(i,l^+)\in S_{\text{neg+}}}[\exp(\text{sim}(\mathbf{z}_i,\mathbf{z}_{l^+}))]}{m_2^2(i)\cdot\epsilon^2} \tag{12}$$

And $\exp[\text{sim}(\mathbf{u},\mathbf{v})/\tau]$ is belong to the close interval $[e^{-1/\tau}, e^{1/\tau}]$, and $Var\big[\exp[\text{sim}(\mathbf{u},\mathbf{v})/\tau]\big] \le \dfrac{e^{1/\tau} - e^{-1/\tau}}{4}$ also hold.

$$\begin{aligned}
&\lim_{\eta,N\to\infty}\mathbb{E}\big[\ell_{\text{ms}}\big] - \mathbb{E}\big[\ell_{\text{ms}}\big] \\
&= \mathbb{E}\Big[\log\mathbb{E}_{(i,l^+)\in S_{\text{neg+}}}\big[\exp\big(\text{sim}(\mathbf{z}_i,\mathbf{z}_{l^+})/\tau\big)\big] - \Big[\log\big[\sum_{(i,l')\in S_{\text{neg-}}}\exp(\text{sim}(\mathbf{z}_i,\mathbf{z}_{l'})/\tau) + \sum_{(i,l^+)\in S_{\text{neg+}}}\frac{C_{\text{w}}\text{-}m_1(i)}{m_2(i)}\exp(\text{sim}(\mathbf{z}_i,\mathbf{z}_{l^+})/\tau)\big] - \log C_{\text{w}}\Big]\Big] \\
&\overset{(1)}{\le} \Big(\sum_{(i,l')\in S_{\text{neg-}}}\frac{1}{C_{\text{w}}}\exp(\text{sim}(\mathbf{z}_i,\mathbf{z}_{l'})/\tau) + \sum_{(i,l^+)\in S_{\text{neg+}}}\frac{C_{\text{w}}\text{-}m_1(i)}{C_{\text{w}}\cdot m_2(i)}\exp(\text{sim}(\mathbf{z}_i,\mathbf{z}_{l^+})/\tau)\Big)^{-1}\cdot\mathbb{E}\Big[\mathbb{E}_{(i,l^+)\in S_{\text{neg+}}}\big[\exp\big(\text{sim}(\mathbf{z}_i,\mathbf{z}_{l^+})/\tau\big)\big] \\
&\qquad - \big[\sum_{(i,l')\in S_{\text{neg-}}}\frac{1}{C_{\text{w}}}\exp(\text{sim}(\mathbf{z}_i,\mathbf{z}_{l'})/\tau) + \sum_{(i,l^+)\in S_{\text{neg+}}}\frac{C_{\text{w}}\text{-}m_1(i)}{C_{\text{w}}\cdot m_2(i)}\exp(\text{sim}(\mathbf{z}_i,\mathbf{z}_{l^+})/\tau)\big]\Big] \\
&\le \exp(1/\tau)\cdot\mathbb{E}\Big[\mathbb{E}_{(i,l^+)\in S_{\text{neg+}}}\big[\exp\big(\text{sim}(\mathbf{z}_i,\mathbf{z}_{l^+})/\tau\big)\big] - \big[\sum_{(i,l')\in S_{\text{neg-}}}\frac{1}{C_{\text{w}}}\exp(\text{sim}(\mathbf{z}_i,\mathbf{z}_{l'})/\tau) + \sum_{(i,l^+)\in S_{\text{neg+}}}\frac{C_{\text{w}}\text{-}m_1(i)}{C_{\text{w}}\cdot m_2(i)}\exp(\text{sim}(\mathbf{z}_i,\mathbf{z}_{l^+})/\tau)\big]\Big] \\
&\le -\frac{m_1(i)}{C_{\text{w}}} + \exp(1/\tau)\cdot\mathbb{E}\Big[\,\|\mathbb{E}_{(i,l^+)\in S_{\text{neg+}}}\big[\exp\big(\text{sim}(\mathbf{z}_i,\mathbf{z}_{l^+})/\tau\big)\big] - \sum_{(i,l^+)\in S_{\text{neg+}}}\frac{C_{\text{w}}\text{-}m_1(i)}{C_{\text{w}}\cdot m_2(i)}\exp(\text{sim}(\mathbf{z}_i,\mathbf{z}_{l^+})/\tau)\|\,\Big] \\
&\overset{(2)}{\le} \frac{m_1(i)}{C_{\text{w}}}(\exp(2/\tau) - 1) + \frac{C_{\text{w}}\text{-}m_1(i)}{C_{\text{w}}}\Big[\Big(1 - Pr(S_{\text{neg+}},\epsilon)\Big)\Big(\exp(1/\tau) - \exp(-1/\tau)\Big) + Pr(S_{\text{neg+}},\epsilon)\cdot\epsilon\Big] \\
&\le \frac{m_1(i)}{C_{\text{w}}}(\exp(2/\tau) - 1) + \frac{1}{4m_2^2(i)\cdot\epsilon^2}\Big(\exp(1/\tau) - \exp(-1/\tau)\Big)^3 + \epsilon
\end{aligned} \tag{13}$$

where, (1) holds because the property of concave function $\log(x)$ and (2) holds due to the Chebyshev's inequality (12). When ϵ is equal to

$\big(\frac{2N(C-1)}{C\cdot K}\big)^{-2/3}\cdot\big(\exp(1/\tau)-\exp(-1/\tau)\big)$, we have

$$\lim_{\eta,N\to\infty}\mathbb{E}\big[\ell_{\text{ms}}\big]-\mathbb{E}\big[\ell_{\text{ms}}\big]\leq\frac{5}{4}\Big(\frac{2N(C-1)}{C\cdot K}\Big)^{-2/3}\cdot\big(\exp(1/\tau)-\exp(-1/\tau)\big)+\frac{N(C-1)(K-1)}{\eta(2N-2)\cdot C\cdot K}(\exp(2/\tau)-1) \tag{14}$$

On the other hand:

$$\begin{aligned}
&\lim_{\eta,N\to\infty}\mathbb{E}\big[\ell_{\text{ms}}\big]-\mathbb{E}\big[\ell_{\text{ms}}\big]\\
&=-\mathbb{E}\Big[\log\big[\sum_{(i,l')\in S_{\text{neg-}}}\exp(\text{sim}(\mathbf{z}_i,\mathbf{z}_{l'})/\tau)+\sum_{(i,l^+)\in S_{\text{neg+}}}\frac{C_{\text{w}}\text{-}m_1(i)}{m_2(i)}\exp\big(\text{sim}(\mathbf{z}_i,\mathbf{z}_{l^+})/\tau\big)\big]-\log\mathbb{E}_{(i,l^+)\in S_{\text{neg+}}}\big[\exp\big(\text{sim}(\mathbf{z}_i,\mathbf{z}_{l^+})/\tau\big)\big]\Big]+\log C_{\text{w}}\\
&\geq-\mathbb{E}\Big[\log\big[\frac{m_1(i)\exp(1/\tau)}{C_{\text{w}}-m_1(i)}+\sum_{(i,l^+)\in S_{\text{neg+}}}\frac{\exp\big(\text{sim}(\mathbf{z}_i,\mathbf{z}_{l^+})/\tau\big)}{m_2(i)}\big]-\log\mathbb{E}_{(i,l^+)\in S_{\text{neg+}}}\big[\exp\big(\text{sim}(\mathbf{z}_i,\mathbf{z}_{l^+})/\tau\big)\big]\Big]-\log\Big(\frac{C_{\text{w}}-m_1(i)}{C_{\text{w}}}\Big)\\
&\geq-\frac{m_1(i)\exp(1/\tau)}{C_{\text{w}}-m_1(i)}\cdot\frac{1}{\exp(-1/\tau)}=-\frac{m_1(i)}{C_{\text{w}}-m_1(i)}\exp(2/\tau).
\end{aligned} \tag{15}$$

Combining Formula.(14) and Formula.(15), we are able to conclude that ℓ_{ms} converges to $-\frac{1}{\tau}\mathbb{E}_{(i,j)\in S_{\text{pos-}}}\big[\text{sim}(\mathbf{z}_i,\mathbf{z}_j)\big]+\log\mathbb{E}_{(i,l^+)\in S_{\text{neg+}}}\big[\exp\big(\text{sim}(\mathbf{z}_i,\mathbf{z}_{l^+})/\tau\big)\big]+\log C_{\text{w}}$.

By the way, the w(i,l) for $\mathbf{M}_{\text{pos}}$ in ℓ_{ms} is set to 0 to avoid the negative-positive-coupling effect, which means avoiding positive pairs appearing in the negative term in Thm.2, which can affect the learning efficiency and performance. The denominator could theoretically be 0. However, we want to clarify that we have implemented a precautionary measure in our code by using *'len(torch.unique(labels))'* to check for it. If this situation occurs, the calculation and backpropagation will be skipped. As these probabilities are exceedingly low, we conducted extensive experiments and didn't encounter this issue in any of trials.

2 Baseline details

We compare MsCtrl with 10 state-of-the-art methods:

1) Domain Adversarial Neural Network (**DANN**) [2] and 2) Maximum Classifier Discrepancy (**MCD**) [9] are strongly related to our work. Like many other DG works, we use these strategies for source domain alignment as comparison baselines.

The methods based on data augmentation and invariant representation learning: 3) Deep Domain-adversarial Image Generation (**DDAIG**) [12], 4) Feature Stylization and Domain-aware Contrastive Learning (**FeatStyl**) [4], 5) Permuted AdaIN (**pAdaIN**) [8], 6) Style Neophile (**StyleNeo**) [5], 7) Self-supervised Contrastive Regularization (**SelfReg**) [6], 8) Domain-specific Optimized Normalization (**DSON**) [10], 9) Risk Extrapolation (**VREx**) [7]

We also compare with methods based on novel learning strategies: 10) Representation Self-challenging (**RSC**) [3], and 11) Diversified Neural Averaging (**DNA**) [1].

Then, we introduce these methods in more detail:

- **ERM[11]** combines the data from all source domains and minimizes the cross entropy loss of classification.
- **DANN[2]** adopts a domain discriminator and uses adversarial learning to align feature distributions across domains.
- **MCD [9]** solves a minimax problem of finding two classifiers that maximize the difference between target samples and then generating features that minimize this difference.
- **DDAIG [12]** maps the source training data to unseen domains, ensuring that the generated data can be correctly classified by the label classifier while confusing the domain classifier.
- **DSON [10]** combines heterogeneous normalization methods to remove domain-specific information and extract domain-agnostic feature representations, optimizing the trade-off between cross-category variance and domain invariance.
- **RSC [3]** discards dominant features that correlate with labels and appears to activate feature representations applicable to out-of-domain data without extra network parameters.
- **pAdaIN [8]** reduces the representation of global statistics in image classifiers by randomly swapping statistics between samples of a given batch. This allows the network to rely on cues such as shape or texture.reduces the representation of global statistics in image classifiers by randomly swapping statistics between samples of a given batch.
- **VREx [7]** reduces differences in risk across training domains to reduce a model's sensitivity to extreme distributional shifts, including those with both causal and anti-causal elements. This paper that REx can recover the causal mechanisms of the targets while also providing robustness to covariate shift.
- **FeatStyle [4]** utilizes feature statistics to stylize original features while preserving class information, and utilizes a contrastive loss to ensure domain invariance and increase class discriminability. The set of ranges chosen for equally positive pairs is the S_{pos} and the corresponding augmented instances.
- **SelfReg [6]** proposes a class-specific domain perturbation layer (CDPL) that enables effective application of mixture augmentation even when using only positive pairs. Different from our work, his positive sample selection scope is all data of the same class S_{pos}.
- **DNA [1]** proposes a fastensemble architecture for DG task, which proposes novel pruned Jensen-Shannon (PJS) divergence and loss.
- **StyleNeo [5]** synthesizes novel styles during training by using a monotone submodular optimization and a greedy algorithm, while managing multiple queues to store previously observed styles.

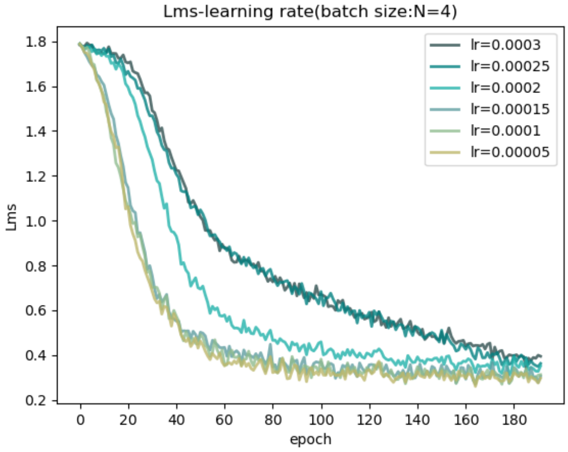

Figure 1. Visualization of learning curves with respect to the number of epoch on different learning rates.

3 Additional Results

3.1 Learning Rate and Convergence of $\mathcal{L}_{ms}$

As shown in Figure 1, the fastest convergence speed of $\mathcal{L}_{\mathrm{ms}}$ can be obtained when the learning rate is 0.0005 or 0.0001. At the same time, the convergence speed of cross entropy decreases with the increase in the learning rate. Therefore, in our experiments, the learning rate of the feature extractor was set to 0.0001.

As emphasized in the main paper, the key idea behind MsCtrl is to reduce the underlying target risk under the "control" of minimizing source risk and the source-source representation discrepancy. Therefore, we further analyze the convergence of the dynamic domain-weighted contrastive loss. A lower value of the dynamic domain-weighted contrastive loss indicates that the latent representations of the same class are more similar and those for the different classes are more dissimilar. Thus, we deem that the dynamic domain-weighted contrastive loss indicates the alignment of subdomains of the source domains. We intend to obtain a generalization model under the condition that the subdomains of the source domain are aligned, so this term should converge quickly.

And as shown in Figure 2, when the learning rate of the feature extractor is 0.0001, the dynamic domain-weighted contrastive loss is generally reduced to the convergence value within 10 epochs. The specific convergence value is affected by the dataset. Datasets with more categories tend to have large convergence values.

3.2 Sensitivity to Hyper-Parameter α and β

The sensitivity results for the hyper-parameters α and β are shown in Tables 1 and 2, respectively.

Intuitively, α can be considered as the movement velocity of the augmented domain after the maximum discrepancy of the classifiers. In general, the results are good as long as the hyper-parameter α is not too small. Setting α too small prevents alignment between the random augmentations and the source domain. The performance of the model does not vary significantly within the nearby range, and our method still outperforms the compared methods.

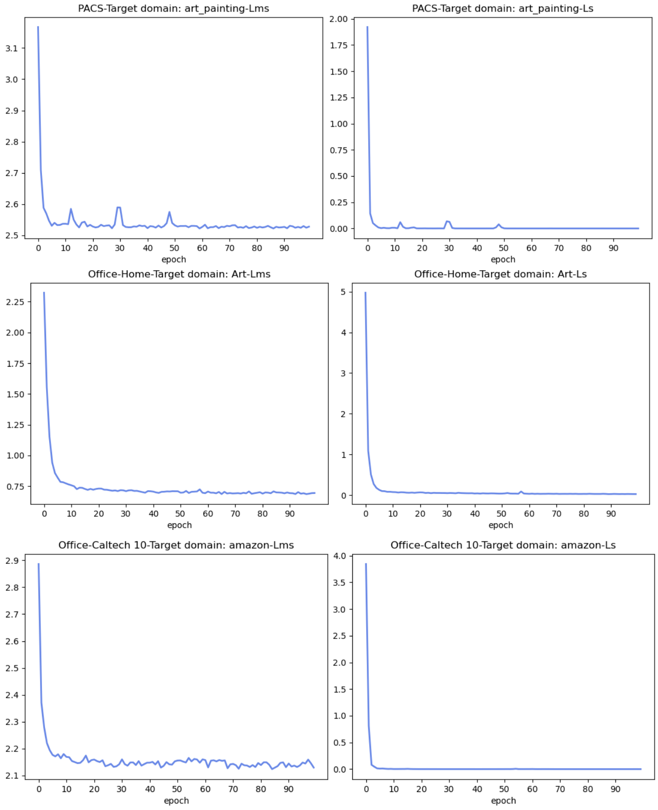

Figure 2. Visualization of convergence with respect to the number of epochs on leave "Art-painting" domain out task on PACS, leave "Art" domain out task on Office-Home, and leave "Amazon" domain out task on Office-Caltech-10, respectively.

Table 1. Sensitivity to hyper-parameter α on the PACS dataset.

α	A	C	P	S	Avg.
			ResNet-18		
α = 0.2	85.74	79.99	95.57	80.71	85.50
α = 0.4	85.52	81.61	95.95	82.50	86.40
α = 0.6	85.35	81.06	96.17	82.39	86.24
α = 0.8	85.25	81.87	95.51	82.03	86.17
			ResNet-50		
α = 0.2	85.94	84.91	96.53	82.62	87.50
α = 0.4	85.89	84.65	96.95	84.65	88.04
α = 0.6	87.06	86.31	97.01	84.47	88.71
α = 0.8	87.68	86.50	96.97	86.12	89.32

In all the experiments in the paper, β is set to 0.5, which is a fixed hyper-parameter. The results are shown in Table 2. Setting β too high can cause disagreement between classifiers in the source domains, while setting it too low can result in a lack of diversity on augmented domain.

Table 2. Sensitivity to hyper-parameter β on the PACS dataset with Resnet18 backbone.

β	A	C	P	S	Avg.
$\beta = 0.8$	$85.82^{\pm 1.}$	$81.48^{\pm 1.}$	$95.99^{\pm .5}$	$81.03^{\pm 1.}$	86.08
$\beta = 0.5$	$85.52^{\pm .8}$	$81.61^{\pm .7}$	$95.95^{\pm .5}$	$82.50^{\pm .5}$	86.40
$\beta = 0.3$	$84.93^{\pm .5}$	$81.63^{\pm .5}$	$96.37^{\pm .2}$	$82.56^{\pm .9}$	86.37

3.3 Sensitivity to the Temperature τ

In the paper, we chose the default temperature parameter τ in the InstDisc method, which was the first to introduce the InfoNCE contrastive loss, and this parameter has been widely adopted in related literature. We conducted additional ablation experiments on τ, as shown in Table 3. We found that the temperature has a minor impact on the model's accuracy within [0.05,0.1], and the performance of our method surpassed the current SOTA methods at all three settings.

Table 3. Sensitivity to temperature τ on the PACS dataset with Resnet18 backbone.

τ	A	C	P	S	Avg.
τ = 0.10	$85.72^{\pm .6}$	$81.50^{\pm .4}$	$96.17^{\pm .4}$	$81.21^{\pm .6}$	86.15
τ = 0.07	$85.52^{\pm .8}$	$81.61^{\pm .7}$	$95.95^{\pm .5}$	$82.50^{\pm .5}$	86.40
τ = 0.05	$85.99^{\pm 1.}$	$81.53^{\pm .5}$	$96.37^{\pm .2}$	$81.62^{\pm .4}$	86.38

3.4 Semantic Consistency of Augmented Data

We study the effect of random augmentation by fully utilizing the source domain data and fully utilizing the augmented data, namely, replacing $\tilde{\mathbf{x}}$ with $\mathbf{x}$ in all formulae and replacing $\mathbf{x}$ with $\tilde{\mathbf{x}}$ in all formulae. The results are shown in Table 4.

Table 4. Ablation study results on image augmentation on the PACS dataset with ResNet-18 backbone.

$\mathbf{x}$	$\tilde{\mathbf{x}}$	A	C	P	S	Avg.
Baseline		$75.20^{\pm .4}$	$78.71^{\pm .2}$	$89.70^{\pm .3}$	$70.41^{\pm .2}$	78.51
✓	–	$83.07^{\pm .4}$	$79.36^{\pm .3}$	$95.20^{\pm .1}$	$74.08^{\pm 3.}$	82.93
–	✓	$82.94^{\pm 2.}$	$81.24^{\pm .5}$	$94.91^{\pm .3}$	$82.15^{\pm .4}$	85.26
✓	✓	$85.52^{\pm .6}$	$81.61^{\pm .5}$	$95.95^{\pm .4}$	$82.50^{\pm .4}$	86.40

For augmented instances $\tilde{\mathbf{x}}$, a basic requirement of image augmentation is that the instance semantic consistency shall be preserved after augmentation. Under the condition that all the $\mathbf{x}$ are replaced with $\tilde{\mathbf{x}}$ in all formulae, the average classification accuracy is 85.26%, which is higher than the case of using only the original instances, but lower than the case of using both the original and the augmented instances. The results suggest that our image augmentation approach $\mathcal{M}$ preserves the original semantics.

At the same time, we find that this result is similar to the result of negative sample selection without domain information. Therefore, we believe that this selection method destroys the domain relationship on the original multi-source domain data set. Thus, we do not use augmented domain data to calculate $\mathcal{L}_{\text{ms}}$.

3.5 Diameter of the Aligned Convex Hull

This part is a supplement to the positive pairs selection strategy module in Section 4.6 of the paper. The algebraic description of convex hull diameter approximation is the maximum squared Euclidean distance over all positive pairs in a batch, that is

$$dist_{\text{max}}(F;S) := -\max_{(i,j)\in S} \|F(\mathbf{x}_{\text{i}}), F(\mathbf{x}_{\text{j}})\|_2^2. \tag{16}$$

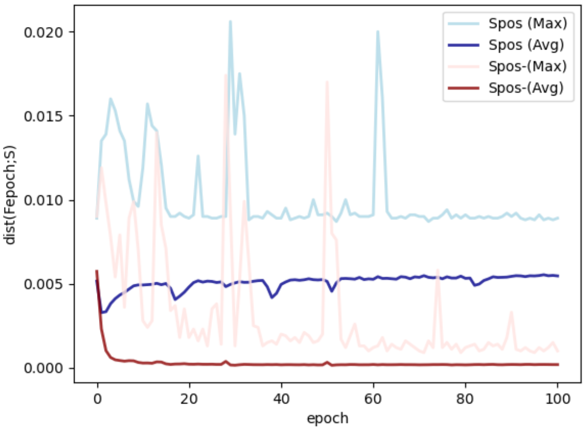

Figure 3. Average distance $dist(F_{\text{epoch}}; S)$ ($\times 10^{-3}$) and maximum distance $dist_{\text{max}}(F_{\text{epoch}}; S)$ with respect to the number of epoch on different selection of positive pairs on PACS with ResNet-18.

The visible results are shown in Figure 3. All results are computed before model parameter backpropagation for each iteration. It can be seen that when pulls each positive instance pair closer, the convex hull diameter also decreases accordingly. In addition to the rapid reduction of the average distance $dist(F_{\text{epoch}}; S_{\text{pos-}})$, the maximum distance of the positive pairs from $S_{\text{pos-}}$ also decreases significantly. In contrast, using positive pairs from S_{pos} show no significant decrease in the maximum distance and average distance. To some extent, the function of $\mathcal{L}_{\text{ms}}$ is to reduce the diameter of the aligned convex hull.

References

[1] Xu Chu, Yujie Jin, Wenwu Zhu, Yasha Wang, Xin Wang, Shanghang Zhang, and Hong Mei, 'Dna: Domain generalization with diversified neural averaging', in *International Conference on Machine Learning*, pp. 4010–4034. PMLR, (2022).

[2] 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] Zeyi Huang, Haohan Wang, Eric P Xing, and Dong Huang, 'Self-challenging improves cross-domain generalization', in *European Conference on Computer Vision*, pp. 124–140. Springer, (2020).

[4] Seogkyu Jeon, Kibeom Hong, Pilhyeon Lee, Jewook Lee, and Hyeran Byun, 'Feature stylization and domain-aware contrastive learning for

domain generalization', in *Proceedings of the 29th ACM International Conference on Multimedia*, pp. 22–31, (2021).

[5] Juwon Kang, Sohyun Lee, Namyup Kim, and Suha Kwak, 'Style neophile: Constantly seeking novel styles for domain generalization', in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7130–7140, (2022).

[6] 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*, pp. 9619–9628, (2021).

[7] David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Dinghuai Zhang, Remi Le Priol, and Aaron Courville, 'Out-of-distribution generalization via risk extrapolation (rex)', in *International Conference on Machine Learning*, pp. 5815–5826. PMLR, (2021).

[8] Oren Nuriel, Sagie Benaim, and Lior Wolf, 'Permuted adain: Reducing the bias towards global statistics in image classification', in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9482–9491, (2021).

[9] Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada, 'Maximum classifier discrepancy for unsupervised domain adaptation', in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3723–3732, (2018).

[10] Seonguk Seo, Yumin Suh, Dongwan Kim, Geeho Kim, Jongwoo Han, and Bohyung Han, 'Learning to optimize domain specific normalization for domain generalization', in *European Conference on Computer Vision*, pp. 68–83. Springer, (2020).

[11] Vladimir N Vapnik, 'An overview of statistical learning theory', *IEEE transactions on neural networks*, **10**(5), 988–999, (1999).

[12] Kaiyang Zhou, Yongxin Yang, Timothy Hospedales, and Tao Xiang, 'Deep domain-adversarial image generation for domain generalisation', in *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 13025–13032, (2020).