Domain Generalization with Correlated Style Uncertainty

Zheyuan Zhang¹, Bin Wang¹, Debesh Jha¹, Ugur Demir¹, Ulas Bagci¹
¹ Machine & Hybrid Intelligence Lab, Northwestern University, USA
ulas.bagci@northwestern.edu

Abstract

Domain generalization (DG) approaches intend to extract domain invariant features that can lead to a more robust deep learning model. In this regard, style augmentation is a strong DG method taking advantage of instance-specific feature statistics containing informative style characteristics to synthetic novel domains. While it is one of the state-of-the-art methods, prior works on style augmentation have either disregarded the interdependence amongst distinct feature channels or have solely constrained style augmentation to linear interpolation. To address these research gaps, in this work, we introduce a novel augmentation approach, named Correlated Style Uncertainty (CSU), surpassing the limitations of linear interpolation in style statistic space and simultaneously preserving vital correlation information. Our method’s efficacy is established through extensive experimentation on diverse cross-domain computer vision and medical imaging classification tasks: PACS, Office-Home, and Camelyon17 datasets, and the Duke-Market1501 instance retrieval task. The results showcase a remarkable improvement margin over existing state-of-the-art techniques. The source code is available https://github.com/freshman97/CSU.

1. Introduction

Recent years have witnessed the remarkable success of Deep learning (DL) in computer vision domain operating under the premise that the training (source) and testing (target) datasets adhere to a principle of independent and identically distributed (iid) data [48]. However, this oversimplified assumption often fails in practice when there is a distribution drift between training and testing datasets. The violation of this assumption induces the phenomenon that well-trained model in the source domain degrades dramatically in the target domain. If domain generalization is successfully implemented within a DL model, it would inherently solve issues related to domain shift. This would not only signify a significant advancement in the field, but also streamline the practical deployment of DL models across various domains. For example, a car detector should perform accurately both on sunny and cloudy days. DL based medical image segmentation algorithm, for another example, should generate stable segmentation regardless of the acquisition and scanner differences, and so on.

Domain adaptation and domain generalization are two distinct approaches for addressing the challenge of domain shift in machine learning. A widely adopted approach to counteract domain shift problems entails acquiring unlabeled data from the target domain. By leveraging this data, we can adapt a model, initially trained on the source domain, to align better with the characteristics of the target domain. This strategy is called Domain Adaptation (DA), which has been the subject of much systematic investigation in the last few years and achieved promising results in many fields [48]. To ensure robust model adaptation, regularization algorithms such as entropy regularization [24,26] are also applied to target domain during training. Domain adaptation assumes that the target domain is known and that some data from the target domain is available for training. However, accessing the target domain data can be quite challenging. Specifically in high-risk applications (e.g., medical data), target domain data might not be available at all. This is a primary research concern of Domain Generalization (DG) as a strong alternative path to DA.

In DG, the objective is to develop models that can perform well across a wide range of domains without explicit training on each individual domain. Unlike DA, DG assumes that the target domain is unknown and that
no data from the target domain is available for training [34, 48]. Since target data is unavailable for investigating domain shift, DG relies solely on extracting robust, domain-invariant feature representations from diverse training distributions. As a result, when domain information is feasible, feature alignments among different domains could significantly leverage the model’s out-of-distribution generalization ability, as shown in previous research [16, 19].

What do we propose? In this paper, we propose a novel DG method, called Correlated Style Uncertainty (CSU). The method is specifically designed to retain the relationship among distinct feature space channels while addressing the distribution drift between target and source domains. The effectiveness of our approach, as depicted in Figure 1(c), is clearly evident when compared with widely-used techniques like MixStyle and DSU. Like MixStyle and DSU, our newly introduced algorithm, CSU, belongs to the family of style augmentation methods employed for DG, but it presents distinctive advantages.

How is CSU different from previous style augmentation? In the context of style augmentation, MixStyle [50] operates by randomly choosing two samples and performing linear interpolation between their corresponding style attributes, to put it succinctly. While MixStyle is useful, it’s confined to generating in-distribution samples, potentially limiting network’s generalization capabilities. pAdaIN [22] rearranges the instance-specific feature statistics within a batch, thus sharing the same problem with MixStyle. DSU [18], contrarily, addresses this limitation by creating out-of-distribution samples via uncertainty modeling of feature statistics. However, DSU’s effectiveness is hinged upon the assumption that each channel operates independently, implying that inter-channel correlations bear no impact on the task at hand. This assumption might not always hold in practical scenarios. Style Neophile [11] select style prototypes that represent the distribution of source styles stored in the source style queue using MMD distance from style storage during training. [42, 46] apply the adversarial attack on the feature statistics for domain generalization while introducing expensive computational burden.

Our proposed method, CSU, is fundamentally different from these methods because i) it produces out-of-distribution samples, ii) correlation among different channels are retained and iii) the sampling process requires a negligible computational burden. Our proposed method, CSU, allows us to generate more meaningful style perturbations in discriminatory tasks, thus enhancing the model’s generalization capacity. Similar to hypotheses in prior research in the literature, we posit that the feature statistics follow a multivariate Gaussian distribution; however, we acknowledge and account for correlations among the variates, unlike many traditional methods. This perspective allows us to consider a more realistic distribution that could potentially improve model performance in real-world scenarios.

In our specific approach, we first compute the covariance matrix at the mini-batch level and then estimate the distribution from the covariance matrix. This allows us to sample correlated feature statistics from the established distribution. This sampling allows us to generate the style statistics outside the linear interpolation while maintaining an identical correlation. In this way, more diverse but meaningful style augmentation can be applied during the training and increase the model’s generalization ability. We highlight our main contributions in this study as follows:

- Our proposed strategy (CSU) is a well-calibrated method that goes beyond the interpolation strategies by preserving correlation between different feature spaces. This allows us to generate more diverse and meaningful style augmentation during training which helps in building a more generalizable model. To the best of our knowledge, such a simple yet effective style augmentation strategy has never been explored.
- To evaluate the effectiveness of the proposed CSU model, we conducted extensive experiments on multi-domain classification benchmarking datasets, including PACS [14], Office-Home [31], Camelyon17 [1] and the Duke-Market1501 dataset for instance retrieval tasks [25, 45]. The quantitative results show that the CSU can significantly improve the model’s generalizability over other state-of-the-art (SOTA) methods.
- We have performed several ablation studies to investigate the optimal parameters for the CSU model. These investigations have covered factors such as the ideal position for model integration, the most efficient sampling hyperparameters, and the batch size that yields the greatest level of generalization.

2. Background and Related Works

Several DG strategies have been proposed in the literature; we briefly cover the mostly used ones under sub-categories as follows.

**Data Augmentation:** is highly valued for its role in exposing the model to an expanded set of instances, an essential element for successful Deep Learning. Many methods have been proposed to achieve strong data augmentation, including traditional image augmentation like BigAug [40], deep neural network-based image generation like in RandConv [36], and adversarial data augmentations [33]. These methods are suitable specifically when the domain tags of samples are agnostic. While data augmentation is a powerful technique that can improve the generalization performance of machine learning models, it also has some potential drawbacks in the context of domain generalization. For instance, one drawback is that data augmentation may not
be effective when the variations between the source and target domains are too significant. **Feature Alignment:** is a popular method in representation learning category of DG approaches. Given domain tags, the model will add regularization terms into loss functions to force the extracted features from all source domains to align to the same distribution. For instance, Li [15] introduced the Maximum Mean Discrepancy as a regularization term to achieve feature alignment across multiple domains. Zhao [44] proposed an entropy regularization term that measures the dependency between the learned features and corresponding labels. This regularization method can ensure the conditional invariance of learned features. One major drawback of feature alignment is that it can be difficult to determine the best way to align the features across domains. **Meta-learning:** has also attracted attention from DG communities [4, 29]. Meta-learning aims to learn the learning algorithm itself by learning from previous experience or tasks. By splitting the source domain samples into pseudo-train and pseudo-test, meta-learning mimics the potential domain shift of the actual target domain. Despite its promise, meta-learning can be computationally expensive and time-consuming. Since meta-learning involves training a model on a large number of tasks or domains, there is a risk that the model may overfit to the training data and not generalize well to new domains.

**Style Augmentation:** The final category of DG is very recent: *style augmentation*. This method comes from the simple observation that instance-specific feature statistics such as mean and standard deviation, contain informative style characteristics and can be applied to the style-transferring model [9]. This phenomenon allows us to generate different style images while maintaining the same semantic concept. For example, Seo et al. [27] proposed one domain-specific normalization method by calculating the feature statistics of each domain. Pan et al. [17] add perturbation on the feature embedding with simple Gaussian noise during training. Zhou et al. [50] presented mixing styles (MixStyle) of training instances, and increased the source domain diversity. As a result, authors leveraged the trained model’s generalizability. Nuriel et al. [22] alternatively proposed a Permutated Adaptive Instance Normalization (pAdaIN) method to rearrange the instance-specific feature statistics within a batch, thus improving the model’s generalizability. In a slightly different angle, Li et al. [18] quantified feature statistics’ uncertainty (DSU) and sampled new style feature statistics from the uncertainty distribution, resulting in novel out-of-distribution domains being synthesized implicitly. Kang et al. [11] select style prototypes from style storage during training using the MMD distance and these prototypes represent the distribution of source styles stored in the source style queue. [42, 46] apply the adversarial attack on the feature statistics for domain generalization while introducing expensive computational burden. Our work is directly related to MixStyle and DSU as we are using the same hypothesis but addressing the major limitations of them, and sharing some similarities with pAdaIN, from the same efforts for synthesizing novel domains. Unlike them, our proposed CSU generates out-of-distribution feature statistics while maintaining the correlation between features.

3. Methods

3.1. Correlation within the style statistics

Given batch level feature maps $x \in \mathbb{R}^{B \times C \times H \times W}$ of the network $f(in, \phi)$ where $in$ denotes the batch-wise inputs and $\phi$ denotes the network parameters. We can formulate the instance-specific feature statistics mean $\mu \in \mathbb{R}^{B \times C}$ and standard deviation $\sigma \in \mathbb{R}^{B \times C}$ as follows

$$\mu(x) = \frac{1}{HW} \sum_{h=1}^{H} \sum_{w=1}^{W} x_{b,c,h,w},$$

$$\sigma^2(x) = \frac{1}{HW} \sum_{h=1}^{H} \sum_{w=1}^{W} (x_{b,c,h,w} - \mu(x))^2.$$  

Thus, we can formulate the channel-wise covariance matrix $\Sigma_{\mu} \in \mathbb{R}^{C \times C}, \Sigma_{\sigma} \in \mathbb{R}^{C \times C}$ of $\mu, \sigma$:

$$\Sigma_{\mu} = \frac{1}{B} (\mu - E(\mu))^T (\mu - E(\mu)),$$

$$\Sigma_{\sigma} = \frac{1}{B} (\sigma - E(\sigma))^T (\sigma - E(\sigma)),$$

where $E(\mu), E(\sigma)$ represents the mean value of $\mu, \sigma$ over batch dimension. It is worth noting that the rank of $\Sigma_{\mu}, \Sigma_{\sigma}$ is strictly limited by $\min(B, C) \leq C$. Previous research indicated that the feature maps are unlikely to be linearly independent over the channel dimension [8, 41]. Without any form of regularization, it becomes difficult to assume a diagonal covariance matrix and a zero correlation across each channel, as suggested in the top row of Figure 2.

We observed that the correlation matrix of style statistics (regardless of the mean or the standard deviation values) is not diagonal. Indeed, there exists a strong correlation among the channels. We applied eigenvalue decomposition over the calculated correlation matrix and find that very few eigenvectors dominate most variance, as shown in the bottom row of the Figure 2. This inspired us to rethink the augmentation of style statistics. The correlation matrix indicates that the combinations of feature statistics are not arbitrary but limited by task objectives and training procedures. Most variances happen within the specific principal directions. Arbitrary augmentation over the style statistics might damage the training itself. Previous research about why InstanceNorm can not outperform the BatchNorm in the discriminative tasks also proves this finding [20].

2002
3.2. Modeling correlated style uncertainty

Given that the correlation matrix is real, symmetric, and positive semi-definite, then we can apply eigenvalue decomposition on $\Sigma_\mu$, $\Sigma_\sigma$ to analyze its subspaces as:

$$\Sigma_\mu = Q_\mu \text{diag}(\Lambda_\mu) Q_\mu^T,$$

$$\Sigma_\sigma = Q_\sigma \text{diag}(\Lambda_\sigma) Q_\sigma^T,$$

$$Q_\mu Q_\mu^T = Q_\sigma Q_\sigma^T = I,$$

$$\Lambda_\mu, \Lambda_\sigma \in \mathbb{R}^C, Q_\mu, Q_\sigma \in \mathbb{R}^{C \times C},$$

where $\Lambda_{\mu,i} \geq \Lambda_{\mu,j} \geq 0$, $\Lambda_{\sigma,i} \geq \Lambda_{\sigma,j} \geq 0$ ($i > j$) represent the sorted eigenvalues, and $Q_\mu, Q_\sigma$ are the corresponding eigenvectors. The eigenvector corresponding to the largest eigenvalue represents the direction that we could apply dense augmentation. Eigenvectors corresponding to the eigenvalues of 0 or close to 0 are not considered in the data augmentation process due to low variance across such directions within the dataset.

Assuming that the $\mu, \sigma$ still follows the multi-variable Gaussian distribution, and $k_\mu, k_\sigma$ represent the independent variable number (or the rank of the corresponding covariance matrix), then we can represent the probability distribution function as:

$$f_\mu = \frac{1}{(2\pi)^{k_\mu} \det^*(\Sigma_\mu)} \exp^{-\frac{(\mu-E(\mu))^T \Sigma_\mu^+ (\mu-E(\mu))}{2}},$$

$$f_\sigma = \frac{1}{(2\pi)^{k_\sigma} \det^*(\Sigma_\sigma)} \exp^{-\frac{(\sigma-E(\sigma))^T \Sigma_\sigma^+ (\sigma-E(\sigma))}{2}},$$

where the $\det^*$ is the pseudo-determinant and $\Sigma^+$ is the generalized inverse. Based on this distribution function, we further derive the correlated uncertainty augmentation after we sample $\epsilon_\mu, \epsilon_\sigma \in \mathbb{R}^{N \times C}$ from the standard Gaussian distribution $Y \sim \mathcal{N}(0, I)$ as follow:

$$P_\mu = Q_\mu \text{diag}(\Lambda_\mu)^{\frac{1}{2}} Q_\mu^T,$$

$$P_\sigma = Q_\sigma \text{diag}(\Lambda_\sigma)^{\frac{1}{2}} Q_\sigma^T,$$

$$\hat{\epsilon}_\mu = \epsilon_\mu P_\mu, \quad \hat{\epsilon}_\sigma = \epsilon_\sigma P_\sigma.$$
Batch Dimension

Channel

\[ \Sigma = (A - E(A))^T (A - E(A)) = PP^T \]

Original Domain

Generated Domain

Intermediate Features

Augmented Features

Standard Gaussian Distribution

Correlated Gaussian Distribution

Extract Feature Statistics

Remove Feature Statistics

Restored Feature Statistics

Intermediate Features

Normalized Features

Augmented Features

Normalized Features

Decomposed Covariance Matrix: \( P \)

Figure 3. Visualization of feature statistics augmentation using correlated style uncertainty (CSU). Given the intermediate features extracted from the network, we first estimate the covariance matrix of feature statistics and decompose the covariance matrix as described in Sec 3.2. Based on this decomposition, we could generate correlated augmentation from the standard Gaussian distribution that shares identical distribution as the original domain. Then, we update raw data feature statistics by adding this correlated augmentation. Finally, we restore the feature statistics back to the normalized features and achieve the augmented features.

3.3. Style augmentation with CSU

Based on the previous two sections, we now present the style augmentation with correlated style uncertainty as follows:

\[
\begin{align*}
\beta(x) &= \mu(x) + \lambda \cdot \hat{\epsilon}_\mu, \\
\gamma(x) &= \sigma(x) + \lambda \cdot \hat{\epsilon}_\sigma,
\end{align*}
\]

where \( \lambda \sim Beta(\alpha, \alpha) \) represents the augmentation intensity generated from the Beta distribution. Hyperparameter \( \alpha \) controls the shape of the distribution. In the ablation experiments, we further show the influence of hyperparameter selections on the final performance. We can understand the equation in one more intuitive way, the first part is to provide in-domain samples to cover the whole training domain, and the second is to provide the extrapolation while maintaining the same data distribution.

\[
\beta(x) = \underbrace{\mu(x)}_{\text{In Domain Sample}} + \underbrace{\lambda \cdot \hat{\epsilon}_\mu}_{\text{Out Domain extrapolation}}.
\]

The final augmented instance feature can be defined as:

\[
CSU(x) = \gamma(x) \left( \frac{x - \mu(x)}{\sigma(x)} \right) + \beta(x).
\]
Figure 4. Some examples from multi-domain classification including (a) PACS, (b) Office-Home, and (c) Camelyon17 dataset.

Table 1. Experimental results on the PACS multi-domain classification task. CSU achieves around highly significant improvements over the baseline in Art, Cartoon, and Sketch domains, respectively, as: 14.3%, 5.6%, and 12.2%. Besides, CSU also shows superiority over other methods, which demonstrates its effectiveness. \( \alpha \) is set as 0.1.

<table>
<thead>
<tr>
<th>Method</th>
<th>Reference</th>
<th>Art</th>
<th>Cartoon</th>
<th>Photo</th>
<th>Sketch</th>
<th>Average(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>ICLR 2018</td>
<td>74.3</td>
<td>76.7</td>
<td>96.4</td>
<td>68.7</td>
<td>79.0</td>
</tr>
<tr>
<td>Mixup [39]</td>
<td>ICLR 2018</td>
<td>76.8</td>
<td>74.9</td>
<td>95.8</td>
<td>66.6</td>
<td>78.5</td>
</tr>
<tr>
<td>Manifold Mixup [32]</td>
<td>ICLR 2018</td>
<td>75.6</td>
<td>70.1</td>
<td>93.5</td>
<td>65.4</td>
<td>76.2</td>
</tr>
<tr>
<td>CutMix [38]</td>
<td>ICCV 2019</td>
<td>74.6</td>
<td>71.8</td>
<td>95.6</td>
<td>63.3</td>
<td>76.8</td>
</tr>
<tr>
<td>JGen [2]</td>
<td>CVPR 2019</td>
<td>79.4</td>
<td>75.3</td>
<td>96.0</td>
<td>71.6</td>
<td>80.5</td>
</tr>
<tr>
<td>RSC [10]</td>
<td>ECCV 2020</td>
<td>78.9</td>
<td>76.9</td>
<td>94.1</td>
<td>76.6</td>
<td>81.7</td>
</tr>
<tr>
<td>L2A-OT [49]</td>
<td>ECCV 2020</td>
<td>83.3</td>
<td>78.2</td>
<td>96.2</td>
<td>76.3</td>
<td>82.8</td>
</tr>
<tr>
<td>SagNet [21]</td>
<td>CVPR 2021</td>
<td>83.6</td>
<td>77.7</td>
<td>95.5</td>
<td>76.3</td>
<td>83.3</td>
</tr>
<tr>
<td>pAdaIN [22]</td>
<td>CVPR 2021</td>
<td>81.7</td>
<td>76.6</td>
<td>96.3</td>
<td>75.1</td>
<td>82.5</td>
</tr>
<tr>
<td>SFA-A [17]</td>
<td>ICCV 2021</td>
<td>81.2</td>
<td>77.8</td>
<td>93.9</td>
<td>73.7</td>
<td>81.7</td>
</tr>
<tr>
<td>MixStyle [50]</td>
<td>ICLR 2021</td>
<td>82.3</td>
<td>79.0</td>
<td>96.3</td>
<td>73.8</td>
<td>82.8</td>
</tr>
<tr>
<td>DSU [18]</td>
<td>ICLR 2022</td>
<td>83.6</td>
<td>79.6</td>
<td>95.8</td>
<td>77.6</td>
<td>84.1</td>
</tr>
<tr>
<td><strong>CSU (Ours)</strong></td>
<td></td>
<td>85.0</td>
<td>81.0</td>
<td>99.3</td>
<td>78.4</td>
<td>85.2</td>
</tr>
</tbody>
</table>

Table 2. Experimental results on Office-Home multi-domain classification task. We achieve around 4.3%, 13.6%, 0.9% improvement over the baseline in the Art, Clipart, and Product domain, respectively. CSU consistently outperforms the other strong baseline models with considerable margins (\( \alpha = 0.4 \)).

<table>
<thead>
<tr>
<th>Method</th>
<th>Art</th>
<th>Clipart</th>
<th>Product</th>
<th>Real</th>
<th>Average(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>58.8</td>
<td>48.3</td>
<td>74.2</td>
<td>76.2</td>
<td>64.4</td>
</tr>
<tr>
<td>Mixup [39]</td>
<td>58.2</td>
<td>49.3</td>
<td>74.7</td>
<td>76.1</td>
<td>64.6</td>
</tr>
<tr>
<td>CrossGrad [28]</td>
<td>58.4</td>
<td>49.4</td>
<td>73.9</td>
<td>75.8</td>
<td>64.4</td>
</tr>
<tr>
<td>Manifold Mixup [32]</td>
<td>56.2</td>
<td>46.3</td>
<td>73.6</td>
<td>75.2</td>
<td>62.8</td>
</tr>
<tr>
<td>CutMix [38]</td>
<td>57.9</td>
<td>48.3</td>
<td>74.5</td>
<td>75.6</td>
<td>64.1</td>
</tr>
<tr>
<td>RSC [10]</td>
<td>58.4</td>
<td>47.9</td>
<td>71.6</td>
<td>74.5</td>
<td>63.1</td>
</tr>
<tr>
<td>L2A-OT [49]</td>
<td>60.6</td>
<td>50.1</td>
<td>74.8</td>
<td>77.0</td>
<td>65.6</td>
</tr>
<tr>
<td>MixStyle [50]</td>
<td>58.7</td>
<td>53.4</td>
<td>74.2</td>
<td>75.9</td>
<td>65.5</td>
</tr>
<tr>
<td>DSU [18]</td>
<td>60.2</td>
<td>54.8</td>
<td>74.1</td>
<td>75.1</td>
<td>66.1</td>
</tr>
<tr>
<td><strong>CSU (Ours)</strong></td>
<td>61.3</td>
<td>54.9</td>
<td>74.9</td>
<td>76.1</td>
<td><strong>66.8</strong></td>
</tr>
</tbody>
</table>

4.1.2 Office-Home classification

Office-Home [31] is another benchmark dataset for DG, containing four domains: Art, Clipart, Product, and Real-World, and each domain consists of 65 categories. The dataset contains 15,500 images with an average of around 70 photos per class. Similarly, we compare our model’s performance with other SOTA methods.
Medical image analysis often suffers the most from domain shifting, given that multiple parameters, like the image acquisition device, and protocol can induce significant domain shift. We validate the model’s performance on the challenging Camelyon17 dataset [1], containing images from five medical centers. This dataset consists of the histopathological images as input and the labels indicating whether the central region includes any tumor tissue. Due to lacking reported performance from the current literature, we conduct this experiment from scratch based on the WILDS framework proposed by Koh [13]. Besides the baseline, we compare our model with three state-of-art strategies, including the MixStyle [50], pAdaIN [22], DSU [18]. For a fair comparison, we repeat the pAdaIN and DSU experiments on the same framework with the MixStyle and use the best configuration reported in the original paper.

Table 3 shows the experiment results using two models in the two domains. We could observe that CSU outperforms other methods by a large margin. CSU achieves around 26.9%, 19.6% advancement in mAP using the ResNet-50 model in the Market1501 to Duke and the Duke to Market1501 experiment, correspondingly. Likewise, CSU achieves around 20.1%, 24.2% improvement for the OSNet model experiment. We could also observe similar advancements in ranking accuracy, and CSU achieves impressive improvement over other methods.

### 4.2. Instance Retrieval Experiments

The undertaking of person re-identification, which involves the cross-camera recognition of individuals, poses a substantial domain generalization challenge. Considering each distinct camera output as its own unique domain amplifies the intricacy inherent to the person re-identification task. Following previous research, we conducted this experiment on the commonly used Duke [25] and Market1501 [45] datasets. To evaluate the model’s generalizability, we take one dataset as training and test the performance on the other domain. The camera data from the test domain does not participate in any training process.

We adopt the exact framework implementation of MixStyle and test the CSU influence on the ResNet50 [7] and OSNet [49]. Similarly, ranking accuracy and mean average precision (mAP) are performance measures. For a fair comparison, we repeat the pAdaIN and DSU experiments on the same framework with the MixStyle and use the best configuration reported in the original paper.

Table 4 shows the experiment results using two models in the two domains. We could observe that CSU outperforms other methods by a large margin. CSU achieves around 26.9%, 19.6% advancement in mAP using the ResNet-50 model in the Market1501 to Duke and the Duke to Market1501 experiment, correspondingly. Similarly, CSU achieves around 20.1%, 24.2% improvement for the OSNet model experiment. We could also observe similar advancements in ranking accuracy, and CSU achieves impressive improvement over other methods. Nevertheless, to show the effectiveness of CSU rather than position fine-tuning, we insert the permutation in all positions as described in Sec 4.3. The supplementary materials show that changing the inserting position can achieve even more significant performance advancement.

### 4.3. Ablation Experiments

**Insert Position Selection:** To answer the question of where we should insert the CSU, we conduct comprehensive experiments on the PACS and Office-Home datasets using the ResNet18 structure. We investigate all possible positions of ResNet18, including the first Convolution, first Max Pooling, and 1, 2, 3, and 4 Res-block, which are named 0-5, respectively. We divide the experiment into several groups according to the inserted CSU number. Within each group, we shift the start position one by one from 0 to end. For example, for the group containing 2 CSU blocks, we will have 01, 12, 23, 34, and 45 potential combinations and five comparison experiments in total. To avoid the influence of hyperparameters, we set $\alpha = 0.3$ for all experiments. Thus, we can reasonably and adequately compare the inserting position’s influence on final performance.

Figure 5 shows the ablation experiment results. Inserting 6 blocks of CSU in all potential positions achieves the best results for the PACS dataset while inserting 4 blocks of CSU...
Hyper-parameter Selection: As described in the previous section, the hyper-parameter alpha determines the intensity of augmentation during training by manipulating the shape of the Beta distribution. Here, we show the influence of alpha on the PACS, Office-Home using the ResNet18 structure. Similarly, to avoid the influence of different inserting positions, we insert CSU block in all positions for every experiment. We select $\alpha$ from 0.1, 0.2, 0.3, 0.4, 0.5, 0.7, and 0.9 for one comprehensive experiment. As shown in Figure 6, we can find that a smaller number of $\alpha < 0.5$ always performs better than the relatively larger number ($> 0.5$). Based on these experiment results, we recommend selecting the alpha from 0.1, 0.2, 0.3, and 0.4, and the best configuration may vary according to the tasks.

Effect of Batch Size: The batch size could have a notable impact on the accuracy of correlation data, making it critical to scrutinize its effect on the final model’s generalizability. Accordingly, we’ve introduced the CSU at every stage, as per the earlier section, and set a fixed value of $\alpha = 0.3$ in all the experiments. We compare the model’s performance with a batch size of 16, 32, 64, 128, 256, and 512. Figure 7 shows the experimental result. We found that it might be hard to estimate an accurate correlation when the batch size is too small. Similarly, when the batch size is too large, the network tends to converge to sharp minimizers of the training and testing functions, leading to poorer generalizations, as shown in previous research [12].

5. Conclusion

In this paper, we introduce the Correlated Style Uncertainty (CSU), a novel domain generalization approach that transcends linear interpolation while upholding the correlation among feature channels. With detailed and exhaustive ablation studies, we’ve determined the optimal position for integrating the CSU model, assessed the impact of sampling hyperparameters, and evaluated the ideal batch size for yielding a more universally applicable model. Comprehensive experiments across numerous datasets corroborate that CSU model notably enhances the model’s ability to generalize. This research is expected to catalyze further in-depth studies on feature statistics augmentation in future.

This work is supported by NIH R01-CA246704, R01-CA240639, R15-EB030356, R03-EB032943, U01-DK127384-02S1, and U01-CA268808.
References


[18] Xiaotong Li, Yongxing Dai, Yixiao Ge, Jun Liu, Ying Shan, and LINGYU DUAN. Uncertainty modeling for out-of-distribution generalization. In *International Conference on Learning Representations*, 2022. 1, 2, 3, 4, 6, 7


[38] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (CVPR)*, pages 6023–6032, 2019. 6


