FFM: Injecting Out-of-Domain Knowledge via Factorized Frequency Modification

Zijian Wang Yadan Luo Zi Huang Mahsa Baktashmotlagh
The University of Queensland
{firstname.lastname}@uq.edu.au

Abstract

This work investigates the Single Domain Generalization (SDG) problem and aims to generalize a model from a single source (i.e., training) domain to multiple target (i.e., test) domains coming from different distributions. Most of the existing SDG approaches focus on generating out-of-domain samples by either transforming the source images into different styles or optimizing adversarial noise perturbations applied on the source images. In this paper, we show that generating images with diverse styles can be complementary to creating hard samples when handling the SDG task, and propose our approach of Factorized Frequency Modification (FFM) to fulfill this requirement. Specifically, we design a unified framework consisting of a style transformation module, an adversarial perturbation module, and a dynamic frequency selection module. We seamlessly equip the framework with iterative adversarial training that facilitates learning discriminative features from hard and diverse augmented samples. Extensive experiments are performed on four image recognition benchmark datasets of Digits, CIFAR-10-C, CIFAR-100-C, and PACS, which demonstrates that our method outperforms existing state-of-the-art approaches.

1. Introduction

Domain shift [3, 32, 8] is a fundamental problem in computer vision and it commonly occurs when the training and test sets follow different distributions due to the shift in illumination, weather, appearance, background, *etc.* Machine learning models suffer considerable performance degradation when they are exposed to the domain shift. To address this problem, domain generalization methods have been introduced [51] that learn to perform well on the out-of-distribution (OOD) data. Most of the existing domain generalization methods [5, 36, 51, 16] assume the access to multiple source domains collected under different environmental conditions and aim to find domain invariant representations. A more realistic setting of domain generalization has been introduced recently [47, 39], where only single source domain exists at the training stage, namely Single Domain Generalization (SDG) [52, 39, 13, 47, 27].

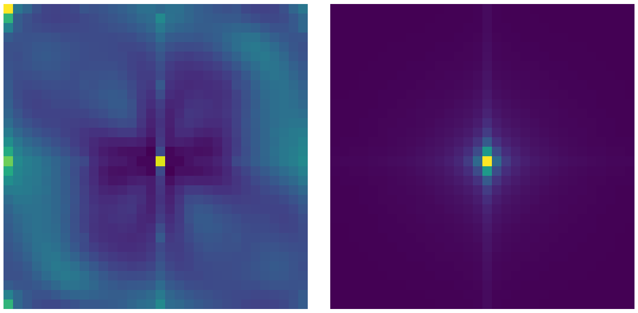

(a) Adversarial Perturbation (b) Style Manipulation

Figure 1: Difference between Fourier spectrum $\mathbb{E}[|\mathcal{F}(X - \hat{X})(m,n)|]$ of original images and (a) adversarially perturbed [47] (b) style modified [52] images. We average the difference between Fourier spectrum over all CIFAR-10 training images. We can see that adversarial perturbations are more concentrated on high frequency components (*i.e.*, brighter to the corner), while style modification methods mainly affect the low frequency components (*i.e.*, brighter to the middle).

Generally speaking, existing SDG approaches focus on to address unforeseen domain shifts by generating fictitious domains and can be categorized as adversarial noise perturbation-based [47, 59] and style manipulation-based [52, 27]. The former one learns adversarial perturbations on source images to form an auxiliary training set to train a generalizable classifier. Style manipulation-based approaches make use of an image generating network to create fictitious domains, with an objective of maximizing entropy [27, 39] or minimizing mutual information [52] between the augmented and the source images. Heuristically, adversarial perturbation-based methods focus on generating hard samples, while the aim of style manipulation-based methods is to create diverse samples.

Despite the encouraging results achieved by either category of the SDG methods, we argue that diversity and hardness are complementary to each other, so that simultaneously considering both characteristics in designing SDG algorithms can potentially enhance the generalization performance. As a proof of concept, recent research on model robustness [57, 50] reveals that for a naturally trained model, imperceptible adversarial noise perturbations are encoded in high-frequency components, while the more obvious style is encoded in the low-frequency components. We visualize the frequency spectrum for the recent SDG methods of [47, 52] in Fig.1, which confirms that the two categories of SDG methods are complementary for domain generalization in Fourier domain.

In light of the above observation, in this paper, we aim to simultaneously generate diverse and hard fictitious domains. To this end, a novel Factorized Frequency Modification (FFM) module is proposed with two learnable branches. To generate hard samples, the noise perturbation branch modifies the high-frequency components of the input samples. The diversity of generated samples is achieved via the style transformation branch by modifying the amplitude of the low-frequency component of the input sample. Unlike the existing approaches of [56, 53] that use hard code frequency selection parameters, we develop a systematic way of frequency selection by dynamically learning high/low-frequency bands for different datasets. The proposed end-to-end framework optimizes the task model and FFM iteratively in an adversarial manner. Under this training scheme, FFM gradually promotes the diversity and hardness of generated samples, while the task model is learning to predict under an enlarged domain gap.

The contribution of our work can be summarized as follows: (1) we propose a single domain generalization framework, namely Factorized Fourier Modification, which expands the source domain by simultaneously augmenting the human-perceptible style-encoded in the low-frequency component and imperceptible noise-encoded in the high frequency component- of input samples; (2) A dynamic frequency selection module is proposed to learn domain invariant high/low frequency band, which is superior to using domain-specific hard-coded frequency band as in [56, 53]; (3) To validate the effectiveness of our model, we conduct extensive experiments on four single domain generalization benchmark datasets, including digits, CIFAF-10-C, CIFAR-100-C, and PACS. The results clearly demonstrate the superiority of our proposed approach over the state-of-the-art single domain generalization methods of ADA [47], MEADA [59], L2d [52], *etc.*

2. Related Work

Domain Generalization (DG) methods aim to tackle the domain shift problem by aligning multiple source domains in the latent space. To this end, the existing approaches either follow statistical matching [36, 35, 41] or domain adversarial learning [29, 34, 42, 26] techniques. While the early DG methods focus on domain invariant learning, the learnt models may overfit to the source domains, limiting their generalization on unseen domains. Inspired by meta-learning [14], some DG works [24, 2, 30, 12, 11] aim to alleviate this issue by exposing the model to meta domain shifts during the training phase. Data augmentation [62, 61] is another way to prevent the model from overfitting to source domains. Specifically, augmentation-based DG methods create novel domains by interpolating among source domains, either in the image- [61, 60] or the feature-level [28, 9]. Some methods take the advantage of image style transfer techniques [19, 7], which either mixes [62] or partially swaps [38] the intermediate convolutional feature statistics of the samples in source domains to broaden the training set.

Single Domain Generalization (SDG) is a more challenging yet realistic setting, and assumes that only one source domain is available at training stage. Since most of the existing DG methods exploit the domain correlations to improve generalization power of the model, they fail to perform well on the SDG setting. The current SDG methods can be categorised to adversarial gradient image augmentation [47, 59] or style augmentation [27, 52, 39] techniques. Methods from the former category optimize adversarial noises to perturb source images, and aim to generate hard samples for the classifier. Specifically, they either optimize the noise perturbation on the source samples by maximizing classification error [47], or entropy maximization [59]. [39] applies an auxiliary Wasserstein autoencoder to promote difference between generated and source images in the input space, and therefore relaxes the feature space constraint. Style augmentation-based methods [52, 27] adopt an image generation network to diversify the source domain which optimize the generator network by minimizing upper bound of mutual information [52] or maximizing InfoNCE loss [27] between the generated samples and the corresponding source samples.

Frequency-domain Analysis and Model Robustness. Recent literature establishes the connection between data manipulation in frequency domain and the robustness of the model. [44] shows the network can be easily misled by slightly perturbing the frequency magnitude of the input. [57] demonstrates that, from the Fourier perspective, adversarial perturbations to a naturally trained model tend to concentrate on the high-frequency components of the data. [40, 50] point out that a model tends to grasp low-frequency information at the early stage of the training, but gradually

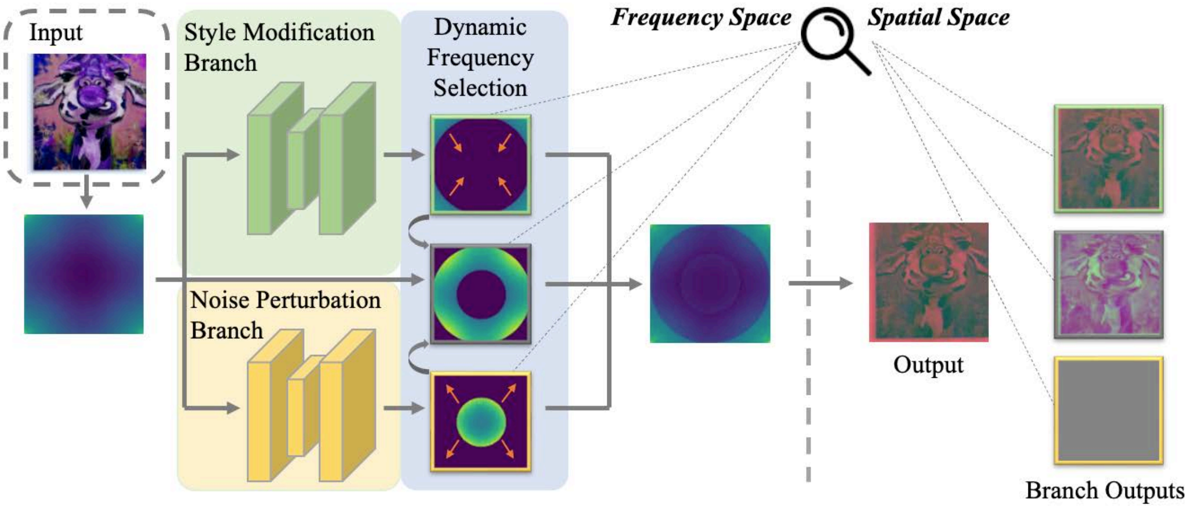

Input
Style Modification Branch
Dynamic Frequency Selection
Frequency Space
Spatial Space
Noise Perturbation Branch
Output
Branch Outputs

Figure 2: The augmentation module of the proposed Factorized Frequency Modification (FFM). FFM augments input images through a style modification branch and a noise perturbation branch. A dynamic frequency selection strategy is proposed to balance the contribution of the frequency components from the two branches. The outputs are visualized in the spatial space, on the right side of the figure.

overfit to the human imperceptible high-frequency components of input, which can result in sacrificing robustness for higher accuracy.

In the area of domain adaptation and generalization, frequency-based methods have been proposed recently [56, 53], which aim to generate images by swapping the styles of the images in different domains. Specifically, [56] proposes to swap the low-frequency components of the amplitude spectrum between the source and target images. Enlightened by Mix-up [58] strategy, [53] calculates the weighted sum between the source and target amplitude spectrum to interpolate among the available source domains. Although a model trained with target-like source images achieves promising generalization performance, current Fourier-based methods still suffer from the following limitations: (1) Swapping or mixing the amplitude of domains implicitly assumes that the unseen target domain is the interpolation of the source domains, and such a strong assumption can be easily breached in practice; (2) The impact of the high-frequency components is overlooked in the current methods; (3) Existing Fourier-based methods manually select the cut-off for low-frequency components, which limits the diversity of the generated samples; (4) Due to lack of access to multiple training domains, existing Fourier-based strategies like [56, 53] are not applicable in the SDG setting.

To overcome the above mentioned limitations, we propose our approach of factorized frequency modification to generate semantic-aware and diverse out-of-domain samples, and propose to dynamically learn high/low frequency band for different tasks. As highlighted by the recent research of [57, 50, 40], naturally trained model eventually overfits to human imperceptible high-frequency patterns. This phenomenon results in higher accuracy, at the cost of model generalization and robustness. Specifically, we argue that only focusing on low-frequency augmentation and neglecting the importance of high-frequency components may hinder the generalization performance.

3. Methodology

Notations and preliminary. Given a source domain $\mathcal{S} = \{x_i, y_i\}_{i=1}^{N_s}$ with N_s samples, SDG aims to learn a domain agnostic task model $\mathcal{H} : \mathcal{X} \rightarrow \mathcal{Y}$ that can perform well on an unseen target domain $\mathcal{T}$. The model can be written as: $\mathcal{H} = f \circ h$, with $f : \mathcal{X} \rightarrow \mathcal{Z}$ denoting the feature extractor and $h : \mathcal{Z} \rightarrow \mathcal{Y}$ the classifier. For a single image x, we omit the dimension of image channels C and we have Fourier Transformation formulated as:

$$\mathcal{F}(x)(m,n) = \sum_{h,w} x(h,w) e^{-j2\pi(\frac{h}{H}m + \frac{w}{W}n)}, j^2 = -1. \tag{1}$$

Here $\mathcal{F} : \mathbb{R}^{H\times W} \rightarrow \mathbb{C}^{H\times W}$ denotes Fast Fourier Transformation (FFT). From the signal $\mathcal{F}(x)$, we can obtain amplitude $\mathcal{F}^A(x)$ and phase spectrum $\mathcal{F}^P(x)$ as follows:

$$\begin{aligned} \mathcal{F}^A(x) &= [Re(\mathcal{F}(x))^2 + Im(\mathcal{F}(x))^2]^{\frac{1}{2}}; \\ \mathcal{F}^P(x) &= arctan\left[\frac{Im(\mathcal{F}(x))}{Re(\mathcal{F}(x))}\right], \end{aligned} \tag{2}$$

where $Re(\cdot)$ and $Im(\cdot)$ are the operators of keeping real and imaginary part of inputs, respectively.

The overview of our framework is shown in Fig.2. With the aim of expanding the distribution of the training set, we propose a Factorized Fourier Modification (FFM) module $\mathcal{G} : \mathcal{X} \rightarrow \hat{\mathcal{X}}$ to generate diverse and hard samples $\hat{\mathcal{X}}$ by applying transformations in the frequency domain. To this end, FFM learns two transformations, namely noise perturbation $g_1(\mathcal{F}(x); \theta_1)$ and style manipulation $g_2(\mathcal{F}(x); \theta_2)$, which aim to affect the high and low granularity information of an input image, respectively. Moreover, we design frequency masks of $M_{1,2} \in \{0, 1\}^{C*H*W}$, with C, H, W being the channel, height, and width. The frequency masks dynamically determine the passing bands for the frequency spectrum $\mathcal{F}(\hat{x})$. By applying a frequency transformation on the input signals, and training the model with the transformed data, we hypothesize that the model can perform well when tested on the unseen target domains.

3.1. Factorized Frequency Modification

Factorized Frequency Modification (FFM) module consists of noise perturbation branch and style transformation branch. Noise perturbation branch aims to affect high granularity information that is hardly perceptible to human, but able of harming the model predictions. Taking a source image signal $\mathcal{F}(x)$ as an input, noise perturbation branch augments the frequency spectrum as follows:

$$\mathcal{F}_1(x) = \mathcal{F}(x) + g_1(\mathcal{F}(x); \theta_1), \tag{3}$$

where $g_1(\cdot; \theta_1) \in \mathcal{G}$ represents the transformation function parameterized by θ_1. We implement $g_1(\cdot; \theta_1)$ with deep complex neural network [43] to transform $\mathcal{F}(x)$ in the Fourier domain.

Style transformation branch modifies the amplitude of low frequency components in the input signal, which is assumed to encode human perceptible style information, such as illumination and color. To enhance the diversity of the generated images, the style transformation branch learns to transform the source signal as follows:

$$\mathcal{F}_2(x) = g_2(\mathcal{F}(x); \theta_2), \tag{4}$$

where $g_2(\cdot; \theta_2)$ is the nonlinear transformation function parameterized by θ_2. Leveraging the property that style information are encoded in the amplitude, we design a group of convolutional layers to augment the style of the input. Specifically, $g_2(\cdot; \theta_2)$ is defined as:

$$g_2(\mathcal{F}(x); \theta_2) = g_2'(\mathcal{F}^A(x); \theta_2) e^{-j\mathcal{F}^P(x)}, \tag{5}$$

with $\mathcal{F}^P(x)$ and $\mathcal{F}^A(x)$ being the phase and amplitude of frequency spectrum $\mathcal{F}(x)$, obtainable from Eq.(2).

To resemble the frequency spectrum of the augmented image $\hat{x}$, FFM sums up the transformed frequency components with the ones not chosen by the frequency masks:

$$\begin{aligned} \mathcal{F}(\hat{x}) = &\mathcal{F}_1(x) \circ M_1 + \mathcal{F}_2(x) \circ M_2 \\ &+ \mathcal{F}(x) \circ (\mathbb{1} - (M_1 \| M_2)), \end{aligned} \tag{6}$$

Here, M_1 and M_2 are the frequency masks for the output of noise perturbation branch $\mathcal{F}_1(x)$ and style transformation branch $\mathcal{F}_2(x)$, so to construct the augmented frequency spectrum of $\mathcal{F}(\hat{x})$. Note that, $\mathcal{F}_{1,2}(x)$, $M_{1,2}$ and $\mathbb{1}$ all have the same dimensionality. Finally, by applying Inverse Fast Fourier Transformation, FFM transforms $F(\hat{x})$ from the frequency domain back to the original domain to obtain the augmented image $\hat{x}$.

3.2. Dynamic Frequency Selection

Unlike previous Fourier based method [56] that use fixed hyper-parameters to mask the low/high pass frequency, we propose a systematic way to dynamically learn the frequency selection mask M_1 and M_2. Inspired by [1], we model the mask learning process as training a binary belief network. Specifically, we first initialize $\tilde{M_1}, \tilde{M_2} \in \mathbb{R}^{C \times H \times W}$ as follows:

$$\begin{aligned} \tilde{M_1}(i,j) &= \begin{cases} 1, if\ d((i,j),(c_i,c_j)) < r_h \\ 0 \end{cases}, \\ \tilde{M_2}(i,j) &= \begin{cases} 1, if\ d((i,j),(c_i,c_j)) > r_l \\ 0 \end{cases}. \end{aligned} \tag{7}$$

$d(\cdot, \cdot)$ represents Euclidean distance between (i, j)th position (i, j) and the center of the mask (c_i, c_j), calculated on each channel of pixels. r_l and r_h represent the radius.

We calculate the confidence of each pixel position based on the amplitude spectrum of the input signal, and obtain the mask by comparing the confidence with a uniform probability distribution $\hat{P} \sim Uniform(0.49, 1)$, as follows:

$$\begin{aligned} M_1 &= \hat{P} < (\sigma(\tilde{M_1} \circ \mathcal{F}^A(x)) + 0.5), \\ M_2 &= \hat{P} < (\sigma(\tilde{M_2} \circ \mathcal{F}^A(x)) + 0.5), \end{aligned} \tag{8}$$

where σ denotes Sigmoid activation function.

3.3. Feature Space Frequency Augmentation

To further promote the diversity of the augmented samples, we equip the framework with feature-level frequency augmentation. We introduce two straightforward Fourier augmentation strategies on the feature-level: (1) random perturbation applied on the amplitude of selected frequency components; and (2) random frequency dropout.

(1) We formulate the first feature-level augmentation strategy as:

$$\mathcal{F}^A(\hat{z}) = \mathcal{F}^A(z) + g_z(\mathcal{F}^A(z), \theta_z) \circ M, \tag{9}$$

with $g_z(\cdot; \theta_z)$ denotes the nonlinear projection function and M being the random binary mask. We apply mask M to the generated amplitude, so as to select frequency components to be perturbed. We then reconstruct the frequency spectrum $\mathcal{F}(\hat{z})$ from the modified amplitude spectrum $\mathcal{F}^A(\hat{z})$ and the original phase spectrum $\mathcal{F}^P(z)$:

$$\mathcal{F}(\hat{z})(m,n) = \mathcal{F}^A(\hat{z})(m,n)e^{-j\mathcal{F}^P(z)(m,n)}. \tag{10}$$

(2) As a second feature-level augmentation, we randomly drop the frequency components as follows:

$$\mathcal{F}(\hat{z}) = \mathcal{F}(z) \circ M. \tag{11}$$

We apply inverse FFT on $F(\hat{z})$ to map back the features from the frequency domain to the spatial domain. In practice, we apply the above-mentioned augmentation strategies on features of the source image z and the augmented image $\hat{z}$. The augmentation layer can be easily integrated with backbone network, such as ResNet, WideResNet, AlexNet, etc. Note that, feature-level frequency augmentations are deactivated at the test time.

3.4. Adversarial Training

We employ adversarial training to improve the generalization ability of task model. Specifically, FFM module learns to increase the diversity and hardness of the generated fictitious domains, while the task model's goal is to obtain domain invariant representations in the latent space. The main goal of the existing adversarial training strategies for domain generalization is either to generate hard samples to confuse the classifier [39, 59], or to minimize the mutual information between the source samples and the corresponding augmented samples [52]. All the aforementioned approaches somehow fail to take into account the intra-class diversity among samples in their adversarial training. To jointly consider the intra-class diversity and hardness of generated samples, we adopt the supervised contrastive loss of [21]:

$$\begin{aligned} L_{supcl} &= -\sum_{i=0}^{N} \frac{1}{|P(i)|} \sum_{p \in P(i)} \log \frac{e^{(z_i \cdot z_p/\tau)}}{\sum_{a \in A(i)} e^{(z_i \cdot z_a/\tau)}} \\ P(i) &= \{p \in A(i) : y_p = y_i\}. \end{aligned} \tag{12}$$

Here, z_i denotes the latent representation of i-th sample. $A(i)$ is a set that contains the positive latent representation z^+ from the positive set $P(i)$, and the negative latent representation z^- of i-th sample. Temperature is denoted by τ. Maximizing the supervised contrastive loss promotes FFM to generate diverse positive samples that are uniformly distributed across the space.

The augmented images from FFM together with the original source images are passed to the task network, where we adopt standard cross entropy for classification:

$$\begin{aligned} L_{task} = -\frac{1}{2N} \big[&\sum_{i=0}^{N} y_i \log(h(f(x_i, \theta_f), \theta_h) \\ &+ \sum_{i=0}^{N} \hat{y_i} \log(h(f(\hat{x_i}, \theta_f), \theta_h)) \big], \end{aligned} \tag{13}$$

Overall Objective Function. A two-step iterative training strategy is adopted to optimize the FFM module $G(\cdot; \theta_1, \theta_2, \tilde{M_1}, \tilde{M_2})$, and the task model, consisting of $F(\cdot; \theta_f)$ and $H(\cdot; \theta_h)$. Specifically, given the source images X and the generated images $\hat{X}$, we freeze the weights of the task model and train the FFM module:

$$\max_{\theta_1, \theta_2, \theta_z, \tilde{M_1}, \tilde{M_2}} L_{supcl} \tag{14}$$

We then freeze the weights of FFM and optimize the task network:

$$\min_{\theta_f, \theta_h} L_{task} + \lambda L_{supcl}, \tag{15}$$

with λ being the hyper-parameter to balance the contribution of L_{supcl} to the overall objective function.

4. Experiments

4.1. Datasets

We evaluate our method on four benchmark SDG datasets, covering diverse object recognition scenes. (1) **Digits** contains 5 digit recognition datasets, including MNIST [22], SVHN [37], MNIST-M [15], SYN [15], and USPS [10]. These datasets are mainly different in the background, font, and image quality of their images. Following [59, 39], we take 10,000 images from MNIST as the source domain, and compute the model accuracy on all other domains. (2) **PACS** consists of 4 domains of photo, art painting, cartoon, and sketch. Each domain contains 7 classes and there are 9,991 images in total with the image size of 224×224. PACS is a more challenging dataset than Digits due to the large distribution shifts from one domain to the other. We follow the official split of the train [23], validation, and test. (3) **CIFAR-10-C** [18] and (4) **CIFAR-100-C** [18] contain tiny 32×32 RGB images from 10 and 100 classes, respectively. There are 19 corruptions from 4 main categories, including noise, blur, digital, and weather. Each corruption has 5 severity levels, with '5' denoting the severest corruption level.

4.2. Implementation details

For noise perturbation branch $g_1(\cdot; \theta_1)$, we employ 3-layer complex convolution neural networks [43], which has input and output channel equal to 3 and a hidden dimension of 64. Style transformation branch $g_2(\cdot; \theta_2)$ is a 3-layer convolution neural networks that has the same dimensionality with $g_1(\cdot; \theta_1)$. Both of the complex convolutional layer and convolutional layer has the kernel size of 1. For all the experiment we set initial radius r_l and r_h to be $0.5W$ and W, respectively.

4.3. Results on Digits

Experimental Setup. Following [59, 39, 52], we duplicate the channel of grey-scale images to convert them into

Table 1: Single domain generalization accuracy (%) on Digits. Models are trained on MNIST and evaluated on the rest of the digits datasets. Best performances are highlighted in bold.

	SVHN	MNIST-M	SYN	USPS	Avg.
ERM [45]	27.83	52.72	39.65	76.94	49.29
CCSA [35]	25.89	49.29	37.31	83.72	49.05
d-SNE [54]	26.22	50.98	37.83	**93.16**	52.05
JiGen [5]	33.80	57.80	43.79	77.15	53.14
ADA [47]	35.51	60.41	45.32	77.26	54.62
M-ADA [59]	42.55	67.94	48.95	78.53	59.49
ME-ADA [59]	42.56	63.27	50.39	81.04	59.32
RSDA [46]	47.4	81.5	62.0	83.1	68.5
RSDA+ASR [13]	52.8	80.8	64.5	82.4	70.1
L2D [52]	62.86	87.30	63.72	83.97	74.46
RandConv [55]	57.52	**87.76**	62.88	83.36	72.88
Ours	64.11	82.25	63.91	83.56	73.45
Ours+RandConv	**64.66**	84.92	**64.70**	84.80	**74.77**

RGB images, and we resize all the images to size 32 × 32. LeNet-5 is adopted as the backbone network for all digits experiments, with the SGD optimizer for both factorized frequency module and the backbone network.

Results. We report the single domain generalization accuracy in Tab. 1. The results shows that FFM outperforms SOTA on challenging domains, *i.e.,* SVHN and SYN, which have different backgrounds and styles from MNIST. For relatively easier domains like MINST-M and USPS, which are either different in background color or font from the source domain, our method achieves comparable results with the SOTA methods.

4.4. Results on CIFAR-10-C and CIFAR-100-C

Experimental Setup. Following [59, 39], we train our model on the training split of clean images, *i.e.,* CIFAR-10 or CIFAR-100, and test them on the test set of corrupted data. To make fair comparison, we use a randomly initialized WideResNet (16-4) backbone similar to the baselines. The network is optimized by using SGD with the initial learning rate of 0.1, which is gradually reduced by a cosine annealing scheduler.

Results. In Tab. 2, we compare the classification accuracy of baselines and our approach on CIFAR-10-C dataset, and on corruptions of severity level '5'. Due to the space limitation, we only report the results on 15 corruption types. As shown in Tab. 2, FFM outperforms both adversarial noise perturbation-based methods and style modification-based methods on a wide variety of corruptions. Specifically, our method achieves the highest results among baselines on most of the different types of 'Weather', 'Blur', and 'Noise' corruptions, with small drops on some of the 'Digital' corruptions. According to [57], most of the corruptions in 'Blur' and 'Noise' categories are biased towards high-frequency components, while corruptions in 'Weather' category exist in the low-frequency components. The superior results of FFM verifies the necessity of considering both high- and low-frequency components when designing domain generalization algorithms.

CIFAR-100-C is a more challenging dataset compared to CIFAR-10-C due to a more comprehensive label space. We report the results on CIFAR-100-C in Tab. 3. Since CIFAR-100-C has the same corruption types as CIFAR-10-C, we observe similar results and behaviour, where we largely outperform the SOTA by approximately 7.6%, 7.3%, 8.4%, and 5.9% on 'Weather', 'Blur', 'Noise', and 'Digital' corruptions, respectively. This confirms the superiority of our approach over baselines.

We further demonstrate the performance of different methods on CIFAR-10-C and CIFAR-100-C under five corruption levels in Fig.3. As shown in Fig.3, parts (a) and (b), the margin of accuracy between our method and the baselines are gradually getting enlarged as the level of corruption increases. This indicates that our method is more robust to the large domain shifts compared to the other baselines. Meanwhile, we observe a larger performance gain in the CIFAR-100-C compared to CIFAR-10-C, which shows that our method performs better than the baselines on the challenging domain generalization tasks.

4.5. Results on PACS

Experimental Setup. We employ ResNet-18 [17] pretrained on Imagenet as the backbone network, with the batch size of 64. For each generalization task, the backbone network is finetuned on the source domain and tested on the rest three target domains. We utilize SGD with a learning rate of 0.002 to optimize the network for 50 epochs. We also investigate the effectiveness of our method in a classical multi-source domain generalization setting, where we leave one domain out for test and use the other three domains to train the network. Under this setting, we utilize AlexNet and ResNet-18 as the backbone networks. We set the batch size to 64 and the learning rate to 0.001 to train the network for 30 epochs.

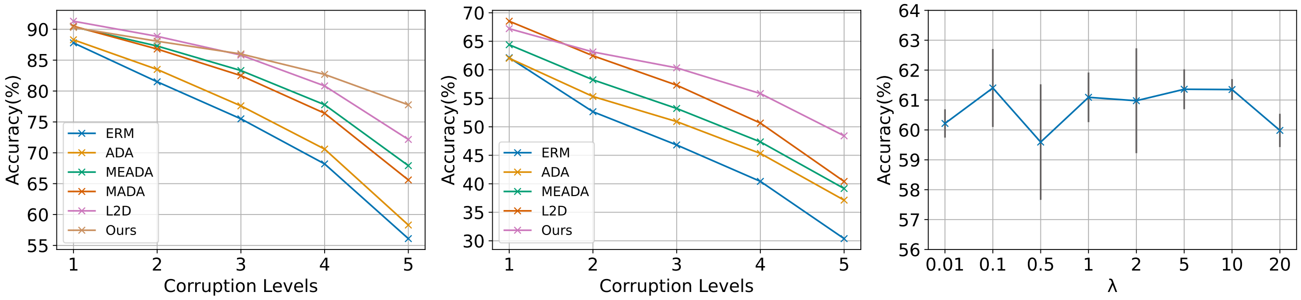
Results. Tab.4 shows the single domain generalization results on PACS dataset. We take one domain as source and report the average accuracy over the other three domains. The results show that our approach significantly outperforms the baselines. We also report the results under multi-source domain generalization setting in Tab.5. Note that under this setting, our method do not require any domain labels during training. Our proposed FFM achieves the SOTA results on different backbone networks.

4.6. Ablation Study

Impact of different components in FFM. To investigate the contribution of each component in the overall frame-

Table 2: Single domain generalization accuracy (%) on CIFAR-10-C. We report accuracy on 15 different types of corruption at the severity level 5. Best performances are highlighted in bold.

Method	Weather			Blur				Noise				Digital				
	Fog	Snow	Forest	Zoom	Defocus	Glass	Motion	Shot	Impulse	Gaussian	Jpeg	Pixelate	Elastic	Brightness	Contrast	Avg.
ERM [45]	65.92	74.36	61.57	59.97	53.71	49.44	63.81	35.41	25.65	29.01	69.90	41.07	72.40	91.25	36.87	56.15
CCSA [35]	66.94	74.55	61.49	61.96	56.11	48.46	64.73	33.79	24.56	27.85	69.68	40.94	72.36	91.00	35.83	56.31
M-ADA [39]	69.36	80.59	76.66	68.04	61.18	61.59	64.23	60.58	45.18	56.88	77.14	52.25	75.61	90.78	29.71	65.59
MEADA [59]	60.07	81.72	82.10	75.45	67.71	**72.55**	70.86	59.73	**46.78**	58.65	**85.52**	**77.48**	**79.80**	88.16	23.92	69.15
L2D [52]	69.21	78.70	81.35	72.86	64.58	61.53	68.52	78.32	13.61	74.81	82.31	53.19	76.50	**91.33**	48.16	69.08
Ours	**80.23**	**84.62**	**84.86**	**81.01**	**79.94**	67.50	**83.71**	**82.67**	23.16	**80.90**	81.80	70.17	77.40	90.82	**78.54**	**77.77**



(a) CIFAR-10-C

(b) CIFAR-100-C

(c) Parameter Sensitivity

Figure 3: The average classification accuracy (%) under five severity levels of corruption on (a) CIFAR-10-C and (b) CIFAR-100-C datasets. (c) Sensitivity to λ on PACS.

Table 3: Single domain generalization accuracy (%) on CIFAR-100-C. We report the average accuracy over 4 main categories of corruption at severity level 5. Best performances are highlighted in bold.

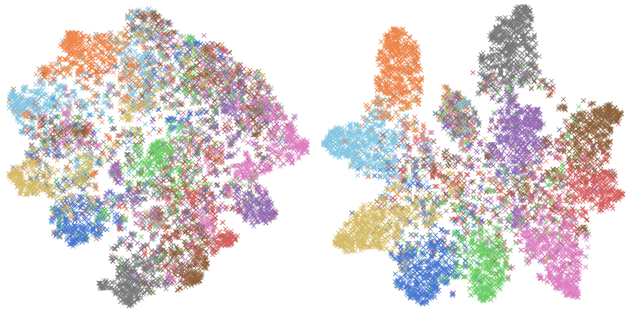
	Weather	Blur	Noise	Digital	Avg.
ERM [45]	6.38	32.48	38.01	37.34	28.55
ADA [47]	19.83	39.70	40.44	45.82	36.45
MEADA [59]	25.64	42.18	38.45	44.66	37.73
L2D [52]	25.40	37.91	43.34	46.07	38.18
Ours	**33.06**	**49.51**	**51.74**	**51.98**	**46.57**

Table 4: Single domain generalization accuracy (%) on PACS. Best performances are highlighted in bold.

	Photo	Art	Cartoon	Sketch	Avg.
ERM [45]	42.2	70.9	76.5	53.1	60.7
RSC [20]	41.6	73.4	75.9	56.2	61.8
L2D [20]	52.3	76.9	77.9	53.7	65.2
RSC+ASR [13]	54.6	76.7	**79.3**	61.6	68.1
GeomTex [31]	49.1	72.1	78.7	60.0	65.0
Ours	**61.4**	**80.5**	77.7	**62.1**	**70.4**
Ablation Study					
Ours-*Dyn. mask*	58.4	79.1	77.2	**62.1**	69.2
Ours-*Noise*	59.2	78.6	77.6	59.3	68.7
Ours-*Style*	55.1	78.0	76.4	59.0	67.1

work, we perform an ablation study on FFM and report the results in Tab.4. Specifically, we conduct 3 sets of experiments by removing (a) dynamic frequency selection; (b) noise perturbation branch, and (c) style modification branch, from the full model. Among those three variants, we find that removing the style modification branch brings the largest performance drop, which shows its importance in our overall framework. By removing noise perturbation, we see a slight drop in performance, shown in 'Ours-*noise*' results. This might due to the fact that the domain shift in PACS mainly comes from style changes, rather than noise and high-granularity information. Lastly, we remove the dynamic frequency selection component from our framework, and instead, use a fixed frequency selection, which results in 1.2% drop in the average performance.

Impact of feature augmentation. We study the impact of the proposed feature level augmentation strategy on the digits dataset and LeNet backbone, and show the results in Fig. 5. Specifically, we compare the performance of our proposed FFM with its two possible variants, including 1) *NoAug:* removing the Fourier feature augmentation from the network; and 2) *Layer1:* inserting feature augmentation after the first convolution. We observe that *Layer1* and



(a) MEADA

(b) Ours

Figure 4: The t-SNE visualizations of extracted target features on CIFAR-10-C. Features with the same semantic labels are plotted with the same color.

Table 5: Multi-source domain generalization accuracy (%) on PADS. D_ID indicates the requirement of domain label for a certain algorithm. Best performances are highlighted in bold.

	D_ID	P	A	C	S	Avg.
AlexNet						
DSN [4]	✓	83.30	61.10	66.50	58.60	67.40
Fusion [33]	✓	90.20	64.10	66.80	60.10	70.30
MetaReg [2]	✓	87.40	63.50	69.50	59.10	69.90
Epi-FCR [25]	✓	86.10	64.70	72.30	65.00	72.00
MASF [11]	✓	90.68	70.35	72.46	67.33	75.21
DMG [6]	✓	87.31	64.65	69.88	71.42	73.32
HEX [49]	✗	87.90	66.80	69.70	56.20	70.20
PAR [48]	✗	89.60	66.30	66.30	64.10	72.08
JiGen [5]	✗	89.00	67.63	71.71	65.18	73.38
ADA [47]	✗	85.10	64.30	69.80	60.40	69.90
MEADA [59]	✗	88.60	67.10	69.90	63.00	72.20
MMLD [34]	✗	88.98	69.27	**72.83**	66.44	74.38
L2D [52]	✗	**90.96**	71.19	72.18	67.68	75.50
Ours	✗	90.78	**71.86**	71.17	**75.31**	**77.28**
ResNet-18						
Epi-FCR [25]	✓	93.90	82.10	77.00	73.00	81.50
MASF [11]	✓	94.99	80.29	77.17	71.68	81.03
DMG [6]	✓	93.55	76.90	**80.38**	75.21	81.46
FACT [53]	✓	95.15	**85.37**	78.38	79.15	84.51
Jigen [5]	✗	96.03	79.42	75.25	71.35	80.51
ADA [47]	✗	**95.61**	78.32	77.65	74.21	81.44
MEADA [59]	✗	95.57	78.61	78.65	75.59	82.10
MMLD [34]	✗	96.09	81.28	77.16	72.29	81.83
L2D [52]	✗	95.51	81.44	79.56	80.58	84.27
Ours	✗	94.55	84.02	79.65	**82.46**	**85.17**

FFM consistently outperform the *NoAug* results, confirming the effectiveness of the purposed feature augmentation strategy.

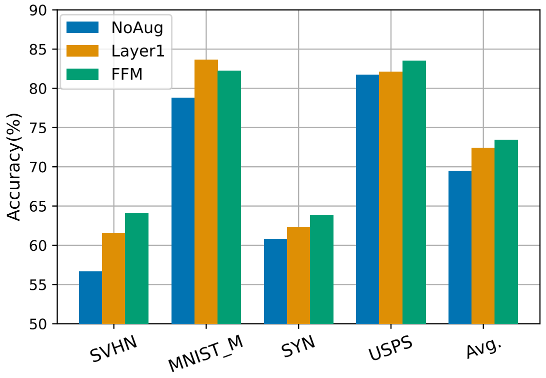


Figure 5: Single domain generalization accuracy (%) on Digits. We compare the performance of our proposed FFM with its two possible variants, including *NoAug* and *Layer1*.

Sensitivity to hyperparameter. We validate the significance of hyperparameter λ in our formulation and show the results in Fig.3(C). We conduct a single domain generalization task on PACS, taking 'photo' as the source domain and the rest as the target domains. We vary λ fron 0.01 to 20, and for each λ value, we conduct three random trials to compute the standard deviation The results show that the average accuracy varies from approximately 60% to 61.4% within a large range of λ in $0.01 - 20$. The results confirms the robustness and stability of the proposed approach with respect to hyperparameter λ.

Visualization We use t-SNE to visualize the feature distribution of MEADA, and our method on CIFAR-10-C. We randomly sample 5% of data from 15 corruptions at severity '5'. As we can see in Fig.4, our proposed method clearly outperform MEADA with better class-wise separations. MEADA fails to mostly accommodate samples in their correct clusters. Moreover, the proposed method has a better separation between the classes, which can help with the prediction.

5. Conclusion

We propose Factorized Frequency Modification (FFM) to address single domain generalization problem. The key idea of FFM is to augment source data with diverse and hard samples by transforming the style and high-granularity information of source images in the Fourier domain. A dynamic frequency selection strategy is developed to balance the contribution of transformed frequency components in the augmented output. Extensive experiments on four benchmark datasets demonstrate that the proposed FFM outperforms SOTA single domain generalization methods.

Acknowledgement This work was supported by Australian Research Council (ARC DP190102353, and CE200100025)

References

[1] Lei Jimmy Ba and Brendan J. Frey. Adaptive dropout for training deep neural networks. In Christopher J. C. Burges, Léon Bottou, Zoubin Ghahramani, and Kilian Q. Weinberger, editors, *NeurIPS*, 2013.

[2] Yogesh Balaji, Swami Sankaranarayanan, and Rama Chellappa. Metareg: Towards domain generalization using meta-regularization. In *NeurIPS*, 2018.

[3] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. 2010.

[4] Konstantinos Bousmalis, George Trigeorgis, Nathan Silberman, Dilip Krishnan, and Dumitru Erhan. Domain separation networks. In *NeurIPS*, 2016.

[5] Fabio Maria Carlucci, Antonio D'Innocente, Silvia Bucci, Barbara Caputo, and Tatiana Tommasi. Domain generalization by solving jigsaw puzzles. In *CVPR*, 2019.

[6] Prithvijit Chattopadhyay, Yogesh Balaji, and Judy Hoffman. Learning to balance specificity and invariance for in and out of domain generalization. In *ECCV*, 2020.

[7] Zhi Chen, Jingjing Li, Yadan Luo, Zi Huang, and Yang Yang. Canzsl: Cycle-consistent adversarial networks for zero-shot learning from natural language. In *WACV*, pages 874–883, 2020.

[8] Zhi Chen, Yadan Luo, Ruihong Qiu, Sen Wang, Zi Huang, Jingjing Li, and Zheng Zhang. Semantics disentangling for generalized zero-shot learning. In *ICCV*, 2021.

[9] Zhi Chen, Yadan Luo, Sen Wang, Ruihong Qiu, Jingjing Li, and Zi Huang. Mitigating generation shifts for generalized zero-shot learning. In *Proceedings of the 28th ACM International Conference on Multimedia*, 2021.

[10] John S. Denker, W. R. Gardner, Hans Peter Graf, Donnie Henderson, Richard E. Howard, Wayne E. Hubbard, Lawrence D. Jackel, Henry S. Baird, and Isabelle Guyon. Neural network recognizer for hand-written zip code digits. In *NeurIPS*, 1988.

[11] Qi Dou, Daniel Coelho de Castro, Konstantinos Kamnitsas, and Ben Glocker. Domain generalization via model-agnostic learning of semantic features. In *NeurIPS*, 2019.

[12] Ying-Jun Du, Jun Xu, Huan Xiong, Qiang Qiu, Xiantong Zhen, Cees G. M. Snoek, and Ling Shao. Learning to learn with variational information bottleneck for domain generalization. In *ECCV*, 2020.

[13] Xinjie Fan, Qifei Wang, Junjie Ke, Feng Yang, Boqing Gong, and Mingyuan Zhou. Adversarially adaptive normalization for single domain generalization. In *CVPR*, 2021.

[14] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In Doina Precup and Yee Whye Teh, editors, *ICML*, 2017.

[15] Yaroslav Ganin and Victor S. Lempitsky. Unsupervised domain adaptation by backpropagation. In *ICML*, 2015.

[16] Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. In *ICLR*, 2021.

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[18] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *ICLR*, 2019.

[19] Xun Huang and Serge J. Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *ICCV*, 2017.

[20] Zeyi Huang, Haohan Wang, Eric P. Xing, and Dong Huang. Self-challenging improves cross-domain generalization. In *ECCV*, 2020.

[21] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *NeurIPS*, 2020.

[22] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 1998.

[23] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M. Hospedales. Deeper, broader and artier domain generalization. In *ICCV*, 2017.

[24] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M. Hospedales. Learning to generalize: Meta-learning for domain generalization. In Sheila A. McIlraith and Kilian Q. Weinberger, editors, *AAAI*, 2018.

[25] Da Li, Jianshu Zhang, Yongxin Yang, Cong Liu, Yi-Zhe Song, and Timothy M. Hospedales. Episodic training for domain generalization. In *ICCV*, 2019.

[26] Haoliang Li, Sinno Jialin Pan, Shiqi Wang, and Alex C. Kot. Domain generalization with adversarial feature learning. In *CVPR*, 2018.

[27] Lei Li, Ke Gao, Juan Cao, Ziyao Huang, Yepeng Weng, Xiaoyue Mi, Zhengze Yu, Xiaoya Li, and Boyang Xia. Progressive domain expansion network for single domain generalization. In *CVPR*, 2021.

[28] Pan Li, Da Li, Wei Li, Shaogang Gong, Yanwei Fu, and Timothy M. Hospedales. A simple feature augmentation for domain generalization. In *ICCV*, 2021.

[29] Ya Li, Mingming Gong, Xinmei Tian, Tongliang Liu, and Dacheng Tao. Domain generalization via conditional invariant representations. In *AAAI*, 2018.

[30] Yiying Li, Yongxin Yang, Wei Zhou, and Timothy M. Hospedales. Feature-critic networks for heterogeneous domain generalization. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *ICML*, 2019.

[31] Xiao-Chang Liu, Yong-Liang Yang, and Peter Hall. Geometric and textural augmentation for domain gap reduction. In *CVPR*, pages 14340–14350, 2022.

[32] Yadan Luo, Zijian Wang, Zi Huang, and Mahsa Baktashmotlagh. Progressive graph learning for open-set domain adaptation. In *ICML*, pages 6468–6478, 2020.

[33] Massimiliano Mancini, Samuel Rota Bulò, Barbara Caputo, and Elisa Ricci. Best sources forward: Domain generalization through source-specific nets. In *ICIP*, 2018.

[34] Toshihiko Matsuura and Tatsuya Harada. Domain generalization using a mixture of multiple latent domains. In *AAAI*, 2020.

[35] Saeid Motiian, Marco Piccirilli, Donald A. Adjeroh, and Gianfranco Doretto. Unified deep supervised domain adaptation and generalization. In *ICCV*, 2017.

[36] Krikamol Muandet, David Balduzzi, and Bernhard Schölkopf. Domain generalization via invariant feature representation. In *ICML*, 2013.

[37] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. 2011.

[38] Oren Nuriel, Sagie Benaim, and Lior Wolf. Permuted adain: Reducing the bias towards global statistics in image classification. In *CVPR*, 2021.

[39] Fengchun Qiao, Long Zhao, and Xi Peng. Learning to learn single domain generalization. In *CVPR*, 2020.

[40] Nasim Rahaman, Aristide Baratin, Devansh Arpit, Felix Draxler, Min Lin, Fred A. Hamprecht, Yoshua Bengio, and Aaron C. Courville. On the spectral bias of neural networks. In *ICML*, 2019.

[41] Mohammad Mahfujur Rahman, Clinton Fookes, Mahsa Baktashmotlagh, and Sridha Sridharan. Correlation-aware adversarial domain adaptation and generalization. *Pattern Recognit.*, 100:107124, 2020.

[42] Shiv Shankar, Vihari Piratla, Soumen Chakrabarti, Siddhartha Chaudhuri, Preethi Jyothi, and Sunita Sarawagi. Generalizing across domains via cross-gradient training. In *ICLR*, 2018.

[43] Chiheb Trabelsi, Olexa Bilaniuk, Ying Zhang, Dmitriy Serdyuk, Sandeep Subramanian, João Felipe Santos, Soroush Mehri, Negar Rostamzadeh, Yoshua Bengio, and Christopher J. Pal. Deep complex networks. In *ICLR*, 2018.

[44] Yusuke Tsuzuku and Issei Sato. On the structural sensitivity of deep convolutional networks to the directions of fourier basis functions. In *CVPR*, 2019.

[45] Vladimir Vapnik. *Statistical learning theory*. Wiley, 1998.

[46] Riccardo Volpi and Vittorio Murino. Addressing model vulnerability to distributional shifts over image transformation sets. In *ICCV*, 2019.

[47] Riccardo Volpi, Hongseok Namkoong, Ozan Sener, John C. Duchi, Vittorio Murino, and Silvio Savarese. Generalizing to unseen domains via adversarial data augmentation. In *NeurIPS*, 2018.

[48] Haohan Wang, Songwei Ge, Zachary C. Lipton, and Eric P. Xing. Learning robust global representations by penalizing local predictive power. In *NeurIPS*, 2019.

[49] Haohan Wang, Zexue He, Zachary C. Lipton, and Eric P. Xing. Learning robust representations by projecting superficial statistics out. In *ICLR*, 2019.

[50] Haohan Wang, Xindi Wu, Zeyi Huang, and Eric P. Xing. High-frequency component helps explain the generalization of convolutional neural networks. In *CVPR*, 2020.

[51] Jindong Wang, Cuiling Lan, Chang Liu, Yidong Ouyang, Tao Qin, Wang Lu, Yiqiang Chen, Wenjun Zeng, and Philip Yu. Generalizing to unseen domains: A survey on domain generalization. *IEEE Transactions on Knowledge and Data Engineering*, 2022.

[52] Zijian Wang, Yadan Luo, Ruihong Qiu, Zi Huang, and Mahsa Baktashmotlagh. Learning to diversify for single domain generalization. *CoRR*, 2021.

[53] Qinwei Xu, Ruipeng Zhang, Ya Zhang, Yanfeng Wang, and Qi Tian. A fourier-based framework for domain generalization. In *CVPR*, 2021.

[54] Xiang Xu, Xiong Zhou, Ragav Venkatesan, Gurumurthy Swaminathan, and Orchid Majumder. d-sne: Domain adaptation using stochastic neighborhood embedding. In *CVPR*, 2019.

[55] Zhenlin Xu, Deyi Liu, Junlin Yang, Colin Raffel, and Marc Niethammer. Robust and generalizable visual representation learning via random convolutions. In *ICLR*, 2021.

[56] Yanchao Yang and Stefano Soatto. FDA: fourier domain adaptation for semantic segmentation. In *CVPR*, 2020.

[57] Dong Yin, Raphael Gontijo Lopes, Jonathon Shlens, Ekin Dogus Cubuk, and Justin Gilmer. A fourier perspective on model robustness in computer vision. In *NeurIPS*, pages 13255–13265, 2019.

[58] Hongyi Zhang, Moustapha Cissé, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *ICLR*, 2018.

[59] Long Zhao, Ting Liu, Xi Peng, and Dimitris N. Metaxas. Maximum-entropy adversarial data augmentation for improved generalization and robustness. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *NeurIPS*, 2020.

[60] Kaiyang Zhou, Yongxin Yang, Timothy M. Hospedales, and Tao Xiang. Deep domain-adversarial image generation for domain generalisation. In *AAAI*, 2020.

[61] Kaiyang Zhou, Yongxin Yang, Timothy M. Hospedales, and Tao Xiang. Learning to generate novel domains for domain generalization. In *ECCV*, 2020.

[62] Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. Domain generalization with mixstyle. In *ICLR*, 2021.