Attention Consistency on Visual Corruptions for Single-Source Domain Generalization (Supplementary Material)

Ilke Cugu[1], Massimiliano Mancini[1], Yanbei Chen[1], Zeynep Akata[1,2]
[1]University of Tübingen, [2]MPI for Intelligent Systems
{ilke.cugu, massimiliano.mancini, yanbei.chen, zeynep.akata}@uni-tuebingen.de

Here, we present more details on our experiments. We first provide detailed information on the hardware used for the experiments and the licenses of the datasets in Section A. Then, in Section B, we discuss the severity levels determined for our Fourier-based visual corruptions.

A. Training Details

Training GPUs. All experiments are run by using 4× NVIDIA Quadro RTX 6000s, and 1× NVIDIA V100.

Dataset	Licence
PACS	Not available
COCO	Creative Commons Attribution 4.0 License
DomainNet	Custom: Non-commercial Research and Educational Purposes

Table A. The datasets employed in the paper and their licences.

Licences of the datasets. In Table A, we provide license information of the datasets we use in our experiments. Note that, for PACS, we did not find any attached license, but the dataset is publicly available[1].

B. Fourier-based Visual Corruptions

In this work, we propose to use three additional visual corruptions based on post-Fourier transform components along with ImageNet-C operations. ImageNet-C comes with 5 severity levels for each operation. For compatibility, we also defined 5 severity levels for each Fourier-based transformation. For all Fourier-based corruptions, we first set the highest severity level through visual inspection, ensuring that the images are highly corrupted but the objects are still easily recognizable to a human observer. Then, the intermediate levels are determined by dividing the interval between the clean image and the highest severity level into five equal parts. In the following we provide a summary of the transformations and their respective severity levels. We use $\mathcal{F}(X)$ to denote the Fourier transform of an image X, with $\mathcal{F}^A(X)$ its amplitude and with $\mathcal{F}^P(X)$ its phase.

[1]The dataset can be downloaded from the scripts in https://github.com/liyiying/Feature_Critic.

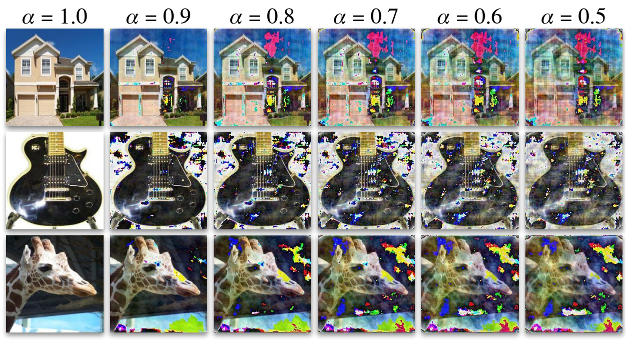

Figure A. Sample images of **phase scaling** corruption for 5 severity levels. The intensity of corruption increases from left → right.

Phase Scaling. Given a random scalar $\alpha \in (0, 1]$, this corruption uses α to scale the phase component, computing:

$$\phi_{\text{P-scaling}}(X) = \mathcal{F}^{-1}([\mathcal{F}^A(X), \alpha\mathcal{F}^P(X)]), \tag{1}$$

where $\mathcal{F}^{-1}$ is the inverse Fourier transform, and the phase is computed by:

$$\mathcal{F}^P_{u,v}(X) = \arctan\left(\frac{R(\mathcal{F}_{u,v}(X))}{I(\mathcal{F}_{u,v}(X))}\right) \tag{2}$$

where R is the real part, and I is the imaginary part of $\mathcal{F}(X)$. In this work, we set minimum value of α as 0.5, and the severity levels as $\{0.9, 0.8, 0.7, 0.6, 0.5\}$. In Figure A, we show the difference between the severity levels.

Constant Amplitude. This corruption replaces $\mathcal{F}^A$ with a constant $\beta \in (0, 1]$, computing the corrupted image as:

$$\phi_{\text{constant-A}}(X) = \mathcal{F}^{-1}([\beta, \mathcal{F}^P(X)]), \tag{3}$$

where the amplitude is computed by:

$$\mathcal{F}^A_{u,v}(X) = \sqrt{R^2(\mathcal{F}_{u,v}(X)) + I^2(\mathcal{F}_{u,v}(X))} \tag{4}$$

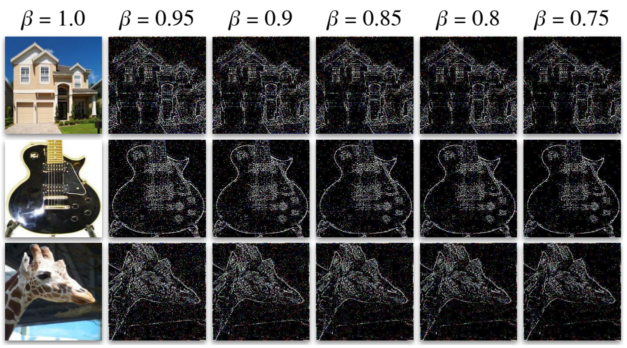

Figure B. Sample images of **constant amplitude** corruption for 5 severity levels. The intensity of corruption increases from left $\rightarrow$ right.

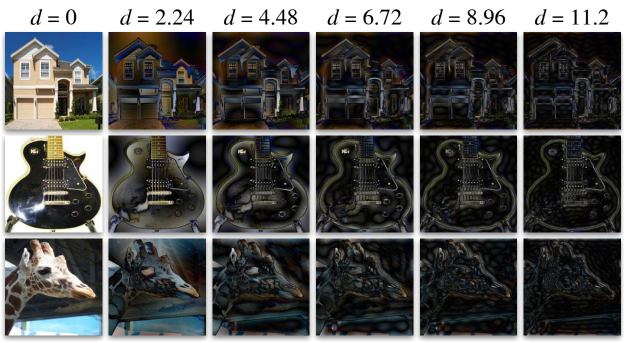

Figure C. Sample images of **high pass filter** corruption for 5 severity levels. The intensity of corruption increases from left $\rightarrow$ right.

In our experiments, β can be $\{0.95, 0.9, 0.85, 0.8, 0.75\}$, with 0.75 being the maximum corruption level. In Figure B, we show the visual effects of these values.

High pass Filter. This transformation corrupts the input image with a high pass filter via frequency windows. It filters out low frequency components by adjusting its diameter d on the centered Fourier spectrum. Formally:

$$\phi_{\text{high-pass}}(X) = \mathcal{F}^{-1}(H^d(\mathcal{F}(X)) \circ \mathcal{F}(X)), \tag{5}$$

where $H^d(F)$ a filtering mask where each spatial coordinate (u, v) has value:

$$H^d_{u,v}(F) = \begin{cases} 1, & \text{if} \quad F_{u,v} \geq d \\ 0, & \text{otherwise.} \end{cases} \tag{6}$$

where d is proportional to the dimensions of the input images. Since we use 224×224 images, the respective values of d are in the set $224 \times \{0.01, 0.02, 0.03, 0.04, 0.05\}$ where $224 \times 0.05 = 11.2$ is the maximum corruption level. In Figure C, we show the difference between the severity levels.