WaveDamp: Enhancing Natural Robustness in Endoscopy Through Wavelet-Based Frequency Damping

Supplementary Material

A. Appendix

A.1. Authentic Robustness

The robustness of DNNs is often assessed using artificially corrupted datasets such as ImageNet-C [19] and ImageNet-$\bar{\text{C}}$ [34]. These datasets consist of images that are incrementally perturbed to assess model stability across varying levels of corruption. These perturbed images retain the same underlying semantics as the original images, since they are created from the same clean image. In real-world scenarios, however, models are evaluated on images that may exhibit diverse semantics *in addition to* distortions. Therefore, model evaluation should extend beyond strictly measuring stability under distortions, and should also encompass the assessment of generalizability in the presence of such distortions. During the development of DNNs, model checkpoints are typically selected based on the performance on a validation dataset and evaluated on a test set, which has not been encountered during training. Nevertheless, images in robustness test sets like ImageNet-C are corrupted versions of the images used for validation, which raises concerns about potential data leakage. In order to mitigate this, the robustness datasets utilized in this study contain images that are not used for validation. Additionally, while the images in the quality triplets in this work are from the same patients, they possess semantics that are sufficiently different to assess both model robustness and generalizability.

A.2. Pre-Trained Weights

The results in Table 5 demonstrate the increased performance when utilizing in-domain pre-trained GastroNet-5M weights over ImageNet-1K weights. Since the ImageNet-1K weights are not in-domain, the initial learning rate is increased to 10^{-4} to allow for more flexibility. The largest differences are observed on the blind IQ triplet, where the model initialized with GastroNet weights improves the AUC by up to 0.139. The in-domain pre-training shows superiority due to the initialized domain relevant features obtained during self-supervised learning, as discussed in [4].

A.3. Training Details of the Frequency Damper

The damper is trained using a loss, $\mathcal{L}$, which is a combination of the TV loss of the output image and the L1 reconstruction loss with respect to the original input image. Because of this combination, the frequency damper tries to achieve the optimal balance between preserving significant semantics and reducing HF texture. Mathematically, the total loss is formulated as:

$$\mathcal{L} = \frac{\text{TV loss}}{2 \times W \times H} + \text{L1 loss}, \tag{4}$$

where the TV loss and L1 loss are defined as

$$\text{TV loss} = \sum_{w,h} \Big(|I_{w+1,h} - I_{w,h}| + |I_{w,h+1} - I_{w,h}| \Big) \tag{5}$$

and

$$\text{L1 loss} = \frac{1}{W \times H} \sum_{w,h} |I_{input,w,h} - I_{output,w,h}|, \tag{6}$$

respectively. Here, $I_{w,h}$ is the pixel intensity for the image I of width W and height H at position w, h. The division by $2 \times W \times H$ in the total loss is necessary to normalize the TV loss to approximately the same order of magnitude as the L1 loss. The total loss is aggregated across the 3 channel dimensions of the images, but this is omitted from the equations for the sake of simplicity.

The damper is trained for 9 epochs using the Adam optimizer. The initial learning rate is set to 10^{-3} and decreased by a factor of 10 after every 3 epochs. A batch size of 128 is used during training. The damper is trained on clean images with standard data augmentation, including random resized cropping and horizontal flipping of the images. After the damper training is completed, the weights are frozen, and the damper is used as an augmentation transformation during model training.

A.4. Deterministic and Stochastic WaveDamp

Table 6 shows the performance enhancements obtained using the stochastic properties introduced in WaveDamp compared against its deterministic variant. The results clearly show that the increased randomness boosts generalization and robustness on all test sets.

A.5. Bias Set Details

The generalizability of the models is examined using 3 different bias sets. These biases explored include the shape bias, LF bias and HF bias, and are created using 5 different severity levels. The shape and LF bias focus on the texture-shape trade-off in CNNs, which often utilize HF texture information to achieve high classification accuracy [17]. Additionally, the HF-bias is investigated to evaluate model performance under LF distortions, such as underexposure and overexposure. The texture information is suppressed using

Table 5. Baseline evaluation using ImageNet-1K and GastroNet-5M initialization.

Dataset	Peak Performance	Curated IQ Triplet			Blind IQ Triplet		
	BM	HQ	MQ	LQ	HQ	MQ	LQ
ImageNet-1K	0.840 ± 0.019	0.902 ± 0.021	**0.915 ± 0.010**	**0.785 ± 0.021**	0.708 ± 0.007	0.673 ± 0.015	0.695 ± 0.023
GastroNet-5M	**0.888 ± 0.014**	**0.941 ± 0.012**	0.889 ± 0.014	**0.785 ± 0.013**	**0.847 ± 0.014**	**0.776 ± 0.015**	**0.762 ± 0.022**

Table 6. Evaluation of deterministic and stochastic WaveDamp, demonstrating the obtained performance enhancements due to the introduced stochastic properties.

Method	Peak Performance	Curated IQ Triplet			Blind IQ Triplet		
	BM	HQ	MQ	LQ	HQ	MQ	LQ
Deterministic	0.902 ± 0.011	0.940 ± 0.012	0.902 ± 0.012	0.803 ± 0.014	0.835 ± 0.013	0.772 ± 0.018	0.771 ± 0.029
Stochastic	**0.922 ± 0.017**	**0.956 ± 0.015**	**0.933 ± 0.016**	**0.842 ± 0.011**	**0.853 ± 0.013**	**0.807 ± 0.007**	**0.846 ± 0.009**

Table 7. Overview of the parameters used to create the various bias test sets for each severity level.

Severity	1	2	3	4	5
Shape bias	$w = 0.05$	$w = 0.10$	$w = 0.20$	$w = 0.50$	$w = 1.00$
LF bias	$\sigma = 25$	$\sigma = 20$	$\sigma = 15$	$\sigma = 10$	$\sigma = 5$
HF bias	$\sigma = 1$	$\sigma = 2$	$\sigma = 3$	$\sigma = 4$	$\sigma = 5$

total variation minimization [5] utilizing various weight coefficients (w) depending on the severity. The HF and LF information is reduced using Gaussian low-pass and high-pass filters featuring various standard deviation values (σ). An overview of the parameters used for the different transformations is shown in Table 7. Visual examples are presented in Fig. 7, illustrating how the images are processed across the different severity levels. Fig. 8 demonstrates an overview of the model performance across all severity levels of the bias sets, highlighting that the models trained using WaveDamp experience the least amount of performance reduction.

A.6. Additional Experiments

The following section provides further information about the additional experiments.

A.6.1. Segmentation

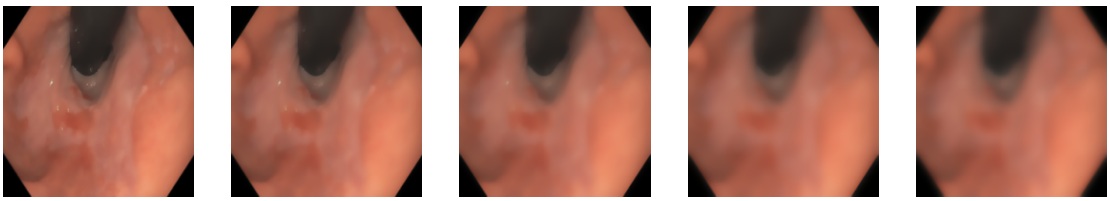
In order to evaluate the performance of WaveDamp on a public dataset, we conduct additional experiments using the Kvasir-SEG segmentation dataset [27]. Kvasir-SEG contains images of colorectal polyps, contributing to the development of automated segmentation models. Since this dataset is centered around localization, our proposed WaveDamp method is only compared against the baseline and APR-P method, since the other methods utilize mixing operations which are not designed for segmentation. For the experiments, the suggested validation split is adopted as the test set, while we randomly sample 120 validation images from the remaining training data. To evaluate the robustness of trained models, artificially corrupted test sets are created using perturbations from ImageNet-C that reflect the corruptions encountered in practice, including motion blur, brightness, and contrast. The Dice coefficient is employed to evaluate model performance. The results presented in Table 8 show that WaveDamp achieves comparable performance as the APR-P method, indicating the damping does not further enhance the robustness. This limitation could be attributed to the restricted dataset size in combination with the segmentation task, where edge information significantly influences the performance.

A.6.2. Blur Augmentation

A legitimate question might be: why not simply apply Gaussian blurring to images to create a more robust model? To address this, we replaced the damper architecture of the augmentation pipeline with a Gaussian filter, utilizing a kernel size of 9 and a standard deviation of 3. This setup generates images visually comparable to those produced by the frequency damper. The results of this experiment, shown in Table 9, highlight the limitations of this method. While employing a blurring kernel does increase the robustness on curated LQ images, it reduces performance on clean images when compared to WaveDamp. Moreover, WaveDamp consistently outperforms the blurring method on all other test sets, indicating the significance of selective HFC suppression.

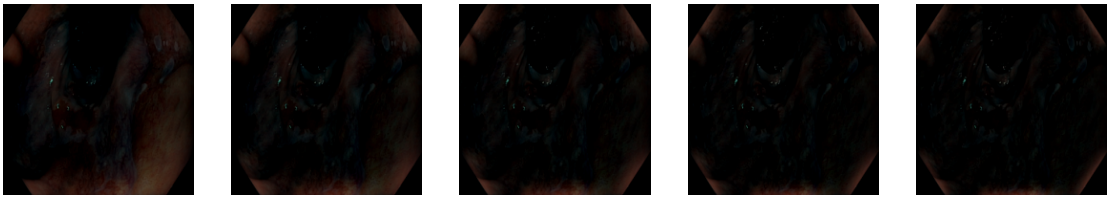
A.6.3. Transformers

In addition to the ResNet architecture, we train a ViT-Small vision transformer [13] following the same experimental setup as provided in Section 3.4. The results presented in Table 10 indicate that WaveDamp does not consistently increase the performance, which could be attributed to the inherent robustness of transformers discussed in [2]. Further exploration of transformer and hybrid CNN-transformer architectures is left for future work.

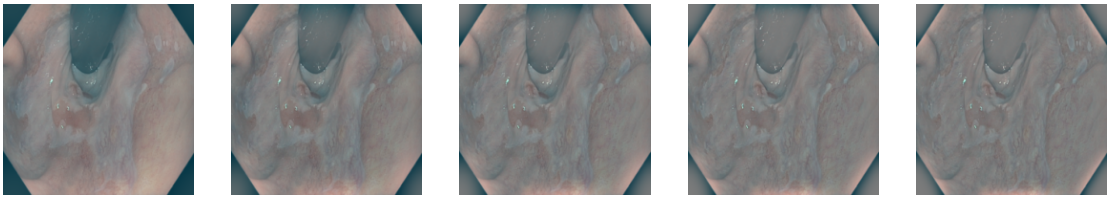


(a) Shape bias (texture removal)

(b) Low-frequency bias (HF removal)



(c) High-frequency bias (LF removal)



(d) High-frequency bias (normalized)

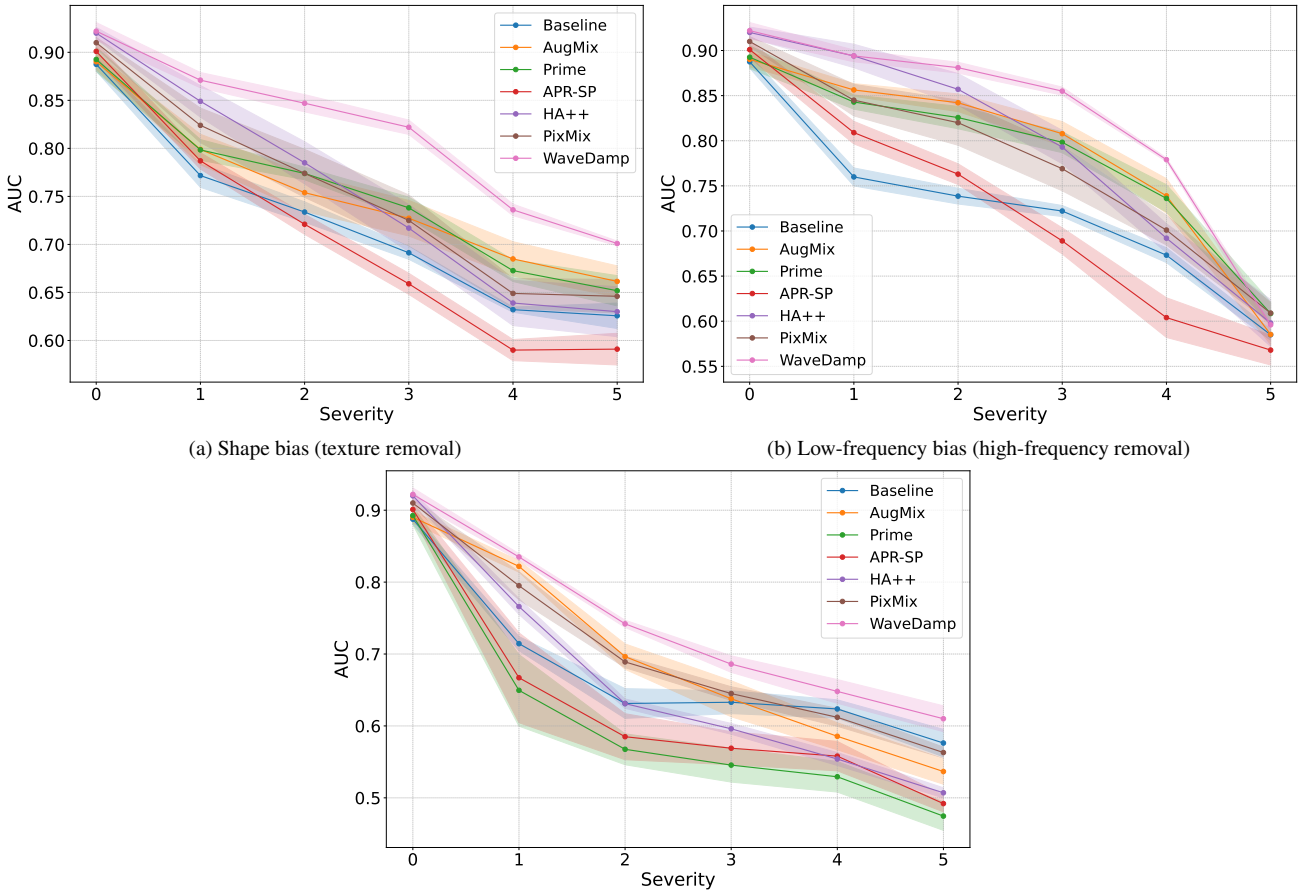
Figure 7. Visual examples of the different images created in the bias experiments with increasing severity (1-5) from left to right. The high-pass filtered images are normalized for visualization purposes only.

Table 8. Segmentation performance on the Kvasir-SEG dataset.

Method	**Clean**	**Corruption**		
		Motion Blur	Brightness	Contrast
Baseline	**0.879 ± 0.008**	0.714 ± 0.182	0.831 ± 0.027	0.548 ± 0.288
APR-P [7]	0.878 ± 0.007	**0.787 ± 0.092**	0.848 ± 0.018	0.655 ± 0.238
WaveDamp	0.878 ± 0.002	0.782 ± 0.104	**0.852 ± 0.014**	**0.663 ± 0.249**

Table 9. Comparison of a Gaussian filter and the proposed frequency damping.

Method	**Peak Performance**	**Curated IQ Triplet**			**Blind IQ Triplet**		
	BM	HQ	MQ	LQ	HQ	MQ	LQ
Blur	0.909 ± 0.020	0.947 ± 0.012	0.921 ± 0.011	**0.851 ± 0.018**	0.844 ± 0.011	0.779 ± 0.006	0.813 ± 0.014
WaveDamp	**0.922 ± 0.017**	**0.956 ± 0.015**	**0.933 ± 0.016**	0.842 ± 0.011	**0.853 ± 0.013**	**0.807 ± 0.007**	**0.846 ± 0.009**



(a) Shape bias (texture removal)

(b) Low-frequency bias (high-frequency removal)

(c) High-frequency bias (low-frequency removal)

Figure 8. Performance degradation for different severities of the bias test sets, showing both the mean and standard deviation of models trained using 5-fold cross validation for each severity grade.

Table 10. Experimental results of the proposed WaveDamp method on a ViT-Small model.

Method	**Peak Performance**	**Curated IQ Triplet**			**Blind IQ Triplet**		
	BM	HQ	MQ	LQ	HQ	MQ	LQ
Baseline	**0.889 ± 0.008**	**0.949 ± 0.001**	0.887 ± 0.010	**0.781 ± 0.022**	**0.736 ± 0.024**	**0.695 ± 0.021**	0.651 ± 0.062
WaveDamp	0.879 ± 0.008	0.948 ± 0.012	**0.894 ± 0.008**	0.770 ± 0.020	0.703 ± 0.038	0.684 ± 0.034	**0.692 ± 0.062**