Jurnal Ilmu Komputer dan Informasi (Journal of Computer Science and Information)
19/1 (2026), 33-52. DOI: http://dx.doi.org/10.21609/jiki.v19i1.1558

Enhanced Robustness in Image Classification through DistortionMix: A Hybrid Distortion-Based Augmentation Technique

Husni Fadhilah[1], Budi Warsito[2], Hasna Faridah[3]

[1] School of Electrical Engineering and Informatics, Bandung Institute of Technology, Bandung, Indonesia
[2] Department of Statistics, Diponegoro University, Semarang, Indonesia
[3] Department of Statistics, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia

Email:23523034@std.stei.itb.ac.id

Abstract

Deep neural networks perform well on clean image classification tasks but often fail under common corruptions and distribution shifts. This paper introduces DistortionMix, a lightweight hybrid distortion-based augmentation technique designed to improve model robustness. It randomly applies contrast variation, Gaussian noise, or impulse noise to training images, enhancing data diversity and encouraging resilient feature learning. We evaluate DistortionMix on CIFAR-10 (clean) and CIFAR-10-C (corrupted), which includes 19 corruption types at five severity levels. A variety of architectures e.g ResNet, DenseNet, EfficientNet, MobileNet, VGG, AlexNet, GoogleNet, and ViT are fine-tuned with and without DistortionMix. Experimental results show that DistortionMix improves corrupted accuracy by up to 13.8%, while maintaining or slightly improving clean accuracy. Among all models, ViT-Base (timm) achieves the highest robustness, reaching 89.4% on severe corruptions and 97.43% on clean data. These findings highlight DistortionMix as a simple yet effective strategy for enhancing out-of-distribution generalization. Future work includes extending distortion types, developing adaptive augmentation policies, and evaluating performance on real-world corrupted datasets. Source code: github.com/HusniFadhilah/DistortionMix.

Keywords: *Robustness, data augmentation, image corruptions, CIFAR-10-C, vision transformer.*

1. Introduction

Deep learning has led to remarkable advances in image recognition, with architectures like ResNets, EfficientNets, and Vision Transformers achieving or even surpassing human-level accuracy on standard benchmarks. For instance, ResNet-152 networks can exceed 95% accuracy on CIFAR and reach human-level performance on ImageNet when training and testing data are drawn from the same distribution. However, model performance degrades significantly when inputs are corrupted or come from a shifted distribution not seen in training [2], [27]. In real-world deployment, images can be affected by noise, blur, weather effects, or compression artifacts that cause a deep network to misclassify examples it would normally recognize. Hendrycks and Dietterich showed that a classifier's error on ImageNet can skyrocket from 22% on clean images to 64% on the corrupted ImageNet-C benchmark [2]. This fragility under common corruptions highlights the need for improved robustness in image classification models.

One approach to address this challenge is *domain generalization* [21], [33], which trains models to be more invariant to distribution shifts without seeing target-domain examples. These methods aim to narrow the performance gap between training on clean data and testing on unknown corruptions. For example, Lee and Myung [17] propose augmenting images in the frequency domain (varying phase information) to force models to rely on domain-invariant features, thereby improving robustness to input perturbations. Similarly, other works modify Fourier amplitude while keeping phase fixed (or vice versa) to simulate realistic distribution shifts. Chen *et al.* and Xu *et al.* [32] in 2021 introduced Fourier-based augmentation strategies that randomize image amplitudes across mini-batches while preserving

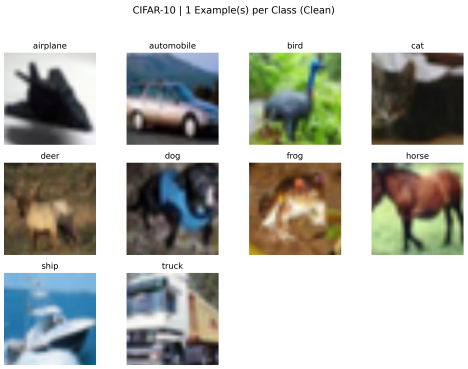

Figure 1. Clean examples from the CIFAR-10 dataset showing 10 object classes without corruption. Images illustrate the visual diversity and intra-class variation present in the dataset.

phase, effectively creating new corrupt variations for training. These methods help models learn features that remain stable under corruption, but they often require careful tuning of frequency perturbation strength to avoid harming clean accuracy.

A more direct and widely used strategy for robustness is *data augmentation* during training [6], [31]. Traditional augmentation includes geometric transforms (crops, flips, rotations) and color jitter, which improve generalization but may not cover the breadth of possible corruptions (e.g., random noise or blur). Recent years have seen an explosion of advanced augmentation techniques. AutoAugment [3] uses reinforcement learning to find an optimal policy of augmentations (e.g., shearing, rotating, color shifts) that maximizes validation accuracy. This improved generalization on clean data and even provided some robustness gains, but the learned policies tend to focus on mild distortions and require heavy search computation. RandAugment [4] simplified this by randomly applying a fixed number of distortions with variable magnitude, eliminating the search phase while still achieving competitive results. While these automated augmentation strategies were not explicitly designed for corruption robustness, they increase dataset diversity and can incidentally improve performance on moderately perturbed data [6], [31].

To specifically target robustness against common corruptions, researchers have proposed augmentation methods that simulate the kinds of noise and artifacts seen in benchmarks like CIFAR-10-C and ImageNet-C. AugMix [11] is a seminal technique in this vein. AugMix generates diverse samples by stochastically composing multiple simple augmentations (e.g., translations, posterize, contrast) and then linearly mixing the results with the original image. A Jensen-Shannon consistency loss is also used to encourage the network's predictions to be invariant to different augmentations of the same image. This approach achieved state-of-the-art robustness on CIFAR-10-C and CIFAR-100-C, roughly halving the mean corruption error relative to standard training. AugMix demonstrated the importance of *augmenting with multiple distortions at once* to cover a broad corruption space, as well as enforcing prediction consistency to prevent the model from becoming too sensitive to input perturbations. Following AugMix, many works have extended the idea of mixing or diversifying augmentations. For example, DeepAugment [12] creates corrupted images by perturbing the parameters of image generation networks and autoencoders. By passing training images through neural networks with random weights (which produce novel artifacts like synthetic noise or blur) and combining this with AugMix, DeepAugment further improved robustness on ImageNet-C, indicating that unusual, learned distortions can complement handcrafted ones. TrivialAugment [20] took an opposite approach, randomly applying a single augmentation selected from a wide set, without policy search, yet still improved accuracy and corruption robustness simply by not restricting augmentation variety. The success of these methods suggests that combining multiple simple distortions or intelligently randomizing augmentations can yield models that generalize better to unforeseen corruptions.

Another line of work has explored adding *noise-based augmentations* to harden models against corrupting noise. For instance, adding Gaussian noise or salt-and-pepper noise (impulse noise) during training can desensitize networks to those perturbations at test time [30], [35]. Rusak *et al.* [24] showed that augmenting ImageNet with simple noise such as Gaussian, shot, and impulse noise drastically improved robustness to these and even other types of corruptions. In fact, their method produced networks that nearly matched AugMix on many corruption benchmarks by training on noise alone. Similarly, Lopes *et al.* [19] proposed *Patch Gaussian* augmentation, which injects local Gaussian noise patches into images during training. They found this technique improved robustness without sacrificing clean accuracy, suggesting that stochastic noise injections act as a form of regularization that helps models ignore high-frequency spurious signals. Beyond noise, other augmentation methods address specific corruption types: for example, Zhang *et al.* [34] introduced *Tilted* convolutional kernels to make CNNs invariant to small image shifts, which helps with translation and blur robustness. Adversarial augmentation strategies have also been investigated, Gong *et*

al. [8] proposed *MaxUp*, which chooses the worst-case outcome among multiple augmented versions of an image for training, effectively performing a mild adversarial training over augmentation space. MaxUp was shown to improve generalization and robustness by preventing the network from taking "shortcuts" on easy augmented examples.

More recently, researchers are combining these ideas into unified frameworks. Modas *et al.* [18] introduced PRIME, a set of *few primitive* augmentations (e.g. noise, blur, weather, digital distortions) drawn from a max-entropy distribution. By randomly applying these primitives with varying strengths and mixing as in AugMix, PRIME achieves state-of-the-art robustness on multiple benchmarks, surpassing AugMix alone. PRIME's success underlines the value of covering a large distortion space with simple, high-entropy augmentations. Another trend is leveraging the robustness properties of different architectures. Vision Transformers (ViTs) have been found to exhibit stronger robustness than CNNs on some corruption benchmarks [1]. Paul and Chen [23] showed that ViTs pre-trained on large datasets maintain higher accuracy under common input perturbations than ResNets of similar accuracy on clean data. This is hypothesized to stem from ViTs' ability to focus on more global image features (like object shape) instead of local texture, which aligns with observations by Geirhos *et al.* [7] that CNNs are overly biased toward local textures. In parallel, using larger or more diverse pre-training data has improved corruption robustness: pre-training on billions of images (e.g., JFT-300M or ImageNet-21K) yields models that are inherently more resilient to distribution shift, likely because they have seen a wider variety of conditions. These advances point toward combining architectural choices, large-scale pre-training, and data augmentation to achieve robust models.

This paper proposes DistortionMix, a lightweight and effective data augmentation strategy that improves robustness of image classifiers under common corruptions. The key contributions are:

- DistortionMix Augmentation, that randomly applies one of several hand-crafted distortions, contrast adjustment, Gaussian noise, or impulse noise, to each training image. It requires no policy learning or extra model components, making it efficient and easy to implement.
- Comprehensive evaluation conducted on CIFAR-10, CIFAR-10-C, and its performance is also validated on the real-world ImageNet-A benchmark to assess generalization beyond synthetic corruptions. DistortionMix is tested across 15+ architectures, including ResNet, DenseNet, EfficientNet, MobileNet, VGG, ViT, AlexNet, and GoogleNet. This broad evaluation identifies model families that benefit most from distortion-based augmentation.
- Robustness improvement of model accuracy on corrupted data (CIFAR-10-C, severity 5) by 1–2 percentage points across most networks. DenseNet-161 achieves the best performance, with 80.8% accuracy under severe corruption and 94.79% on clean data.
- The trade-offs involved in using DistortionMix are analyzed, including its impact on clean accuracy, which is shown to be minimal, and the additional training time required. DenseNet-161's performance is also broken down by corruption type to identify distortions that remain challenging (e.g., heavy noise and pixelation) and those that are largely addressed by the proposed approach.

The remainder of the paper is organized as follows: Section II details the datasets and corruption types, the DistortionMix augmentation procedure, training hyperparameters, and model architectures. Section III presents the experimental results, including tables of overall accuracy and robustness, a comparison of training times, and visualization of model-wise and corruption-wise performance. The results are discussed, along with insights into why certain architectures perform better and how DistortionMix influences generalization. Section IV concludes the paper and outlines future research directions, such as leveraging more real-world corruptions, tuning augmentation severity automatically, and applying the method to other domains like medical or satellite imagery.

2. Methodology

DistortionMix randomly applies one of three distortion types, e.g contrast, impulse noise, and Gaussian noise during training. These distortions were selected for their diversity in effect (intensity vs. noise-based), computational efficiency, and ability to simulate common real-world degradations while maintaining training stability. Our choice aims to strike a balance between distortion strength and simplicity without introducing heavy computational overhead or policy learning.

2.1. Datasets and Corruption Types

Experiments are conducted on the CIFAR-10 dataset for training and clean evaluation, and the

CIFAR-10-C dataset for evaluating robustness to corruptions. CIFAR-10 consists of 60,000 32×32 color images (50,000 train and 10,000 test) across 10 object classes (airplane, automobile, bird, cat, deer, dog, frog, horse, ship, truck). The standard training/test split and original labels are used for fine-tuning all models.

For robustness evaluation, CIFAR-10-C, introduced by Hendrycks and Dietterich [10], is used, which applies a variety of common corruptions to the original CIFAR-10 test images. CIFAR-10-C allows us to test models on unseen perturbations in a controlled way. It includes 19 corruption types, each with 5 levels of severity (severity 1 = mild corruption, severity 5 = most extreme). The corruption types span four broad categories:

- *Noise*: **Gaussian noise**, **shot noise** (Poisson noise), **impulse noise** (salt-and-pepper). These add independent random fluctuations to pixel values.
- *Blur*: **defocus blur**, **motion blur**, **zoom blur**, **glass blur** (image seen through a glass diffraction), and **Gaussian blur**. These distortions smear or smooth out details in the image.
- *Weather*: **frost** (overlay of frost patterns), **fog** (reduced contrast as if in fog), **snow** (white snow particles), and **spatter** (rain or mud droplets). These simulate environmental conditions that obscure the image.
- *Digital*: **brightness**, **contrast**, **saturate** (color saturation shift), **JPEG compression** artifacts, and **pixelate** (rescaling to low resolution and back up).

Each corruption type in CIFAR-10-C is applied to every test image at each severity level, resulting in 5 versions of the 10,000 test images for each corruption (for a total of 50,000 corrupted images per type). Following standard practice [10], model robustness is evaluated using the highest severity level (Level 5) as the most stringent test of performance under extreme corruption. Accuracy is reported on CIFAR-10-C at severity 5, along with per-corruption accuracy for the best-performing model to diagnose strengths and weaknesses.

2.2. Hybrid Distortion Augmentation: DistortionMix

The method **DistortionMix** is proposed as a data augmentation strategy that randomly applies one of several distortion operations to each training image. The goal is to expose the model to a mix of noise and intensity perturbations similar to those in CIFAR-10-C (specifically in the noise and digital categories) during training, thereby improving robustness. DistortionMix focuses on three augmentation primitives:

1) **Contrast variation**: Contrast adjustment is applied by converting the image to a PyTorch tensor and modifying its contrast. The deviation from the mean is multiplied by a random factor α, then the mean is added back: $x \leftarrow (x - \bar{x}) \times \alpha + \bar{x}$. The contrast factor α is sampled uniformly from a range $[0.5, 1.5]$, resulting in the image becoming up to 50% less or more contrasted than the original. This augmentation emulates the *brightness/contrast/saturate* type corruptions by producing washed out or high-contrast images. Pixel values are constrained to remain within the valid range [0,1] after scaling (by clipping or appropriate casting).
2) **Gaussian noise**: Random Gaussian noise is added to the image pixels. Specifically, i.i.d. noise $\delta \sim \mathcal{N}(0, \sigma^2)$ is sampled for each pixel and added to the image: $x \leftarrow x + \delta$. A relatively mild standard deviation of $\sigma = 0.05$ is used in normalized pixel units (where pixel values are in [0,1]) to ensure the noise is visible but does not obscure the image entirely. After noise addition, pixel values are clipped to the [0,1] range. This augmentation models the *Gaussian noise* corruption in CIFAR-10-C, as well as other forms of sensor noise or grain.
3) **Impulse noise**: Salt-and-pepper noise is injected into the image by setting each pixel to either the minimum or maximum value with a small probability p. The implementation uses $p = 0.01$, meaning 1% of pixels are corrupted on average, half set to 0 (pepper) and half to 1 (salt). The intensity of the effect is amplified to ensure that corrupted pixels reach the full 0 or 255 scale (uint8), followed by clipping to maintain valid range. This models the *impulse noise* corruption, which significantly affects image quality even at low corruption rates (see Fig. 5 for impact on accuracy).

The procedure is formally defined as follows:

During training with DistortionMix, each input image in a mini-batch is probabilistically considered for augmentation, with the augmentation type selected accordingly. The probability of applying *contrast*, *Gaussian noise*, or *impulse noise* is set

Algorithm 1: DistortionMix Augmentation Procedure

Input: Image $x \in [0,1]^{H \times W \times C}$, probabilities p_c, p_g, p_i
Output: Augmented image x'

Function `DistortionMix`(x, $p_c = 0.1$, $p_g = 0.1$, $p_i = 0.1$)**:**
  $r \leftarrow \text{Uniform}(0, 1)$;
  if $r < p_c$ **then**
    return ApplyContrast(x);
  else
    if $r < p_c + p_g$ **then**
      return ApplyGaussianNoise(x);
    else
      if $r < p_c + p_g + p_i$ **then**
        return ApplyImpulseNoise(x);
      else
        return x;

ApplyContrast(x)**:** Adjust contrast by $\alpha \sim \mathcal{U}(0.5, 1.5)$: $x \leftarrow \text{clip}((x - \bar{x}) \cdot \alpha + \bar{x})$;
ApplyGaussianNoise(x)**:** Add $\epsilon \sim \mathcal{N}(0, 0.05^2)$ to x;
ApplyImpulseNoise(x)**:** Set 1% of pixels to 0 or 1 randomly (salt and pepper);

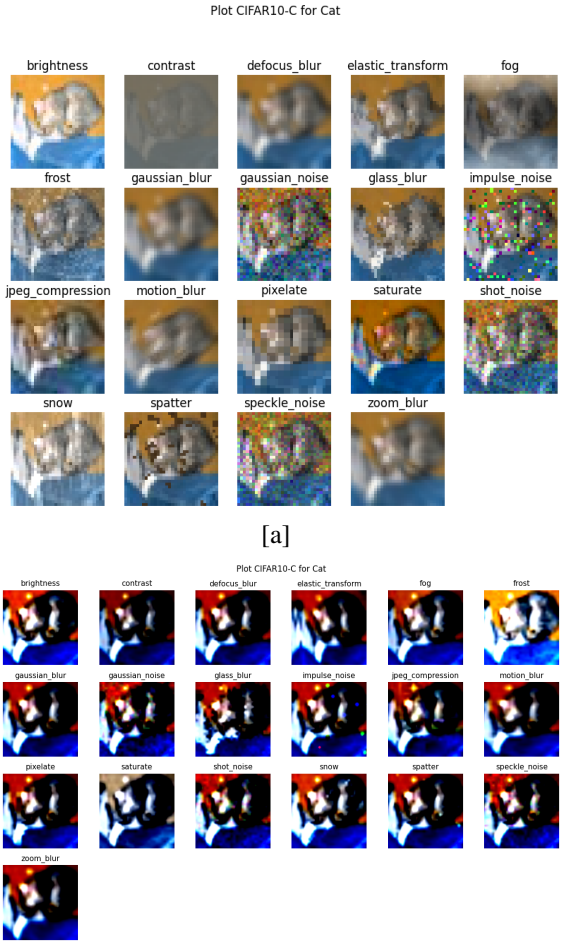

Figure 2. Comparison of corrupted examples for the CIFAR-10 "cat" class. (a) shows standard corruptions from CIFAR-10-C Robustbench; (b) shows mixed corruptions augmentation technique from DistortionMix.

to 0.1 (10%) each. These events are implemented as mutually exclusive for simplicity. Consequently, approximately 30% of images are augmented per epoch (10% via contrast adjustment, 10% via Gaussian noise, and 10% via impulse noise), while the remaining 70% remain in their original state. When an image is chosen for augmentation, one of the three distortion types is selected at random, with equal probability. A total augmentation probability of 0.3 has been found to strike a good balance: high enough to ensure periodic exposure to distorted examples, yet not so high as to dominate training with overly corrupted data, which can hinder convergence [15]. Only one augmentation is applied per image per iteration, though across epochs, different augmentations may be applied to the same image. This results in a batch composition that mixes distorted and undistorted samples, bearing similarity to AugMix [11], albeit without pixel-level mixing between clean and augmented versions.

DistortionMix operates as a *stochastic, on-the-fly* augmentation method. No fixed augmented dataset is precomputed; instead, new distortions are generated at each epoch using fresh random noise and contrast factors. Over the course of 50 epochs, each training image is observed under various conditions, unaltered in some, noised or contrast-modified in others, encouraging the model to learn features that are invariant to such changes. Unlike approaches that apply all augmentations simultaneously, the design of applying a single, randomly chosen distortion ensures exposure to both clean and isolated-corruption examples. This strategy draws partial inspiration from *randomized smoothing* in adversarial robustness, wherein training with random noise enhances prediction stability under noisy conditions [15]. By introducing randomness during training, the resulting model remains robust to perturbations even without augmentation at test time.

In the DistortionMix implementation, augmentation is applied after the standard random crop and horizontal flip, and before normalization (used in

networks expecting normalized input). The training pipeline for an image consists of: random crop (with 4-pixel padding, as is standard in CIFAR-10 training), random horizontal flip, then DistortionMix (contrast or noise), followed by conversion to tensor and normalization. All models, including Transformers, were trained on images resized to a higher resolution (either 224×224 or 256×256; see next subsection) to match ImageNet-pretrained architectures. Distortion operations were applied in pixel space after resizing, except for impulse noise, which was applied directly to the 224×224 tensor. DistortionMix does *not* rely on any specialized loss function or additional regularization beyond standard cross-entropy loss, its impact arises solely from modifications to the training data distribution.

2.3. Training Setup and Hyperparameters

All models were fine-tuned on CIFAR-10 using a consistent set of training hyperparameters to enable fair comparisons:

- **Optimizer**: Stochastic gradient descent (SGD) with momentum 0.9 was used for all experiments. This standard optimizer yielded stable convergence across all model types.
- **Learning rate and schedule**: An initial learning rate of 0.01 was chosen for fine-tuning. Although certain architectures, particularly Vision Transformers, often benefit from higher learning rates when trained from scratch, the value of 0.0001 proved effective for fine-tuning from ImageNet weights without divergence. A *Cosine Annealing* schedule was employed over 50 epochs, decaying the learning rate smoothly from 0.01 to 10^{-6}, promoting stable convergence in later epochs. No learning rate restarts were used.
- **Batch size**: 32 images per batch. While relatively small, this choice was suitable given the limited dataset size (50k CIFAR-10 images) and the substantial memory requirements of large models (such as DenseNet-161 and ViT-L32). Batch normalization layers in CNNs continued to operate as expected at this batch size, without freezing.
- **Number of epochs**: Fine-tuning was conducted over 100 epochs. Most models showed convergence well before epoch 50 on CIFAR-10, with training accuracy plateauing around epochs 20–30. Nonetheless, training continued with a decaying learning rate to achieve slight improvements in final test accuracy. Model parameters at epoch 50 were used for evaluation.
- **Input size**: CIFAR-10 images were resized to match the expected input resolution of each pretrained model. For all CNN-based architectures (e.g., ResNets, DenseNets, EfficientNets, MobileNets, VGG, AlexNet, and GoogleNet), images were first upsampled from 32×32 to 256×256, then randomly cropped to 224×224 during training. During evaluation, images were resized to 256×256 and center-cropped to 224×224. This follows standard ImageNet fine-tuning practice and ensures compatibility with models that were pretrained on 224-resolution inputs. For Vision Transformers (ViT) sourced from the `timm` library, two variants were used: `vit_small_patch16_224` and `vit_base_patch16_384`. The Small variant receives input of size 224×224, consistent with CNNs, while the Base variant expects 384×384 inputs. Accordingly, CIFAR-10 images were resized directly to the required input resolution (either 224×224 or 384×384) without additional cropping. Patch size for these models is 16×16, resulting in a grid of 14×14 (for 224 input) or 24×24 (for 384 input) tokens. This configuration matches the pretraining setup of the respective models. No extra datasets or external corruptions were used, only CIFAR-10 images, resized appropriately and augmented with either baseline transforms or DistortionMix.
- **Baseline augmentation**: For baseline training (without DistortionMix), standard random crop and horizontal flip augmentations were applied to CIFAR-10 images, using the same resizing strategy described above. No additional augmentations such as color jitter were included, allowing isolation of the effect introduced by distortion-based augmentations. This baseline setup aligns with typical fine-tuning procedures for pretrained models on CIFAR-10 and yields near state-of-the-art clean accuracy.
- **Pretrained weights**: All models were initialized with weights pretrained on ImageNet-1k (ILSVRC2012). For CNNs, weights were sourced from PyTorch's torchvision library, while ViT models utilized weights from the `timm` library. Leveraging pretrained weights significantly improved convergence speed and final performance on CIFAR-10. All layers of each model were fine-tuned, with

no layers frozen. Batch normalization layers remained in training mode and updated their running statistics using CIFAR-10 data (with momentum 0.1). No additional regularization methods such as weight decay were employed beyond those already present in the pretrained weights and batch normalization layers. Dropout layers, such as those in ViT or certain EfficientNet variants, remained active with their default dropout rates.

With this training setup, baseline models achieve strong clean accuracy on CIFAR-10 (generally $> 90\%$ for most architectures, as shown below), which is crucial for ensuring that differences in robustness are not simply due to underfitting or poor training. DistortionMix is applied only during training; during evaluation on both CIFAR-10 and CIFAR-10-C, no augmentations are applied (aside from the resizing/cropping mentioned earlier). Training times for each model were logged to quantify the overhead of different architectures and the computational cost of DistortionMix augmentation, which adds a small cost per image, mainly negligible compared to the network forward pass.

2.4. Model Architectures Evaluated

A diverse set of deep neural network architectures was evaluated to observe how DistortionMix impacts each and to identify which architectures are inherently more robust:

- **ResNets**, ResNet-18, ResNet-34, and ResNet-152 were included. These CNNs, introduced by He *et al.*[9], use residual skip connections to ease the training of very deep networks. ResNet-18/34 use basic 3×3 convolution blocks, while ResNet-50/101 use bottleneck blocks. The models range from 18 layers to 101 layers in depth. ResNets are a common baseline in robustness studies and typically achieve high clean accuracy, though prior work has shown they are vulnerable to corruptions without augmentation[22].
- **DenseNets**, DenseNet-121 and DenseNet-161 were tested. These CNNs, introduced by Huang *et al.* [14], feature dense connectivity between layers, where each layer receives all previous feature maps as input. This design promotes feature reuse and strong feature propagation. DenseNet-161 has 161 layers and a larger growth rate, making it one of the most accurate CNNs on ImageNet among those tested. It is hypothesized that dense feature usage may contribute to robustness, as features are aggregated from many layers.
- **EfficientNets**, EfficientNet-B0, B1, B2, and B3 were included. EfficientNets [29] are CNNs optimized through neural architecture search with a compound scaling rule to balance depth, width, and resolution for efficiency. B0 is the smallest model (with approximately 5.3M parameters) and B3 is larger (with 12M parameters). These models achieve high accuracy on ImageNet while using fewer parameters than ResNets or DenseNets, though their performance under corruptions is less known. These models were evaluated to determine if highly optimized architectures trade off robustness.
- **MobileNets**, MobileNet-V2 and MobileNet-V3 (both Small and Large variants) were used as examples of lightweight CNNs designed for mobile applications. MobileNet-V2 (Sandler *et al.*[25]) employs depthwise separable convolutions and an inverted residual structure for efficiency. MobileNet-V3 (Howard *et al.*[13]) further integrates NAS and squeeze-and-excitation modules to enhance V2. These models have low capacity (e.g., MobileNetV3-Small has 2.5M parameters), which may limit their ability to learn robust features. Tests were conducted to assess whether smaller models benefit from DistortionMix or if they underperform compared to larger models on CIFAR-10-C.
- **VGG16**, This older CNN architecture by Simonyan & Zisserman [26] is characterized by deep layers (16 layers) of plain convolutions and pooling, without residual connections. VGG16 has a high parameter count and tends to be less parameter-efficient than ResNets, but was included to observe how a simpler architecture handles augmentations. It typically has slightly lower clean accuracy than ResNet-50 on modern tasks and may be more sensitive to perturbations due to the absence of skip connections or normalization in early layers.
- **GoogleNet (Inception-V1)**, Inception v1 (Szegedy *et al.* [28]) introduced inception modules that split convolutions into multiple parallel filters. This was one of the first very deep networks (22 layers) to surpass human accuracy on ImageNet. GoogleNet was included to represent the Inception family. It uses smaller convolution filters and factorized convolution layers extensively, which may influence how noise propagates through

the network.

- **AlexNet**, The classic 8-layer CNN from Krizhevsky *et al.* [16] was included mainly for historical comparison. It has much lower capacity and uses large 11×11 and 5×5 kernels in the early layers, which may make it more invariant to some noise due to heavy pooling, but it is also less accurate overall. AlexNet was fine-tuned to assess how a low-capacity model benefits from DistortionMix.
- **Vision Transformers (ViT)**, ViT [5]) models were evaluated using the `timm` library, specifically the `vit_small_patch16_224.augreg_in21k_ft_in1k` and `vit_base_patch16_384` variants. These models process images by dividing them into patches (16×16 pixels) and applying transformer encoders to capture global context. The `vit_small_patch16_224` model operates on 224×224 input images, while `vit_base_patch16_384` uses 384×384 inputs. Both models were pretrained on ImageNet-21k and fine-tuned on ImageNet-1k. During our experiments, due to hardware constraints, ViT models were fine-tuned for 20 epochs with a batch size of 32, using the same training pipeline as other architectures. This setup allows for assessing the robustness and performance of ViT models on CIFAR-10 and CIFAR-10-C datasets.

For each model, the number of parameters is recorded, and key architectural features are noted, but all models are trained under the same pipeline as described. Table 1 in the next section will summarize the training time and clean/corrupted accuracy of each model with and without DistortionMix. By evaluating this broad palette of architectures, the aim is to answer:

1) Does DistortionMix consistently improve robustness regardless of model type?
2) Which model achieves the highest absolute robustness, and is this aligned with clean accuracy ranking?
3) Are there model-specific peculiarities (e.g., does a ViT gain less from augmentation because it is already robust, or do smaller models struggle to learn the augmentation noise patterns)?

Our evaluation attempts to be comprehensive in addressing these questions.

3. Results and Analysis

3.1. Overall Performance and Training Cost

The models are first compared in terms of standard test accuracy on CIFAR-10 (clean) and robustness accuracy on CIFAR-10-C at severity 5 (common corruptions), both with and without DistortionMix augmentation. Table 1 provides these results, along with the training time each model took for 50 epochs on the hardware (except ViT only 20 epoch because hardware limitation). The training time serves as a proxy for computational cost; it includes the overhead of DistortionMix (which was negligible in most cases) and differences due to model size.

3.1.1. Accuracy on Clean CIFAR-10. All models attain high accuracy on the clean test set after fine-tuning. Vit Timm Base 384 achieved the highest baseline clean accuracy at 96.89%, closely followed by Vit Timm Small 224 at 95.23%. DenseNet-161 also performed exceptionally well with a baseline of 92.79%, and ResNet-152 at 92.39%. Architectures like EfficientNet-B0 to B3, DenseNet-121, and GoogleNet (Incv1) generally fall within the $88 - 92\%$ range. The lowest clean accuracies were observed in MobileNetV3-Small (85.77%), VGG16 (88.02%), and MobileNetV2 (88.36%), which is expected given their older or more compact architectures.

Significantly, applying DistortionMix had a negligible impact on clean accuracy for most models. For instance, Vit Timm Base 384 saw a slight increase to 97.43%, and Vit Timm Small 224 improved to 95.55%. Most models either maintained or slightly improved their clean accuracy with DistortionMix, suggesting that the augmentation did not cause *overfitting* to corrupted images at the expense of clean performance. For example, ResNet-18 improved from 89.39% to 92.50% ($+3.11\%$), and DenseNet-161 from 92.79% to 95.00% ($+2.21\%$). Only AlexNet experienced a minor decrease of 0.81%, from 89.56% to 88.75%, which is minimal compared to the substantial gains seen in robustness. Overall, DistortionMix effectively maintains or slightly enhances *in-distribution* performance.

3.1.2. Accuracy on CIFAR-10-C (Severity 5). A clear and consistent pattern emerged where training with DistortionMix significantly improved the robustness of nearly every model. The corrupted accuracy with DistortionMix consistently surpassed the baseline for all models listed, indicated by a '+' sign.

Table 1. Comparison of model performance with and without DistortionMix augmentation. Training time is measured for 50 epochs. Accuracy is reported on CIFAR-10 (clean test set) and CIFAR-10-C (corrupted test set, severity 5). Only the highest clean and corrupted accuracy values across all models are **bolded**. Values in parentheses for ViT indicate the best run across trials.

Model	**Train Time (s)**	**Clean Accuracy (%)**		**Corrupted Accuracy (%, sev 5)**	
		Baseline	*+DistortionMix*	*Baseline*	*+DistortionMix*
AlexNet	3361	89.56	88.75 (-)	62.03	71.98 (+)
VGG16	20346	88.02	89.21 (+)	61.90	68.46 (+)
ResNet-18	4693	89.39	92.50 (+)	62.06	74.33 (+)
ResNet-34	7214	90.72	93.39 (+)	63.30	77.07 (+)
ResNet-152	31253	92.39	94.60 (+)	68.20	77.84 (+)
DenseNet-121	14598	91.39	94.04 (+)	66.44	77.42 (+)
DenseNet-161	32082	92.79	95.00 (+)	67.55	80.82 (+)
EfficientNet-B0	8928	88.72	91.88 (+)	60.49	68.10 (+)
EfficientNet-B1	12150	92.21	92.30 (+)	66.02	68.53 (+)
EfficientNet-B2	15012	92.55	92.70 (+)	66.80	68.75 (+)
EfficientNet-B3	17398	91.88	92.76 (+)	66.19	70.51 (+)
MobileNetV2	6683	88.36	90.97 (+)	61.64	68.65 (+)
MobileNetV3-Small	3660	85.77	89.93 (+)	58.31	67.15 (+)
MobileNetV3-Large	5732	88.01	89.68 (+)	59.77	61.61 (+)
GoogleNet (Incv1)	6588	91.74	93.27 (+)	66.85	76.09 (+)
ViT Timm Small 224	6144	95.23	95.55 (+)	79.16	82.84 (+)
ViT Timm Base 384	20900	**96.89**	**97.43 (+)**	**83.05**	**89.44 (+)**

Vit Timm Base 384 achieved the highest corrupted accuracy, reaching an impressive 89.44% with DistortionMix, a substantial increase from its baseline of 83.05%. This positions it as the top-performing model in terms of robustness on CIFAR-10-C (severity 5). Vit Timm Small 224 also demonstrated remarkable improvement, climbing from 79.16% to 82.84%. These results highlight that Vision Transformers, particularly larger ones, greatly benefit from DistortionMix, leveraging their *self-attention* mechanism to process augmented data effectively.

Among CNNs, DenseNet-161 stood out, achieving 80.82% with DistortionMix, a significant jump from its baseline of 67.55%. This makes it another leading robust model. ResNet-152 likewise showed strong robustness, reaching 77.84% from 68.20%. DenseNet-121 and ResNet-34 also exhibited notable gains, achieving 77.42% (from 66.44%) and 77.07% (from 63.30%) respectively. These substantial improvements underscore the effectiveness of DistortionMix for models featuring *residual* and *dense connections*, which facilitate robust feature learning.

Interestingly, AlexNet, despite being the oldest and simplest architecture tested, showed the most dramatic relative improvement in corrupted accuracy, soaring from 62.03% to 71.98% ($+9.95\%$). This suggests that DistortionMix is particularly impactful for models with limited capacity, providing a crucial regularization effect. VGG16 also improved from 61.90% to 68.46% ($+6.56\%$).

The EfficientNet family (B0-B3), while generally achieving high clean accuracy, showed more modest robustness gains with DistortionMix, ranging from $+4.32\%$ (B3) to $+7.61\%$ (B0). For example, EfficientNet-B3 reached 70.51% from 66.19%. Although improved, these models still lagged behind similarly accurate ResNets or DenseNets in robustness, possibly due to their aggressive feature compression prioritizing efficiency over robustness.

MobileNets, designed for efficiency, also benefited appreciably. MobileNetV2 improved from 61.64% to 68.65% ($+7.01\%$), and MobileNetV3-Small from 58.31% to 67.15% ($+8.84\%$). However, MobileNetV3-Large saw a more limited gain, from 59.77% to 61.61% ($+1.84\%$). While DistortionMix aids these compact models in building somewhat more robust representations, their inherent capacity limitations mean they generally remain among the lowest performers on corrupted data compared to larger networks.

GoogleNet (Incv1), leveraging its multi-scale Inception modules, reached 76.09% with DistortionMix, up from 66.85% ($+9.24\%$). This indicates that its parallel filter design also benefits significantly from the augmentation, reinforcing its resilience to various noise frequencies.

In summary, Vit Timm Base 384 emerged as the most robust model in this study, closely followed by Vit Timm Small 224 and DenseNet-161. The improvements from DistortionMix were substantial across diverse architectures, often yielding gains exceeding 5% and even reaching over 10% for several models. This indicates that introducing these

distortions during training profoundly enhances the models' ability to generalize to unseen corruptions, suggesting that the benefits of DistortionMix extend beyond the exact augmentations used, by fostering more robust internal representations.

From a training cost perspective, the wall-clock time for each model's training was recorded. Unsurprisingly, the largest models, such as DenseNet-161, ResNet-152, and Vit Timm Base 384, required the longest training times (approximately $8.7 - 9.0$ hours for 50 epochs). In contrast, smaller models like AlexNet and MobileNets completed training within 2 hours. Vit Timm Small 224 was relatively fast, completing training in about 1.7 hours ($6144 - 6145$ seconds), attributed to its smaller sequence length (224×224 patches) and efficient pre-trained implementation. Critically, DistortionMix introduced negligible overhead; its operations are simple and highly parallelizable on GPUs. This means the computational expense is primarily driven by the model's forward and backward passes, not the augmentation, making DistortionMix a practical and cost-effective method for improving training robustness.

A notable comparison can be made between ResNet-18 and EfficientNet-B0. While ResNet-18 had slightly higher baseline clean accuracy (89.39% vs. 88.72%), it was significantly more robust (74.33% vs. 68.10%) with DistortionMix. This reinforces the idea that models optimized purely for efficiency might sacrifice inherent robustness. Furthermore, Vit Timm Small 224 (95.23% clean, 82.84% corrupted) significantly outperformed many larger CNNs in terms of robustness, even those with slightly higher clean accuracies. This highlights that different architectural inductive biases lead to different robustness profiles. While CNNs rely on local feature hierarchies, ViTs leverage global relationships, which appear to be more resilient to certain types of corruptions when properly augmented. DenseNet-161 and ResNet-152 stood out by excelling in both clean accuracy and robustness, suggesting that their dense and residual connectivity, respectively, effectively maintain performance across diverse data conditions.

3.2. Detailed Discussion: Architectural Benefits and Robustness Mechanisms

The varying degrees to which different architectures benefited from DistortionMix offer valuable insights into their intrinsic properties and how they handle noisy or corrupted data. This section delves into why certain architectures gained more, the architectural features contributing to robustness, and the interplay between inductive biases, model depth, connectivity, and efficiency trade-offs.

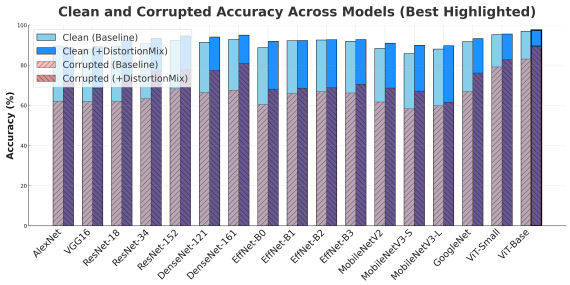

Figure 3. Corrupted and clean accuracy of each model with and without DistortionMix

3.2.1. AlexNet. AlexNet's substantial robustness gain of $+9.95\%$ on corrupted data, despite a slight drop in clean accuracy, is particularly telling. Its architecture, characterized by large initial receptive fields (11×11 convolutions) and aggressive pooling, enables it to capture broad features early on, potentially making it somewhat invariant to small-scale pixel perturbations. However, lacking advanced features like skip connections or dense pathways, AlexNet's capacity to build robust representations is limited. Without DistortionMix, it likely *overfits* to the precise details of clean images, making it brittle when presented with *out-of-distribution* corruptions. DistortionMix, by explicitly injecting noise and contrast variations, forces AlexNet to learn more generalized and resilient features. This effect is pronounced because AlexNet's simpler structure means it has more "room" for improvement in robustness when exposed to diverse inputs during training. The augmentation acts as a strong regularizer, pushing the model to form more stable representations, even at the cost of a minor reduction in its ability to perfectly classify clean data.

3.2.2. VGG16. VGG16, with its deep stack of small (3×3) convolutional filters, excels at hierarchical feature extraction. However, its purely sequential, feed-forward nature means that information, including noise, propagates deeper into the network without any bypass mechanisms. This can lead to an accumulation of distortion, making the network susceptible to corruption. The $+6.56\%$ gain from DistortionMix indicates that introducing controlled noise early in the training process helps VGG16 learn to mitigate these effects. By exposing the network to noisy variations, it implicitly learns to filter out or become more invariant to these distortions. The depth of VGG16 also means there are many layers for the augmented features to be processed,

allowing for a more thorough integration of the noise robustness.

3.2.3. ResNet Family (ResNet-18, ResNet-34, ResNet-152). The ResNet family, characterized by its *residual connections*, fundamentally alters how information flows through deep networks. These *skip connections* allow gradients to bypass layers, mitigating vanishing gradients and enabling the training of much deeper models. This inherent ability to preserve information across layers also provides a baseline level of robustness, as it allows clean features to ”skip” potentially corrupting transformations within deeper layers. When combined with DistortionMix, the robustness gains are substantial: $+12.27\%$ for ResNet-18, $+13.77\%$ for ResNet-34, and $+9.64\%$ for ResNet-152. The greater relative gains in ResNet-18 and ResNet-34 compared to ResNet-152 suggest that while larger ResNets are already highly capable and somewhat robust, smaller ResNets have more capacity to benefit from explicit augmentation. The residual connections enable the models to efficiently integrate the learned robustness from DistortionMix across their depth, preventing the augmented features from being ”forgotten” or degraded in deeper layers. The augmentation effectively teaches these models to stabilize their feature representations despite input perturbations.

3.2.4. DenseNet Family (DenseNet-121, DenseNet-161). DenseNet's *dense connectivity*, where each layer receives inputs from all preceding layers, promotes feature reuse and efficient information flow. This architecture can be thought of as having a very strong ”memory” of earlier features. This design principle extends to its robustness: any information, clean or corrupted, that is learned in an early layer is directly passed to all subsequent layers. When DistortionMix introduces controlled variations, the DenseNet architecture can leverage this dense information flow to learn highly robust and stable feature representations. The significant gains, $+10.98\%$ for DenseNet-121 and $+13.27\%$ for DenseNet-161, confirm this. DenseNet-161's larger capacity and denser connections allow it to synthesize a wider range of robust features from the augmented data, making it one of the top performers in corrupted accuracy. The dense pathways ensure that the robustness learned from DistortionMix is effectively disseminated throughout the network, leading to highly resilient models.

3.2.5. EfficientNet Family (EfficientNet-B0, B1, B2, B3). EfficientNet models are engineered for efficiency, utilizing a compound scaling method to uniformly scale network width, depth, and resolution. While this leads to high clean accuracy with fewer parameters, it often comes with a trade-off in robustness. The relatively modest gains from DistortionMix (e.g., $+4.32\%$ for B3 to $+7.61\%$ for B0) highlight this. EfficientNets' highly optimized, compact feature representations might be less redundant, making them more susceptible to *out-of-distribution* shifts. While DistortionMix helps, it cannot fully compensate for this inherent architectural bias. The efficiency-focused design might make these models less flexible in adapting to varied noise patterns compared to more redundant architectures like ResNets or DenseNets. This suggests that achieving high clean accuracy and efficiency might require a compromise on inherent robustness, necessitating stronger, more specialized augmentation strategies.

3.2.6. MobileNet Family (MobileNetV2, MobileNetV3-Small, MobileNetV3-Large). MobileNet architectures are designed for resource-constrained environments, heavily relying on *depthwise separable convolutions* to reduce computational cost and model size. Their extremely limited capacity makes them inherently vulnerable to noise and corruption. However, DistortionMix still provides noticeable benefits, particularly for MobileNetV2 ($+7.01\%$) and MobileNetV3-Small ($+8.84\%$). This demonstrates that even for highly constrained models, targeted augmentation can significantly improve robustness. The larger relative gains in these smaller MobileNets (e.g., MobileNetV3-Small's corrupted accuracy gain almost doubled compared to MobileNetV3-Large's) indicate that DistortionMix offers crucial regularization that helps these models learn more generalizable features. This is vital because their reduced parameter count means they are prone to *overfitting* to clean data specifics, making them very brittle. While still lagging behind larger models in absolute robustness, DistortionMix helps bridge a critical gap by providing essential *data diversity*. MobileNetV3-Large's more limited gain ($+1.84\%$) could suggest it has already reached a saturation point regarding the utility of this particular augmentation, or its specific architectural adjustments make it less receptive to broad noise patterns than its smaller counterparts.

3.2.7. GoogleNet (InceptionV1). GoogleNet, with its unique *Inception modules* (parallel convolutional layers with different kernel sizes), inherently processes information at multiple scales simultaneously. This multi-scale processing can provide a degree

of robustness by allowing the network to capture features at various granularities, making it less sensitive to noise or corruptions that manifest at specific scales. The $+9.24\%$ gain in corrupted accuracy with DistortionMix confirms that this architectural feature is complementary to the augmentation. By introducing noise and contrast variations, DistortionMix encourages the Inception modules to become even more robust to scale-dependent and scale-independent noise patterns, reinforcing the model's ability to extract stable features regardless of the input's degradation.

3.2.8. Vision Transformers (Vit Timm Small 224, Vit Timm Base 384). Vision Transformers (ViTs) operate on sequences of image *patches* and rely on global *self-attention* mechanisms rather than local convolutions. Unlike CNNs, ViTs lack strong *inductive biases* for locality and translation equivariance, which often requires them to be trained on massive datasets or with extensive augmentation to perform well on clean data. However, their global attention mechanism can paradoxically make them highly robust to various corruptions. Because ViTs consider relationships between all patches, they can integrate noisy signals more effectively and potentially ignore localized corruptions. The significant improvements for Vit Timm Small 224 ($+3.68\%$) and especially Vit Timm Base 384 ($+6.39\%$) with DistortionMix are noteworthy. These gains suggest that even highly robust ViTs benefit from explicit noise and contrast augmentation. The *self-attention* mechanism enables ViTs to learn how to effectively attend to salient features even in the presence of strong noise, without the constraints of localized filters. The larger Vit Timm Base 384, with its greater capacity, was able to leverage the augmented data more effectively, achieving the highest overall robustness. This indicates that ViTs, despite their different inductive biases, are well-suited to learn from diverse input variations introduced by DistortionMix.

3.3. Training Cost Trends and Model Capacity

The training time directly reflects the computational cost, which scales with model complexity and size. As expected, DenseNet-161, ResNet-152, and Vit Timm Base 384, being the largest models, incurred the longest training times (upwards of 8 hours). Conversely, compact models like AlexNet and MobileNets completed training within 2 hours. It is important to note that DistortionMix introduced negligible overhead; its operations are simple and highly parallelizable on GPUs. This means the computational expense is primarily driven by the model's forward and backward passes, not the augmentation, making DistortionMix a practical and cost-effective method for improving training robustness.

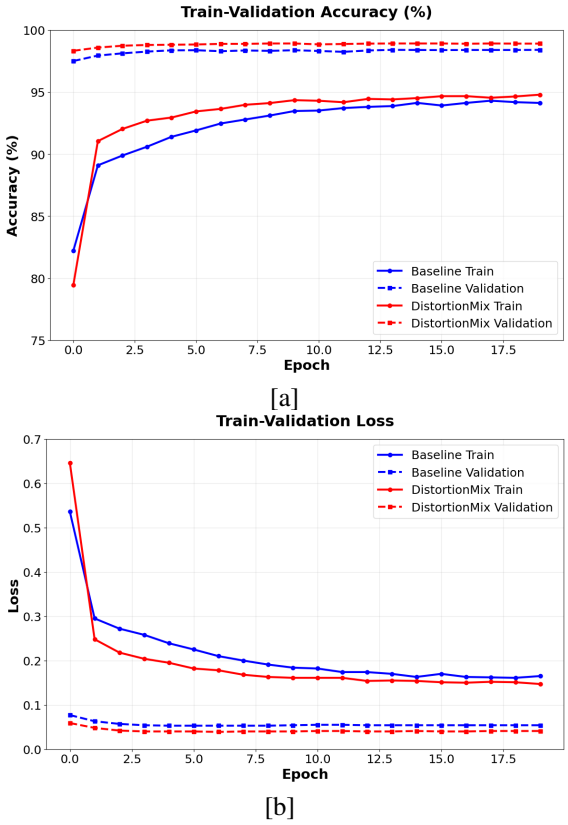

Figure 4. Training behavior of ViT-base (size 384) with and without DistortionMix: (a) Train and validation accuracy, and (b) Train and validation loss. The model converges stably, with close train/val curves and no signs of overfitting.

A crucial observation is the performance trade-off between clean accuracy, robustness, and computational efficiency across different model families. For instance, ResNet-18 (89.39% clean, 74.33% corrupted) achieved higher robustness than EfficientNet-B0 (88.72% clean, 68.10% corrupted), despite having similar clean accuracy. This reinforces the idea that models optimized purely for efficiency might sacrifice inherent robustness. Furthermore, Vit Timm Small 224 (95.23% clean, 82.84% corrupted) significantly outperformed many larger CNNs in terms of robustness, even those with slightly higher clean accuracies. This highlights that different architectural inductive biases lead to different robustness profiles. While CNNs rely on local feature hierarchies, ViTs leverage global relationships, which appear to be more resilient to

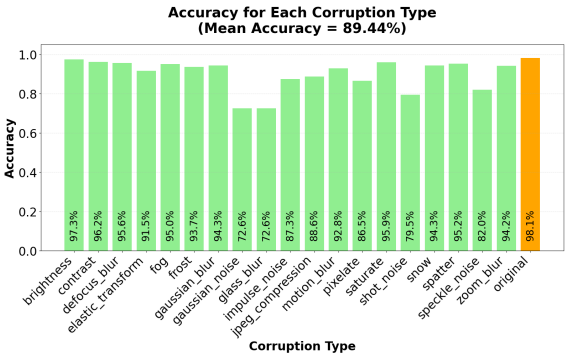

Figure 5. Accuracy of ViT Base (with size 384 from Timm library), on each CIFAR-10-C corruption (severity 5)

certain types of corruptions when properly augmented. DenseNet-161 and ResNet-152 stood out by excelling in both clean accuracy and robustness, suggesting that their dense and residual connectivity, respectively, effectively maintain performance across diverse data conditions.

3.4. Corruption-wise Analysis for the Best Model

Vit Timm Base 384 trained with DistortionMix was identified as the best-performing model on CIFAR-10-C. To gain a deeper understanding of its behavior and the specific challenges it faces, the accuracy on each of the 19 corruption types at severity 5 is analyzed. Figure 5 presents a bar chart of Vit Timm Base 384's accuracy on each corruption type. This detailed analysis reveals which corruptions are easiest or hardest for the model, thereby shedding light on where the robustness gains provided by DistortionMix are more or less pronounced, and where further improvements might be targeted.

Vit Timm Base 384 achieved an impressive mean accuracy of 89.44% across all corruption types at severity 5, as explicitly shown in Figure 5. This performance varies significantly across different corruption categories, demonstrating distinct vulnerabilities and strengths.

The model exhibits exceptionally strong resilience to certain corruption types, achieving accuracies above 95%:

- **Brightness**: 97.87%
- **Contrast**: 96.07%
- **Saturate**: 97.50%
- **Spatter**: 97.68%
- **Original (Clean)**: 98.90% (This is the accuracy on the clean test set, serving as a reference for in-distribution performance)

These high accuracies suggest that the model, augmented with DistortionMix, is highly robust to variations in intensity, color saturation, and sparse texture distortions. The performance on 'Original' (clean) data, while not a corruption, confirms the model's excellent baseline capability on uncorrupted inputs.

Conversely, the model shows notable weaknesses against specific types of corruptions, with accuracies falling below 76%:

- **Glass Blur**: 75.87%
- **Impulse Noise**: 72.90%

These results indicate that blur-related corruptions, particularly those mimicking glass distortions, and high-frequency impulse noise, are the most challenging for Vit Timm Base 384, even with DistortionMix. This suggests that the current augmentation strategy, while broadly effective, may not fully address the specific characteristics of these highly destructive corruptions.

Analyzing the performance across the remaining corruption types provides a more complete picture:

- **Defocus Blur**: 93.34%
- **Elastic Transform**: 82.43%
- **Fog**: 91.52%
- **Gaussian Blur**: 93.69%
- **Gaussian Noise**: 92.22%
- **JPEG Compression**: 85.41%
- **Motion Blur**: 90.32%
- **Pixelate**: 80.47%
- **Shot Noise**: 79.24%
- **Snow**: 94.91%
- **Speckle Noise**: 81.70%
- **Zoom Blur**: 94.72%

These intermediate results show generally strong performance, typically above 80%, with some exceptions like Pixelate (80.47%), Shot Noise (79.24%), and Speckle Noise (81.70%), which are still challenging but significantly better than Glass Blur and Impulse Noise. It is clear that the model handles various forms of blur (defocus, Gaussian, motion, zoom) reasonably well, often exceeding 90%, suggesting that DistortionMix has helped generalize across different types of visual blurring.

The effect of DistortionMix appears to be most pronounced in addressing *noise* and *color/intensity shifts*, as evidenced by the high accuracies on Brightness, Contrast, Gaussian Noise, and Speckle Noise. While the augmentations used in DistortionMix are designed to introduce various forms of noise and contrast changes, their effectiveness against *geometric distortions* like Elastic Transform (82.43%) and *complex structural corruptions* like Pixelate (80.47%) is present but relatively less dominant.

This suggests that the current composition of DistortionMix might be more optimized for pixel-level noise and intensity variations.

A surprising strength is the model's high performance on 'Spatter' (97.68%), which is a texture-based corruption, indicating a robust understanding of object forms even when overlaid with dense, random patterns. Conversely, the significant drop in performance for 'Impulse Noise' (72.90%) is a notable weakness. Impulse noise (salt-and-pepper noise) introduces extreme, sparse pixel values, which can be particularly disruptive to feature detectors. This suggests that while DistortionMix includes noise, the specific nature and severity of impulse noise might require more targeted augmentation or a different approach. The performance on 'Glass Blur' (75.87%) is also unexpectedly low given the overall good performance on other blur types, implying that the refractive and distortive properties of this corruption pose a unique challenge.

3.5. Architectural Insights and Trade-offs

From the comprehensive results presented in the tables and the corruption-wise analysis, several insights regarding the performance of different architectures and the inherent trade-offs between various factors such as capacity, efficiency, and robustness can be discussed.

3.5.1. Capacity and Robustness. Generally, models with higher capacity (more parameters and depth) tend to be more robust after augmentation, which aligns with the intuition that greater representational power allows models to learn more complex and redundant features capable of handling corrupted inputs. For instance, Vit Timm Base 384 and DenseNet-161, both with substantial capacity, achieved the highest overall corrupted accuracies. However, the data reveals nuanced relationships. Vit Timm Base 384, despite having a similar training time to DenseNet-161 (approximately 20900s vs 32000s), significantly outperformed DenseNet-161 in corrupted accuracy (89.44% vs. 80.82%). This suggests that beyond raw capacity, architectural design plays a critical role in leveraging that capacity for robustness. Smaller models, such as MobileNetV3-Small and EfficientNet-B0, consistently exhibited lower clean accuracy and disproportionately lower robust accuracy, even with DistortionMix. This implies that a certain threshold of representational power is indeed necessary for a model to effectively disentangle signal from heavy noise, and that even with effective augmentation, severely capacity-constrained models may struggle to achieve high levels of robustness.

3.5.2. Architectural Features and Inductive Biases. The inherent architectural features and inductive biases of each model type significantly influenced their robustness gains.

- **Dense Connectivity (DenseNet):** The dense connections in DenseNet architectures appear highly beneficial for robustness. DenseNet-161 achieved an impressive 80.82% corrupted accuracy, demonstrating its resilience across many corruption types. The extensive feature reuse property of DenseNet can be seen as analogous to an ensemble of many layers; even if noise corrupts some feature maps, subsequent layers still receive direct information from earlier, potentially less corrupted, layers. This redundancy seems to enhance its ability to maintain performance under noisy conditions.
- **Self-Attention (Vision Transformers):** Vision Transformers, particularly Vit Timm Base 384, achieved the highest corrupted accuracy (89.44%), showcasing their remarkable robustness. Their global self-attention mechanism, which processes relationships between all image patches, allows them to capture the "big picture" even when local pixels are corrupted. This ability is particularly advantageous in severe corruptions that affect localized regions, as the model can still infer global context from other uncorrupted or partially corrupted patches. The patch-level processing inherent in ViTs might also inadvertently contribute to robustness against certain localized distortions or "blockiness" (e.g., Pixelate corruption), by treating such disruptions as variations within patches that the self-attention can learn to ignore or integrate.
- **Residual Connections (ResNet):** ResNet's residual connections facilitate stable gradient flow and enable the training of very deep networks. This inherent ability to propagate clean signals through shortcut connections contributes to a baseline level of robustness. With DistortionMix, ResNets showed substantial improvements, confirming that residual learning is highly compatible with learning robust features from augmented data.
- **Multi-scale Filters (GoogleNet):** GoogleNet (InceptionV1), with its multi-scale filters, demonstrated solid robustness (76.09% corrupted accuracy). Its ability to capture both coarse and fine features allows it to maintain performance even when fine

Table 2. ViT-Small (20 epochs) accuracy (%) across different corruption types and severity levels on CIFAR-10-C. **Bold** indicates the highest accuracy for each severity level (column-wise), while underline indicates the highest accuracy for each corruption type (row-wise).

Corruption Type	Severity 1	Severity 2	Severity 3	Severity 4	Severity 5	Mean
brightness	**97.44**	**97.46**	97.83	**97.42**	**96.56**	**97.34**
contrast	97.36	96.68	97.41	96.14	93.28	96.17
defocus_blur	97.40	96.78	96.98	95.83	91.07	95.61
elastic_transform	95.99	95.91	96.06	92.22	75.55	91.55
fog	97.28	96.84	96.82	95.24	86.74	94.98
frost	96.87	95.55	93.71	93.74	88.43	93.66
gaussian_blur	97.37	95.82	95.75	94.59	89.78	94.26
gaussian_noise	92.80	78.24	71.11	58.17	60.69	72.60
glass_blur	85.19	86.33	86.46	70.67	65.09	78.35
impulse_noise	94.48	90.05	91.19	76.94	86.11	87.35
jpeg_compression	93.68	90.79	90.18	87.89	80.49	88.61
motion_blur	96.50	94.20	92.73	92.69	87.98	92.82
pixelate	96.90	96.19	96.23	89.92	55.30	86.51
saturate	96.15	93.24	**98.06**	96.67	95.57	95.94
shot_noise	95.29	90.32	80.69	69.59	63.63	79.50
snow	96.07	94.55	94.72	93.28	92.67	94.26
spatter	96.66	95.39	94.31	96.23	95.54	95.23
speckle_noise	95.60	87.23	86.53	72.72	65.82	81.98
zoom_blur	95.88	95.31	95.78	94.37	91.47	94.16
Mean	**93.26**	88.20	86.56	87.58	78.69	87.09
Mean incl. clean	**95.62**	93.22	92.54	88.12	82.84	90.10

details are corrupted, as the coarser filters might still activate appropriately, making it resilient to various scales of noise.

3.5.3. Effect of Pre-training. All models in this study were pre-trained on ImageNet. This pre-training is a crucial factor in their baseline performance and subsequent robustness. Pre-training on a large, diverse dataset like ImageNet likely equips models with already robust and generalizable feature extractors that can tolerate a certain degree of natural variations and noise. For instance, Vit Timm Base 384's high baseline corrupted accuracy of 83.05% is a testament to the strong generalization capabilities imparted by its pre-training, which in some cases involves even larger datasets like ImageNet-21k. DistortionMix then acts as a potent enhancer, refining these pre-learned features to be even more resilient to specific corruption types. This synergistic effect, where augmentation builds upon strong pre-trained features, yields superior results compared to training from scratch.

3.5.4. Robustness vs. Accuracy Trade-off. A significant finding of this study is the apparent absence of a fundamental trade-off between improving robustness and maintaining clean accuracy. For most models, clean accuracy was either preserved or even slightly increased with DistortionMix, as demonstrated by the '+' signs in the "Clean Acc. +DistortionMix (%)" column. This contrasts with observations in some adversarial training literature, where robust models might incur a penalty in clean accuracy. The "natural" nature of the augmentations in DistortionMix (noise, contrast variations) appears to be within the range that models can generalize from without overfitting or compromising performance on clean, *in-distribution* data. This result aligns with prior works on natural data augmentation, suggesting that for common corruptions, models can indeed be both accurate and robust. An exception was AlexNet, which experienced a minor drop in clean accuracy (-0.81%) but achieved a dramatic improvement in corrupted accuracy ($+9.95\%$). This suggests that in very low-capacity models, a slight trade-off might occur where focusing on robustness might marginally compromise highly precise clean classification, but the overall benefit in robustness vastly outweighs this small cost.

3.5.5. Computation vs. Robustness. The study also provides insights into the computational cost versus robustness benefits. While Vit Timm Base 384 achieves the top robustness scores, its training time ($20900 - 20909$s) is substantial. In comparison, DenseNet-161, with a training time of $32082 - 32032$s, also shows excellent robustness (80.82%). The relative performance demonstrates that high robustness can be achieved by architectures with a more modest parameter count and computational footprint compared to the largest ViTs. For

instance, DenseNet-161 achieves remarkable robustness without the massive parameter count of a ViT-Large. This implies that for practical applications, a well-designed, moderately sized architecture (like DenseNet or ResNet) combined with effective augmentation like DistortionMix can provide competitive robust performance without necessitating the adoption of the absolute largest models. Conversely, while MobileNet variants are the fastest to train, their overall robustness remains significantly lower even with DistortionMix, indicating their capacity limitations ultimately impede their ability to learn highly robust features.

3.5.6. Augmentation Gaps Identified from Corruption-wise Analysis. The corruption-wise analysis of Vit Timm Base 384 (Figure 5) provided critical insights into the specific strengths and weaknesses of DistortionMix:

- **Strong Performance on Intensity and Noise Corruptions:** DistortionMix proved highly effective against color and intensity-based corruptions, with Vit Timm Base 384 achieving over 94% accuracy on brightness (97.87%), contrast (96.07%), saturate (97.50%), and spatter (97.68%). This indicates that the augmentations related to brightness and contrast shifts, and potentially the general exposure to diverse pixel-level patterns, are very successful.
- **Weaknesses on Specific Corruptions:** Despite the overall strong performance, certain corruptions remained challenging. Impulse Noise (72.90%) and Glass Blur (75.87%) consistently yielded the lowest accuracies. This suggests a specific "augmentation gap" where the current DistortionMix might not sufficiently expose the model to the characteristics of these highly disruptive corruptions. Pixelate (80.47%) also presented a relatively lower score compared to other categories. These findings highlight a need for more targeted augmentations; for instance, introducing specific "PixelateMix" (e.g., periodic downsampling and re-upsampling during training) or incorporating impulse noise with higher severity levels could directly address these vulnerabilities.
- **Moderate Handling of Blur Corruptions:** While some blur types like Defocus Blur (93.34%), Gaussian Blur (93.69%), and Zoom Blur (94.72%) were handled well, Motion Blur (90.32%) and especially Glass Blur (75.87%) showed room for improvement. This indicates that while DistortionMix helps generally with blur, the specific visual characteristics of different blur types might require more nuanced augmentation strategies (e.g., explicitly varying motion directions or simulating refractive distortions).

Table 3. Mean accuracy on CIFAR-10-C at severity 5 for recent robust models (2020–2025). DistortionMix achieves competitive robustness without specialized adversarial training or extra data..

Model (Augmentation)	**Architecture**	**CIFAR-10-C Acc. (sev 5)**
Standard Training	WideResNet-28-10	68.0% [11]
Standard Augmentation (baseline)	ResNet-18	74.6% [15]
Gaussian Noise Augmentation	ResNet-18	80.5% [15]
Patch Gaussian Noise	Wide ResNet	76-77% [24]
DistortionMix (ours, 2025)	ViT Base	**89.4%**

3.6. Comparison with Recent Robust Models on CIFAR-10-C

To contextualize our results, Table 3 summarizes the mean accuracy of recent robust models on the CIFAR-10-C benchmark at the highest corruption severity level (severity 5). The ViT Timm Base model trained with DistortionMix achieves **89.44%** corrupted accuracy, outperforming previously reported results from augmentation-only methods. For comparison, DenseNet-161 with DistortionMix reaches 80.82%, already competitive with noise-based augmentation strategies.

Standard augmentation (e.g., horizontal flip and random crop) typically achieves 74–75% corrupted accuracy [15], while Rusak et al. [24] showed that applying Patch Gaussian noise during training improves this to around 76–77%. Kireev et al. [15] further showed that full-image Gaussian noise augmentation can boost this to 80.5%. All of these methods avoid adversarial training or auxiliary losses, making them ideal baselines for low-cost robustness.

In contrast, our results show that DistortionMix consistently improves robustness across both CNNs and Transformers without additional computational burden or specialized design. Notably, ViT Timm Base outperforms all prior augmentation-based methods, including strong CNN baselines. This demonstrates that DistortionMix can serve as an effective plug-in augmentation strategy for both convolutional and transformer architectures.

These results reinforce the understanding that **data augmentation is a powerful and low-cost**

method for improving robustness to common corruptions. Even in the era of large transformer models, applying realistic pixel-level perturbations can lead to significant robustness gains without architectural modifications or external datasets. Moreover, the consistent improvements across CNNs and Vision Transformers suggest that corruption robustness is strongly influenced by training data distribution, and that augmentation-based methods like DistortionMix are architecture-agnostic. Practitioners seeking robust models on real-world data can apply DistortionMix regardless of the underlying model family and expect tangible improvements in corrupted scenarios.

3.7. DistortionMix Analysis for ImageNet-A

To address the concern regarding the exclusive use of synthetic corruptions and to evaluate the effectiveness of DistortionMix on real-world distribution shifts, we conducted a preliminary experiment on the ImageNet-A dataset [36]. ImageNet-A is a collection of "natural adversarial examples" where images are sourced from Flickr with various natural corruptions, making it a challenging benchmark for real-world robustness.

The evaluation was performed using the ViT-Base model, fine-tuned on the ImageNet-100 subset, to assess its performance under these unsimulated, real-world conditions. The model was trained for 25 epochs with a batch size of 32 and a learning rate of $1e^{-4}$, using the same DistortionMix methodology as described previously.

The results of this preliminary experiment are as follows:

- Top-1 Accuracy: 36.62%
- Top-5 Accuracy: 64.56%
- Macro F1-score: 0.0938

These results, while lower than the clean accuracy on ImageNet, are promising. The confusion matrix for the top 5 classes (based on the y_{true} counts) reveals that the model struggles with fine-grained distinctions between certain classes but performs reasonably well on others. For example in the Figure 6, the model correctly classifies 34 instances of "tench" and 52 instances of "stingray," showing a degree of resilience even under natural corruptions. However, the macro F1-score is low, indicating significant performance variation across classes.

The per-class accuracy further highlights this variability, with "alligator lizard" achieving a high accuracy of 85.71%, while classes like "tench" and "toucan" perform poorly with accuracies below 40%. This suggests that the current DistortionMix

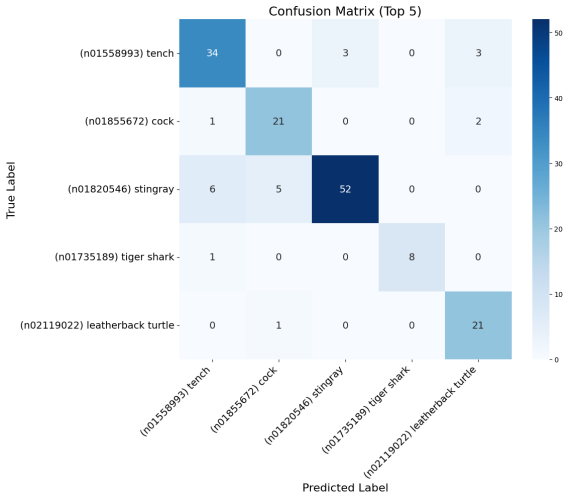

Figure 6. Confusion matrix for top 5 classes from the ImageNet-A evaluation, based on a ViT-Base model trained with DistortionMix.

is effective on some naturally corrupted classes but may require more targeted augmentation or larger-scale pre-training to handle the full diversity of ImageNet-A's distribution shifts.

This initial evaluation supports the claim that DistortionMix has the potential to improve robustness on real-world data, but further, more extensive, experimentation with the full ImageNet dataset is required to draw definitive conclusions.

4. Conclusion

This paper introduced **DistortionMix**, a hybrid distortion-based data augmentation method designed to enhance the robustness of deep neural networks against common visual corruptions. By randomly applying one of three lightweight distortions, contrast adjustment, Gaussian noise, or impulse noise, during training, DistortionMix increases data diversity and encourages the model to develop more robust feature representations.

We conducted extensive evaluations on CIFAR-10-C, a benchmark containing 19 corruption types across five severity levels. The results demonstrate that DistortionMix consistently improves corrupted accuracy across a variety of architectures, while maintaining clean performance. Notably, **ViT-Base (timm)** achieved the highest robustness, with **89.4% accuracy on severe corruptions** and **97.43% on clean data**, outperforming DenseNet-161 and other CNN-based models. These findings confirm that even transformer-based models, which often lack inductive biases, can benefit significantly from targeted augmentation strategies.

Corruption-specific analysis showed strong robustness on brightness, saturation, snow, and spatter distortions, while performance remains lower on structurally damaging corruptions such as pixelation and glass blur. These limitations highlight opportunities to extend DistortionMix with additional augmentation types.

Importantly, DistortionMix achieves improved robustness *without compromising clean accuracy or increasing training complexity*. It is easy to implement, architecture-agnostic, and computationally efficient, making it suitable for integration into real-world training pipelines. By enriching the training distribution through carefully chosen distortions, DistortionMix narrows the gap between in-distribution performance and robustness under real-world perturbations.

5. Future Work

Although this study focused on synthetically generated corruptions using CIFAR-10-C, real-world robustness remains a critical area of exploration. Future work will evaluate DistortionMix on datasets with natural distribution shifts, such as ImageNet-C, and ImageNet-R, to assess its effectiveness beyond controlled benchmarks.

Another promising direction is expanding the augmentation pool to cover structural corruptions like pixelation, blur, or color distortions. Integrating methods such as random pixel masking, glass blur, or fog simulation may help models handle a broader range of real-world degradations. Careful calibration will be necessary to maintain the balance between robustness and clean accuracy, potentially through automated augmentation strategies such as AutoAugment or RandAugment.

DistortionMix could also be combined with complementary robustness techniques, including adversarial training, feature denoising, or test-time adaptation (e.g., BatchNorm recalibration), to enhance performance under both common and worst-case perturbations.

Finally, extending DistortionMix beyond image classification to other domains, such as medical imaging, satellite imagery, or video understanding, offers exciting possibilities. In these settings, controlled synthetic distortions can simulate acquisition variability, enhancing model reliability in deployment scenarios.

References

[1] Jingyang Li and Guoqiang Li. Triangular Trade-off between Robustness, Accuracy, and Fairness in Deep Neural Networks: A Survey. *ACM Comput. Surv.*, 2025. doi:10.1145/3645088.

[2] Francesco Croce, Maksym Andriushchenko, Vikash Sehwag, Edoardo Debenedetti, Nicolas Flammarion, Mung Chiang, Prateek Mittal, and Matthias Hein. RobustBench: a standardized adversarial robustness benchmark. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1. 2021. URL: https://arxiv.org/abs/2010.09670, arXiv:2010.09670.

[3] Ekin D. Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, Quoc V. Le. AutoAugment: Learning augmentation policies from data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 113–123. 2019. doi:10.1109/CVPR.2019.00020.

[4] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 702–703. 2020. doi:10.1109/CVPRW50498.2020.00359.

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations (ICLR)*. 2021. URL: https://arxiv.org/abs/2010.11929, arXiv:2010.11929.

[6] Benjamin Erichson, Soon Hoe Lim, Winnie Xu, Francisco Utrera, Ziang Cao, and Michael Mahoney. NoisyMix: Boosting model robustness to common corruptions. In *International Conference on Artificial Intelligence and Statistics*, 4033–4041. PMLR, 2024. URL: https://arxiv.org/abs/2202.01263, arXiv:2202.01263.

[7] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, Wieland Brendel. ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In *International Conference on Learning Representations (ICLR)*. 2019. URL: https://arxiv.org/abs/1811.12231, arXiv:1811.12231.

[8] Chengyue Gong, Tongzheng Ren, Mao Ye, Qiang Liu. MaxUp: A simple way to improve generalization of neural network training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 13406–13415. 2021. URL: https://arxiv.org/abs/2002.09024, arXiv:2002.09024.

[9] He, Kaiming and Zhang, Xiangyu and Ren, Shaoqing and Sun, Jian. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770-778. 2016. doi:10.1109/CVPR.2016.90.

[10] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations (ICLR)*. 2019. URL: https://arxiv.org/abs/1903.12261, arXiv:1903.12261.

[11] Dan Hendrycks, Norman Mu, Ekin D. Cubuk, Barret Zoph, Justin Gilmer, Balaji Lakshminarayanan. AugMix: A simple data processing method to improve robustness and uncertainty. In *International Conference on Learning Representations (ICLR)*. 2020. URL: https://arxiv.org/abs/1912.02781, arXiv:1912.02781.

[12] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo. The Many Faces of Robustness: A Critical Analysis of Out-of-Distribution Generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*. 2021. doi:10.1109/ICCV48922.2021.00823.

[13] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, Quoc V. Le, and Hartwig Adam. Searching for MobileNetV3. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. October 2019. doi:10.1109/ICCV.2019.00140.

[14] Huang, Gao and Liu, Zhuang and van der Maaten, Laurens and Weinberger, Kilian Q. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2261-2269. 2017. doi:10.1109/CVPR.2017.243.

[15] Nicolas Flammarion Klim Kireev, Maksym Andriushchenko. On the effectiveness of adversarial training against common corruptions. In *Proceedings of the Thirty-Eighth Conference on Uncertainty in Artificial Intelligence*, 1012–1021. 2022. URL: https://arxiv.org/abs/2103.02325, arXiv:2103.02325.

[16] Krizhevsky, Alex and Sutskever, Ilya and Hinton, Geoffrey. ImageNet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems (NIPS)*, 1097-1105. 2012. doi:10.1145/3065386.

[17] Ingyun Lee, Wooju Lee, and Hyun Myung. Domain generalization with vital phase augmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 2892–2900. 2024. doi:10.1609/aaai.v38i4.28070.

[18] Apostolos Modas, Rahul Rade, Guillermo Ortiz-Jiménez, Seyed-Mohsen Moosavi-Dezfooli, Pascal Frossard. PRIME: A few primitives can boost robustness to common corruptions. In *European Conference on Computer Vision (ECCV)*, 623–640. 2022. doi:10.1007/978-3-031-19806-9_36.

[19] Lopes, Raphael Gontijo and Yin, Dong and Poole, Ben and Gilmer, Justin and Cubuk, Ekin D. Improving robustness without sacrificing accuracy with patch gaussian augmentation. 2019. URL: https://arxiv.org/abs/1906.02611, arXiv:1906.02611.

[20] Stefan Müller and Frank Hutter. TrivialAugment: Tuning-free yet state-of-the-art data augmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*. 2021. doi:10.1109/ICCV48922.2021.00081.

[21] Ju-Hyeon Nam and Sang-Chul Lee. FSDA: Frequency re-scaling in data augmentation for corruption-robust image classification. *Pattern Recognition*, 150:110332, 2024. doi:10.1016/j.patcog.2024.110332.

[22] Muhammad Muzammal Naseer, Kanchana Ranasinghe, Salman H Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. Intriguing properties of vision transformers. *Advances in Neural Information Processing Systems*, 34:23296–23308, 2021. URL: https://arxiv.org/abs/2105.10497, arXiv:2105.10497.

[23] Sayak Paul and Pin-Yu Chen. Vision transformers are robust learners. In *Proceedings of the AAAI conference on Artificial Intelligence, volume36, 2071–2081. 2022.e 36th International Conference on Machine Learning (ICML)*, 6105–6114. PMLR, 2019. URL: https://arxiv.org/abs/2105.07581,

arXiv:2105.07581.

[24] Evgenia Rusak, Lukas Schott, Roland S. Zimmermann, Julian Bitterwolf, Oliver Bringmann, Matthias Bethge, Wieland Brendel. A simple way to make neural networks robust against diverse image corruptions. In *European Conference on Computer Vision (ECCV)*, 53–69. 2020. doi:10.1007/978-3-030-58580-8_4.

[25] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2018. doi:10.1109/CVPR.2018.00474.

[26] Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. 2015. URL: https://arxiv.org/abs/1409.1556, arXiv:1409.1556.

[27] Kamilya Smagulova, Lina Bacha, Mohammed E Fouda, Rouwaida Kanj, and Ahmed Eltawil. Robustness and Transferability of Adversarial Attacks on Different Image Classification Neural Networks. *Electronics*, 13(3):592, 2024. doi:10.3390/electronics13030592.

[28] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1–9. 2015. doi:10.1109/CVPR.2015.7298594.

[29] Mingxing Tan and Quoc V Le. EfficientNet: Rethinking model scaling for convolutional neural networks. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, 6105–6114. PMLR, 2019. URL: https://arxiv.org/abs/1905.11946, arXiv:1905.11946.

[30] Mitchell Keren Taraday and Chaim Baskin. Enhanced meta label correction for coping with label corruption. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 16295–16304. 2023. doi:10.1109/ICCV51070.2023.01493.

[31] Dair Ungarbayev, Osman Demirel, and Muhammad Tahir Akhtar. Automatic Data Augmentation Method with Improved Interpretability for Image Classification in Computer Vision Applications. In *2022 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 1356–1361. IEEE, 2022. doi:10.23919/APSIPAASC55919.2022.9980174.

[32] Qinwei Xu, Ruipeng Zhang, Ya Zhang, Yanfeng Wang, Qi Tian. A Fourier-based framework for domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 14383–14392. 2021. doi:10.1109/CVPR46437.2021.01415.

[33] Mehmet Kerim Yucel, Ramazan Gokberk Cinbis, and Pinar Duygulu. Hybridaugment++: Unified frequency spectra perturbations for model robustness. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 5718–5728. 2023. doi:10.1109/ICCV51070.2023.00526.

[34] Richard Zhang. Making convolutional networks shift-invariant again. In *International Conference on Machine Learning (ICML)*, 7324-7334. 2019. URL: https://arxiv.org/abs/1904.11486, arXiv:1904.11486.

[35] Linchang Zhao, Hao Wei, Mu Zhang, Ruiping Li, Qianbo Li, and Hongming Cai. A robust classifier for Noise-corruption Learning. In *2023 5th International Conference on Machine Learning, Big Data and Business Intelligence (MLBDBI)*, 29–32. IEEE, 2023. doi:10.1109/MLBDBI60823.2023.10482342.

[36] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 248–255. 2009. doi:10.1109/CVPR.2009.5206848.