.07848

DMB

DATABASE MANAGEMENT
AND
BIOMETRICS

LEARNING DATA AUGMENTATION POLICIES FOR COMPUTER VISION USING ADDITIVE FOURIER-BASIS NOISE

Yijing Zeng

MASTER'S ASSIGNMENT

Committee:
Nicola Strisciuglio
Faizan Ahmed
Shunxin Wang

February, 2023

Data Management and Biometrics
EEMathCS
University of Twente
P.O. Box 217
7500 AE Enschede
The Netherlands

UNIVERSITY OF TWENTE. | DIGITAL SOCIETY INSTITUTE

Learning Data augmentation policies for computer vision using additive Fourier-basis noise

Yijing Zeng
zengyijing123@gmail.com

February 25, 2023

Abstract

Data augmentation is an important tool to improve model robustness. This study uses Fourier-basis noise to augment images. A new approach is introduced that utilizes Reinforcement Learning to find useful combinations of noise as augmentation policies. The results demonstrate that the searched Fourier-basis augmentation is more effective in improving the model's robustness to corruption than the baseline model. Furthermore, combining different augmentation techniques further enhances the model's performance, indicating that Fourier-basis augmentation positively affects model robustness.

keywords: Data augmentation, Fourier-basis noise, Robustness, Reinforcement Learning

1 INTRODUCTION

Many efforts have been made to improve the image classification performance of deep learning models [1, 2]. However, these models still have poor performance when data distribution changes between the training set and testing set [3]. Common corruptions, such as noise, blur, weather, and digital distortions, may cause deep learning models to fail the classification. Using data augmentation can be a successful method to enhance the robustness of models against corruption, and it involves expanding the variability of the data [4]. In the past few years, data augmentation's rapid development has dramatically boosted its applications in many fields, such as medical imaging [5, 6, 7], agriculture [8, 9], satellite imagery [10, 11].

Most common image data augmentation mainly applies operations in the spatial domain, such as image transformations (e.g., flip, rotation, scale, crop, and translation) and colour modification (e.g., brightness, contrast, and grayscale) [4, 12, 13, 14]. Strategies that mix samples have been proven to be effective. For example, MixUp [15], proposed by Zhang et al., randomly selects two samples with their labels for random weighted summation. This method strengthens the generalization of models and performs well on object detection [16]. CutMix [17] in contrast to MixUp, which mixes the samples by interpolating two images proportionally, mixes the samples by cutting some areas and then patching them. So CutMix enables the model to identify two objects from a composite image, improving training efficiency.

There are some automatic data augmentation methods. AutoAugment [18] uses Reinforcement Learning [19, 20] to search for optimal augmentation policy and achieves significant performance on CIFAR-10-C and ImageNet-C benchmarks [21]. Different operation-related parameters allow the augmentation policy found by AutoAugment to match the target dataset well. Similar to AutoAugment, AugMix [22] applies a variety of data augmentation (Aug) to the image and combines (Mix) several data-augmented images. Based on AugMix, Soklaski et al. [23] propose AugSVF to enhance the model effectively. AugSVF [23] includes spatial, vision, and Fourier-basis perturbation in the AugMix framework, and can be customized to effective perturbation while improving the overall model robustness.

However, it has been observed that models have different performances facing various corruptions [21]. According to the analysis from a Fourier perspective of Yin et al. [24], corruptions can be broadly divided into two categories depending on the energy distribution: low-frequency corruptions and high-frequency corruptions. There is a trade-off that improved robustness to high-frequency corruptions always comes at the cost of decreasing robustness to low-frequencies corruptions, and the frequency information of these corruptions is a critical factor in explaining the trade-off.

Performance trade-off in robustness between low-frequency and high-frequency corruptions raises the question if there is a data augmentation method to enable models to be robust to different corruptions. This research proposes the hypothesis that adding mix-frequency Fourier-basis noise to images improves the robustness of models and mitigates the trade-off effectively. Adding mix-frequency noise can add useful information that related to the data, making the model more robust to corruptions focused on different frequencies. This research first analyzes the effect of applying different frequencies of Fourier-basis noise on improving models' robustness. Then, a combined Fourier-basis augmentation policy is searched by Reinforcement Learning. Finally, the effectiveness of searched policy is evaluated and compared with other augmentation methods(AutoAugment, AugMix, etc.).

Goal: Investigate a suitable augmentation strategy using the Fourier-basis noise and Reinforcement Learning method.
The goal is specified as the following research questions:

- **RQ 1**: What effect does using additive Fourier-basis noise as augmentation have on the robustness of convolutional networks to common computer vision corruptions?

- **RQ 2**: How to combine different frequencies of the Fourier-basis in a data augmentation stage to maximize network robustness?
  - **RQ 2.1**: How to find a suitable data augmentation strategy based on adding different Fourier-basis by Reinforcement Learning?
  - **RQ 2.2**: How do the augmentation strategy in **RQ 2.1** and other methods(AutoAugment, AugMix, etc.) compare in improving the robustness to common computer vision corruptions?

To answer **RQ 1**, we augment the data with different probability weighting of noises and compare performance on testing set with the baseline model. The answer to **RQ 1** reveals Which frequency band of Fourier-basis noise would be most helpful in improving the robustness of the model. Answering **RQ 2.1** obtains an augmentation policy, which is evaluated by several metrics. The following experiment provides results comparing the performance with other methods.

The paper is organized as follows: Section 2 covers relevant research on augmentation, including Additive Fourier-basis noise, spectral bias, and model robustness to common computer vision corruption. In Section 3, the methods adding Fourier-basis noise and searching for augmentation policy are explained. Section 4 introduces the experimental design. Results and discussion are presented in Sections 5 and 6. Finally, the main research questions are answered in the conclusion in Section 7.

2 RELATED WORK

2.1 Data Augmentation

Besides augmenting images by image transformation and colour modification, some augmentation methods increase data variability. A useful technique is Generative adversarial networks(GAN) [25]. GAN is a class of artificial intelligence algorithms for unsupervised machine learning. It aims to analyze training examples, discover the statistics of the training set, and create more data. Wang and Perez [26] prove GAN's effectiveness in improving the robustness of the model and propose combining different data augmentation methods. Adversarial training [27] defenses adversarial attacks effectively by retraining the model on purely adversarial examples or regenerated data of original and adversarial examples. Several recent papers [28, 29] improve Adversarial training, making this method more suitable for large-scale problems and faster.

AutoAugment[18] automatically searches for augmentation strategies that fit the dataset, achieving good performance in classification and improving the robustness of the model to corruptions. In the AutoAugment framework, a controller samples an augmentation policy from the search space, which contains image processing operations, and then uses it to train data. Image processing operations include ShearX/Y, TranslateX/Y, Rotate, AutoContrast, Invert, Equalize, Solarize, Posterize, Contrast, Color, Brightness, Sharpness, Cutout, and Sample Pairing. Validation accuracy is the reward for upgrading the controller to produce better policies. However, this method takes a lot of time. Based on AutoAugment, Lim et al. propose a faster approach to search optimal augmentation policies, named Fast AutoAugment (FAA) [30]. With the same search space as AutoAugment, FAA exploits optimal augmentation policies from transformation candidates by Bayesian optimization [31]. FAA explicitly looks for augmentation strategies that maximize the match between augmented and unaugmented distribution. The performance of FAA doesn't get a higher accuracy than AutoAugment, but FAA speeds up the search process significantly.

2.2 Additive Fourier-basis Noise

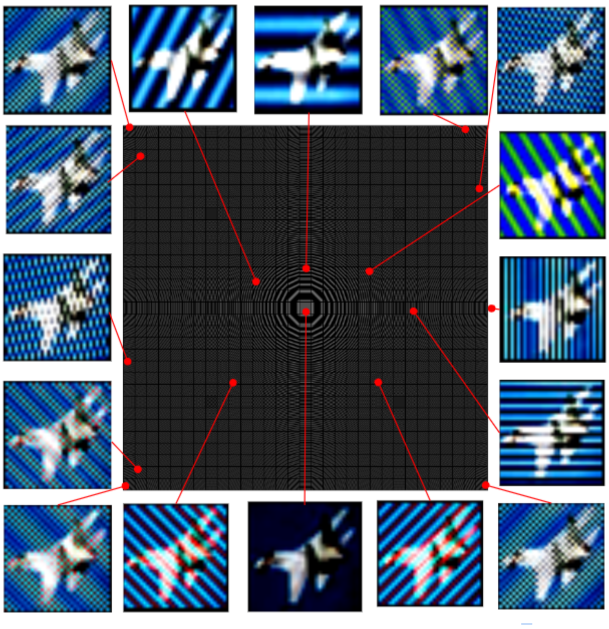
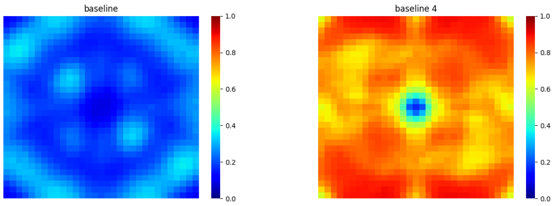


Figure 1: The effect of adding Fourier-basis noise with norm 4 at different frequencies and phases on clean image.

Yin et al. [24] introduce a generation process of perturbed images with Fourier-basis noise. Images are augmented by adding perturbations with

$$\tilde{X_{i,j}} = X + rv\mathcal{F}(U_{i,j})$$

where X denotes an original image, $\mathcal{F}(U_{i,j})$ indicates 2D Fourier-basis matrices [32], $U_{i,j}$ represents a matrix with only two non-zero elements located at (i, j) and its symmetrical coordinates relative to the center. The variable r is a random number between -1 and 1; the variable v is the norm of the noise, representing the degree of added noise. Hence, the testing set with Fourier-basis noise is a group of $\tilde{X_{i,j}}$. There is one Fourier-basis noise in every entry, with size 32×32, which can be added to images in CIFAR-10. The frequency of each noise is related to its radius to the central noise; the larger the radius, the higher the frequency. Hence, the noise in the centre has the lowest frequency, and the noise at the four corners has the highest frequency. Figure 1 shows a 31×31 type 2D Fourier-basis noise and examples of adding Fourier-basis noise to images. From a visual perspective, the noise creates stripes in the image, with the density of the stripes increasing as the frequency of the added noise increases.



(a) (b)

Figure 2: Baseline model sensitivity to additive noise aligned with different Fourier-basis vectors on CIFAR-10. The norm of Fourier-basis noise is 1 in heat map (a) and 4 in heat map (b). Dark blue indicates that the model is robust to attack, and dark red indicates that it is weak. (b) indicates that the model is powerful to low-frequency noise attacks and vulnerable to high-frequency noise attacks.

The Fourier heat map [24] reflects the sensitivity of the CNN to high-frequency and low-frequency damage. As shown in Figure 2, each entry (i, j) represents the average error rate of the model on the testing set with Fourier-basis noise of (i, j).

2.3 Spectral bias of neural networks

Yin et al. [24] demonstrate that models prefer utilizing low-frequency information of corruptions during the augmentation process. Guo [33] also explains it. He finds that limiting the search for adversarial images to the low-frequency domain offers significant advantages for attacks in a black-box environment. Based on the bias, Saikia et al. [34] propose RoHL. RoHL mixes two augmentation methods with good performance in the face of low-frequency and high-frequency corruptions, respectively, which is an effective model to avoid the trade-off mentioned in Section 1.

2.4 Robustness to common image corruptions

Hendrycks and Dietterich [21] introduce IMAGENET-C, which contains images with common visual corruptions. As shown in Figure 3, there are 19 types of corruptions: Gaussian Noise, Shot, Impulse, Defocus, Speckle, Gaussian Blur, Glass, Motion, Zoom, Snow, Frost, Fog, Bright, Contrast, Elastic, Pixel, JPEG, Spatter, and Saturate, which can be categorized into noise, blur, weather, and digital. Similarly, CIFAR-10-C [35] are created by combining CIFAR-10 images and those corruptions and perturbations.

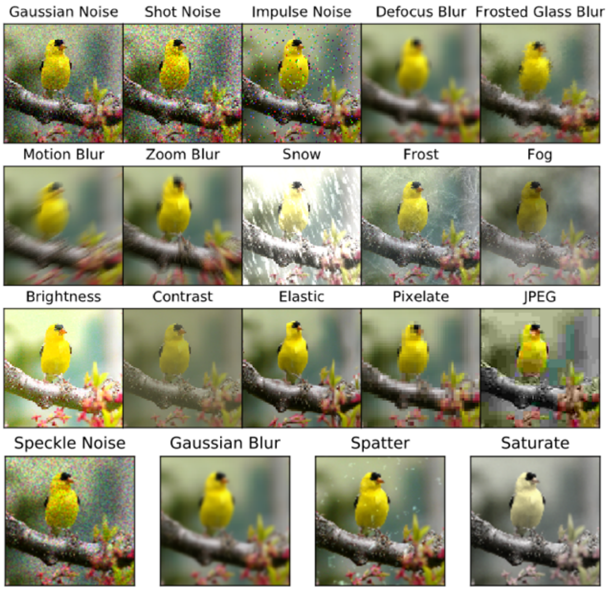


Figure 3: Examples of images with 19 types of corruption from noise, blur, weather, and digital categories.[21]

Yin et al. [24] compare the robustness of the naturally trained model (a model that has been trained on the CIFAR-10 dataset without any additional data augmentation), Gaussian data augmentation, adversarially trained model, and AutoAugment by testing them on CIFAR-10-C. As shown in Table 1, the AutoAugment model has the highest accuracy rate of 86% among the other three models and the lowest error rate on the CIFAR-10-C dataset.

3 METHODS

In this section, we begin by discussing the frequency information of Fourier-basis noise. And then adding Fourier-basis noise with different selection probabilities to data, to examine the effect of frequency addition on improving the model's robustness. We then describe how we use Reinforcement Learning based on FAA to search for augmentation policies.

3.1 Data augmentation with Fourier-basis noise

The noise is generated using the method described in Section 2.2. The noise can be divided into 22 groups based on the radius, with a radius ranging from 1 to 22, and the noise in each group having the same radius and frequency.

We design four models, each with a different preference for selecting frequency. To emphasize the impact of low, medium, and high frequencies, we assign a high probability to these three frequency ranges when we add noise to the model separately. The possibility of selecting each group of noise is determined by a normal distribution using the following functions, where x denotes the radius of the group. The followings are the four models and their corresponding probability functions:

- Uniform: The model adds noise where every frequency has a uniform probability.

$$P(x) = 0.5$$

- Low: The model has a higher probability of selecting high-frequency noise.

$$P(x) = \frac{1}{\sqrt{2\pi}} exp(-\frac{(x-1)^2}{8}) * 5$$

- Mid: The model has a higher probability of selecting mid-frequency noise.

$$P(x) = \frac{1}{\sqrt{2\pi}} exp(-\frac{(x-11.5)^2}{8}) * 5$$

- High: The model has a higher probability of selecting high-frequency noise.

$$P(x) = \frac{1}{\sqrt{2\pi}} exp(-\frac{(x-22)^2}{8}) * 5$$

As indicated in Figure 4, the Low model focuses on adding low-frequency noise, resulting in a much higher probability of selecting groups 1 to 5. The Mid model mainly adds noise in the middle-frequency range, and the probability of choosing low and high frequencies is relatively low. Contrary to the Low model, the High model mainly chooses adding high-frequency noise as its augmentation method.

model	acc	mCE	noise			blur					weather			digital			
			speckle	shot	impulse	defocus	Gauss	glass	motion	zoom	snow	fog	bright	contrast	elastic	pixel	jpeg
natural	77	100	70	68	54	85	73	57	81	80	85	90	95	82	86	73	80
Gauss	83	98	92	92	83	84	79	80	77	82	88	72	92	57	84	90	91
adversarial	81	108	82	83	69	84	82	80	80	83	83	73	87	77	82	85	85
AA	86	64	81	78	86	92	88	76	85	90	89	95	96	95	87	71	81

Table 1: Comparison between naturally trained model (natural), Gaussian data augmentation (Gauss), adversarially trained model (adversarial), and AutoAugment (AA) on CIFAR-10-C [24].

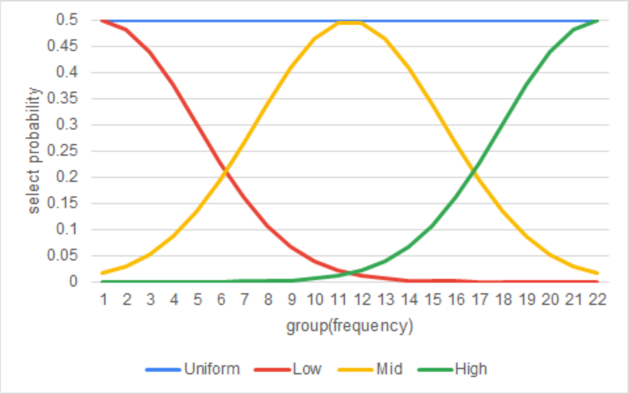


Figure 4: The distribution of the possibility of adding Fourier-basis noise in 22 different groups is displayed. The x-axis represents the frequency of the noise, and the y-axis shows the probability of adding noise at that specific frequency.

3.2 Search for augmentation policies via Reinforcement Learning

This subsection introduces the process of searching for optimal augmentation policies, including the choice of augmentation candidates and search strategy.

3.2.1 Search space

Unlike AutoAugment and FAA, which choose common image processing operations as augmentation candidates, our research chooses to add Fourier-basis noise with various frequencies. According to 3.1, there are 22 groups of Fourier-basis noise, resulting in 22 transformation candidates. The search space also contains probabilities of adding every noise frequency and the magnitudes of added noise. To be precise, the search space can be indicated by (o, p, m), where o is the transformation candidate, p is the probability of applying this transformation, and m represents the magnitude of this transformation. p has a continuous value range of 0 to 1, while m has a continuous value range of 0 to 5. Each augmentation sub-policy consists of two operations, and the final augmentation policy consists of eight sub-policies. As a result, the policy combines various noise groups, allowing the model to incorporate a diverse range of frequency information.

The initial design of the search space is based on categorizing noise by frequency. As a result, the search space includes 22 possible transformations. To increase the search space, the method of dividing noise by frequency and phase was employed. The phase represents the difference in the signal from a standard phase reference. By dividing the noise of the same frequency into four quadrants based on phase, with ranges from 0-90°, 90-180°, 180-270°, and 270-360°, the search space is expanded to include 85 transformation candidates. Furthermore, the phase is divided into 45° increments, and the number of candidates increase to 165.

3.2.2 Search strategy

We use FAA to search for the optimal augmentation strategy. Figure 5 shows the process. First, the training set D_{train} is divided into K subsets, each composed of two subsets D_A and D_M. The purpose of D_M is to train the model parameters. After the model parameters have been trained, for each step t where $1 \leq t \leq T$, the algorithm explores a set of B candidate policies, using a Bayesian optimization method [31]. This method involves repeatedly sampling a series of sub-policies from the search space S to create a policy P. The probability of applying each sub-policy in the policy and the magnitude are then adjusted to minimize the expected loss on the augmented set $P(D_{train})$.

The sampled sub-policies are used to calculate the model's loss as a reward, and the top N sub-policies with the lowest loss are chosen to form the augmentation policy. In this case, the top 8 sub-policies are selected from a 5-fold stratified shuffle on CIFAR-10, with 200 candidate policies evaluated (using T=2 and B=100). These parameters can be adjusted to improve performance and increase the number of sub-policies in the final aug-

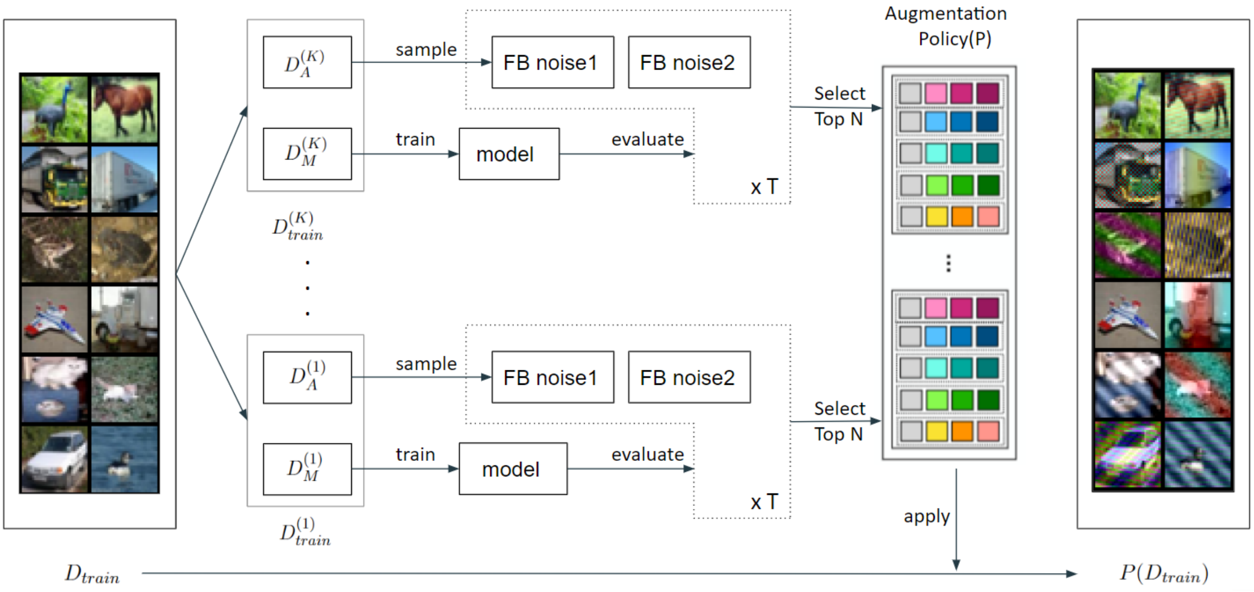


Figure 5: An overview of augmentation search process. The process of finding the optimal augmentation policy involves randomly selecting sub-policies from a pool of candidates. The sub-policies are evaluated for their effectiveness, and the top N performers are ultimately selected to make up the final augmentation policy.

mentation policy.

3.3 Evaluation metrics

To evaluate the results of experiments, we use several metrics to represent the performance of the augmentation strategies. First, this research uses accuracy to measure the performance of classification. Corruption error(CE) and mean corruption error(mCE) are used to compare the robustness between this model and the baseline model. Test the trained model on every corruption in CIFAR-10-C with severity levels from 1 to 5, and the error is denoted by $E^f_{s,c}$. Then the classification error is denoted by $E^{base}_{s,c}$. To compare the performance between the model f and the baseline model, we compute the CE by the following formula:

$$CE^f_c = \frac{\sum_{s=1}^{5} E^f_{s,c}}{\sum_{s=1}^{5} E^{base}_{s,c}}$$

To summarize the overall corruption robustness of model f, we computer mCE by averaging all corruption error values.

4 EXPERIMENT

In this section, we introduce two experiments that aim to study the robustness of a machine learning model under Fourier-basis augmentation. The first experiment focuses on the effect of adding different frequency noise on the model's performance, and the second experiment is designed to find the optimal data augmentation policy for the model and then evaluate its performance.

4.1 Experiment 1: Data augmentation with Fourier-basis noise

This experiment compares the performance of five models on CIFAR-10 and CIFAR-10-C. The baseline model is Wide Resnet-28-10 without augmentation. The other four models are baseline with augmentations: Uniform, High, Mid, and Low. These models are augmented by adding Fourier-basis noise with frequencies-bias probability distribution introduced in Section 3.1, and the noise magnitude is 4.

Augmentation setup: In addition to the Fourier-basis noise, other methods are used for the augmentation step. Before adding Fourier-basis noise selected according to the assigned probability, the transformation for all models consists of padding, random horizontal flipping, and random cropping.

Training: The procedure is implemented in PyTorch using the Wide Resnet-28-10 architecture. The training data of CIFAR-10 is divided into training and validation sets in the ratio of 90:10. The optimizer uses Adam, with a learning rate of

0.0001, a weight decay of 1e-4, and the loss function is cross-entropy loss. Each experiment consists of 100 epochs, and training is stopped early after 30 epochs of no improvement in validation loss.

Testing: After the training phase, the models are tested on clean images and images with corruptions. The results indicate the models' sensitivity to various types of corruption, and with the Fourier spectrum of these corruptions, we can analyze the result from a frequency perspective. The evaluation tools are introduced in 3.3.

Experiment 1 helps us to gain more insight into whether adding different frequency information can make the models more robust or bring the opposite result. And it also allows us to understand the trade-off that improved robustness to high-frequency corruptions always comes at the cost of decreasing robustness to low-frequency corruptions.

4.2 Experiment 2: Search for augmentation policies

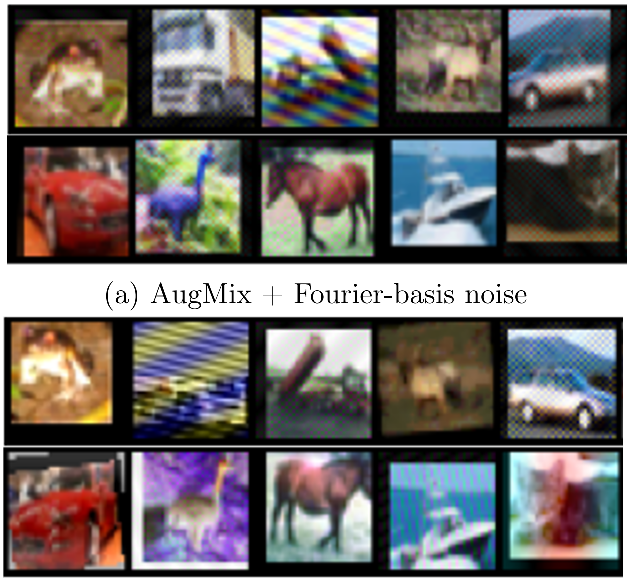
This experiment aims to find the optimal data augmentation policy that can improve the robustness of the machine learning model to corruptions. The approach uses Reinforcement Learning to search for an optimal policy. And then the performance of searched augmentation policy is evaluated. The results obtained from this policy are compared with other models to assess its effectiveness of enhancing model robustness. The experiment also includes evaluating the performance of combinations of different augmentation techniques, such as AutoAugment and AugMix, with the optimal policy found.

Search process: To find the best augmentation policy, the same algorithm discussed in Section 3.2.2 is used.

Augmentation setup: This experiment includes the use of the optimal policy found through the search process and the combination of this policy with other augmentation techniques such as AutoAugment and AugMix. The combination process involves adding Fourier-basis noise after applying the transformations offered by AutoAugment and AugMix. The results of this combination are shown in Figure 6 and demonstrate the effect of combining these techniques on the visualization of images. Before applying these augmentations, the transformation for all models consists of padding, random horizontal flipping, and random cropping.

Training: The training process in Experiment 2 is identical to that of Experiment 1.

Testing: In the experiment's test phase, several models' performance is evaluated and compared. These models include the baseline model, the optimal policy found through the search process, CutMix, MixUp, AugMix, AugMax, and AutoAugment. The comparison process is designed to analyze the effectiveness of combining AutoAugment and AugMix with Fourier-basis noise in mitigating the trade-off between low-frequency and high-frequency corruptions. Additionally, the experiment compares the test accuracy of the models on clean images with other models based on the Wide ResNet-28-10 architecture. To evaluate models' sensitivity to Fourier-basis noise, we test models on images with additive Fourier-basis noise with magnitude 1 to 5 and draw the heat maps as visualization.



(a) AugMix + Fourier-basis noise

(b) AutoAugment + Fourier-basis noise

Figure 6: (a) shows examples of images with AugMix and Fourier-basis transformations, (b) shows examples of images with AutoAugment and Fourier-basis transformations.

5 RESULTS

5.1 Experiment 1

Table 2 presents results of test accuracy on clean images and corruption error on CIFAR-10-C for five models. Among five models, **High** has the

(a) Test accuracy on clean images and corrupted images of every model, and corruption error to noise and blur corruption.

model	acc	acc	noise				blur				
	(clean)	(corrupted)	gaussian	shot	speckle	impulse	gaussian	zoom	defocus	motion	glass
Baseline	90.5	66.7	100	100	100	100	100	100	100	100	100
Uniform	89.4	70	55	58	61	69	110	122	118	114	80
Low	**90.9**	66.1	97	101	100	90	102	108	105	**98**	105
Mid	87.6	67.3	58	62	66	72	127	135	139	127	75
High	88.4	**74.9**	**48**	**53**	**58**	**61**	**84**	**86**	**92**	101	**48**

(b) Corruption error of every model to weather and digital corruption and mean corruption error of all corruptions.

model	weather					digital					mCE
	snow	frost	fog	brightness	spatter	contrast	elastic	pixelate	jpeg	saturate	
Baseline	100	100	100	100	100	100	100	100	100	100	100
Uniform	74	78	107	100	103	127	105	91	106	91	90
Low	104	115	**91**	**85**	106	**95**	108	108	123	**69**	101
Mid	83	94	140	115	100	150	120	83	107	108	98
High	**70**	**67**	123	116	**75**	119	**100**	**46**	**78**	115	**75**

Table 2: Test accuracy on CIFAR-10, mean corruption error and corruption error of every model to noise, blur, weather and digital corruption. The value of mCE and CE are judged based on the Natural model, if the value is below 100 means the model outperforms the baseline model, while value is above 100 means worse. The best results for each type of corruption are marked in bold. We calculate the average from 5 runs to obtain our results.

best mCE and highest accuracy 74.9% on corrupted images, and **High** performs best in the face to fourteen corruptions. In particular, adding high-frequency noise to model significantly improves robustness to noise corruption. On the other hand, **Low** has the lowest error rate for some corruption types such as fog, brightness, and saturate but performs worst overall. The mCE of **Uniform** and **Mid** are 90 and 98, respectively, do not significantly improve the models' robustness. As for the test accuracy on clean images, the **Low** is the best with 90.9%, and the difference in test accuracy between these models is not very large.

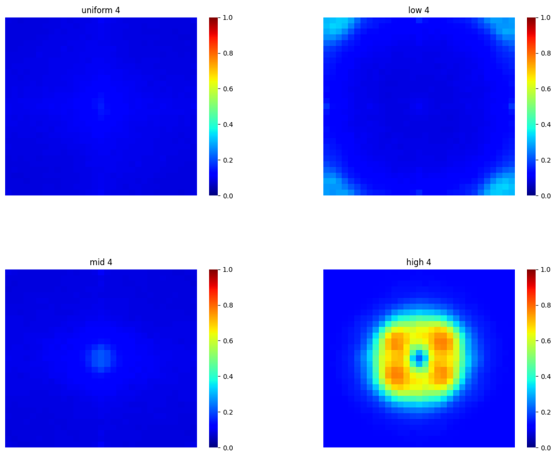


Figure 7: Fourier heat maps from Uniform, Low, Mid, and Low. The magnitude of Fourier-basis noise is 4.

The heat maps in Figure 7 display the models' sensitivity to 31*31 noise frequencies. **Uniform** shows low error rate on all noises, **Low** shows its robustness to noise except high-frequency noise, **Mid** shows low error rate except in very low area. And **High** only improves its robustness to high-frequency noise. The mCE of five models to each type of corruption are illustrated by Figure 8. The results are discussed separately based on the type of noise.

Noise: For the noise type, **High** shows the lowest mCE and the smallest error rate for all types of noise. **Uniform** and **Mid** also improve robustness to noise corruption. On the other hand, **Low** performs the worst, with significantly higher CE values than all the other models.

Blur: Regarding the type of blur, **High** shows the lowest CE for all subcategories of blur except for motion, while **Low** has the lowest CE for motion blur. **Uniform** and **Mid** have a lower CE than the baseline model for glass blur but they do not improve robustness because they do worse than the baseline model for other corruptions. **Mid** has the worst performance, with a mCE of 120.

Weather: For the weather type, **High** has the best performance, and **Uniform** follows closely behind. **High** has the lowest CE for snow, frost, and spatter, while **Low** has the lowest CE for fog and brightness. Only **Mid** performs worse than the baseline model. Adding noise of mid-high frequency improves models' robustness to snow and frost, while achieves a opposite effect on fog and brightness.

Digital: For the digital type, **High** is the only model that outperforms the baseline model. And for pixelate and jpeg, **High** achieves the lowest error rate. In terms of contrast and saturation, the model performs best with low-frequency noise as compared to other frequency ranges. However, **Mid** has a CE of 150 on contrast and **High** has a CE of 115 on saturate.

Overview: Overall, only **High** performs better than the baseline model. And apart from facing noise, this model does not improve robustness much in the face of the other three types of corruption.

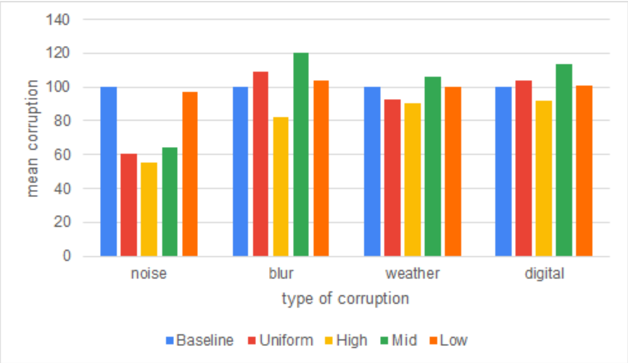


Figure 8: The mCE of models facing each type of corruption.

5.2 Experiment 2

5.2.1 Searched augmentation policies

The algorithm searches three distinct search spaces, which vary depending on how the noise is grouped: either by frequency, by frequency with further subdivision based on phase into four quadrants, or by dividing noise with identical frequency into eight equal parts. These three search space contains 22, 85, and 165 transformation candidates, respectively. The resulting augmentation policies are named FB1, FB2, and FB3. The choice of frequencies and its corresponding probability and magnitude of FB1 are shown in Table 5.2.1. The details of FB2 and FB3 are introduced in Appendix A.

	Operation1	Operation2
1	FB_5(0.4,4.24)	FB_15(0.71,1.64)
2	FB_9(0.02,2.2)	FB_22(0.86,3)
3	FB_4(0.77,2.48)	FB_8(0.55,3.45)
4	FB_15(0.2,1.31)	FB_21(0.64,2.94)
5	FB_13(0.39,4.56)	FB_3(0.96,3.59)
6	FB_2(0.53,4.39)	FB_15(0.7,3.58)
7	FB_22(0.47,4.63)	FB_14(0.18,2.21)
8	FB_2(0.66,3.68)	FB_9(0.1,4.26)

Table 3: Searched augmentation policies FB1. $FB\_5(0.4, 4.24)$ means adding a Fourier-basis noise in group 5 with probability of 0.4, and magnitude of 4.24.

5.2.2 Performance of searched policies

Table 4 shows the results of Experiment 2. Compared to the baseline model's accuracy of 90.5% on CIFAR-10, the AutoAugment (AA) model achieves the highest accuracy with 94.1%. AA combined with **FB2** has the second-highest accuracy of 94%. The performance of these models on clean images is similar, with accuracy ranging from 90% and 94%. However, these models perform very differently when faced with corrupted data. From a lowest of 64.3%(CutMix) to a highest of 85.8%(AA+**FB2**). As expected, training with Fourier-basis noise is effective augmentation. Fourier-basis noise training is effective in augmenting images, **FB1**, **FB2**, and **FB3** improve robustness to corruptions, with mCE scores of 72, 72, and 75, respectively. Augmentation such as AugMax and CutMix behave similarly to the baseline model, and CutMix even has a mCE higher than 100. AA and AugMix, which apply diverse transformations during the training phase, perform better than Fourier-basis augmentation when facing corruptions. Robustness has been significantly improved when combining AA or AugMix with Fourier-basis augmentation. The last six rows of Table 4 illustrate the performance improvement with AA+**FB2** having the lowest mCE of 42, followed by AugMix+**FB2**, with an mCE of 43. In addition, the error rates of the last six models are reduced by half compared to the baseline model, which is a significant improvement. The results will be discussed separately according to the type of noise.

Noise: AugMix combined with the noise type **FB2** achieves the lowest cross-entropy (CE) for all types of noise corruption. In contrast, the corruption error of CutMix is higher than the baseline model for all noise except impulse. Models with

(a) Test accuracy on clean images and corrupted images of every model, and corruption error to noise and blur corruption.

model	acc	acc	noise				blur				
	(clean)	(corrupted)	gaussian	shot	speckle	impulse	gaussian	zoom	defocus	motion	glass
Baseline	90.5	66.7	100	100	100	100	100	100	100	100	100
FB1	91.6	75.8	40	43	46	64	101	107	103	97	72
FB2	91.1	76	44	47	51	61	91	95	92	87	60
FB3	91.6	74.9	50	52	56	65	91	92	93	88	73
AugMax	91.4	68	97	98	99	90	98	92	96	95	97
CutMix	93.5	64.3	131	136	135	98	115	110	107	98	110
MixUp	93.7	74	86	85	85	85	90	83	80	73	70
AA	**94.1**	79.5	82	79	76	56	42	41	43	54	87
AugMix	92.3	79.9	59	56	54	61	39	43	43	49	74
AA+FB1	93.8	85	32	33	35	43	**33**	43	**39**	41	59
AugMix+FB1	93.1	85	30	**31**	**32**	42	35	41	40	40	50
AA+FB2	94	**85.8**	31	33	35	44	36	40	40	40	49
AugMix+FB2	92.7	85.4	**29**	**31**	33	**41**	36	**39**	40	**39**	**38**
AA+FB3	93.8	83.5	40	41	43	49	38	43	43	44	66
AugMix+FB3	92.6	84	34	35	36	44	36	41	41	41	53

(b) Corruption error of every model to weather and digital corruption and mean corruption error of all corruptions.

model	weather					digital					mCE
	snow	frost	fog	brightness	spatter	contrast	elastic	pixelate	jpeg	saturate	
Baseline	100	100	100	100	100	100	100	100	100	100	100
FB1	65	65	82	83	96	94	101	61	74	71	72
FB2	62	63	99	90	77	106	86	66	67	85	72
FB3	66	71	95	83	81	101	86	77	71	78	75
AugMax	107	96	94	88	93	99	93	91	97	88	96
CutMix	92	117	96	72	62	93	97	89	107	78	107
MixUp	65	66	69	67	78	66	73	80	75	72	78
AA	69	76	52	**55**	**51**	30	68	93	79	55	61
AugMix	72	69	72	74	65	65	68	52	72	79	60
AA+FB1	48	43	**39**	57	58	**27**	67	62	66	**50**	45
AugMix+FB1	49	47	54	67	63	57	61	43	56	61	45
AA+FB2	**43**	**39**	47	**55**	52	28	**57**	59	51	52	**42**
AugMix+FB2	48	45	69	72	57	66	**57**	**36**	**49**	72	43
AA+FB3	50	50	47	56	56	28	62	73	60	51	49
AugMix+FB3	53	53	67	72	64	66	60	42	53	70	48

Table 4: Test accuracy on CIFAR-10, mean corruption error and corruption error on CIFAR-10-C of models. FB1, FB2, and FB3 are different models that are trained using different augmentation policies. We calculate the average from 5 runs to obtain our results.

models	Noise	Blur	Weather	Digital
Baseline	100	100	100	100
FB1	48.25	96	78.2	80.2
FB2	50.75	85	78.2	82
FB3	55.75	87.4	79.2	82.6
AugMax	96	95.6	95.6	93.6
CutMix	125	87.8	87.8	92.8
MixUp	85.25	69	69	73.2
AA	73.25	53.4	60.6	65
AugMix	57.5	49.6	70.4	67.2
AA+FB1	35.75	43	49	54.4
AugMix+FB1	33.75	41.2	56	55.6
AA+FB2	35.75	41	**47.2**	**49.4**
AugMix+FB2	**33.5**	**38.4**	58.2	56
AA+FB3	43.25	46.8	51.8	54.8
AugMix+FB3	37.25	42.4	61.8	58.2

Table 5: The *mCE* of models facing each type of corruption.

Fourier-basis noise significantly reduce the model's sensitivity to noise attacks, reducing the CE to below 40. AugMax, Mixup, AA, and AugMix outperform the baseline model but have higher CE compared to **FB1**, **FB2**, and **FB3**.

Blur: For the blur type, AugMix+**FB2** performs the best, achieving a mCE of 38.4. CutMix performs the worst among all models, even worse than the baseline. AugMix + **FB1** acts the best on gaussian and defocus, while AugMix + **FB2** is the most robust model for zoom, motion, and glass. AA and AugMix have low error rates for all blur corruption except glass. However, by adding Fourier-basis noise, the error rates of AA and AugMix on glass are greatly reduced. Combining Fourier-basis noise with automatic augmentation reduces robustness to all types of blur. These results suggest that combining various augmentations can offset their drawbacks and enhance their advantages.

Weather: Among the weather types, AA combined with **FB2** achieves the best corruption error on snow, frost, and brightness, while AA +**FB1** shows its robustness to fog. AA has the best performance on spatter with a CE of 51, with AA + **FB2** being the second best with a CE of 52. Combining Fourier-basis noise with other augmentation improves robustness to weather corruption in most cases, although exceptions exist. For example, **FB1** performs poorly on spatter, indicating the negative effects of adding this set of noise. Almost all models improve robustness to weather corruption compared to the baseline model.

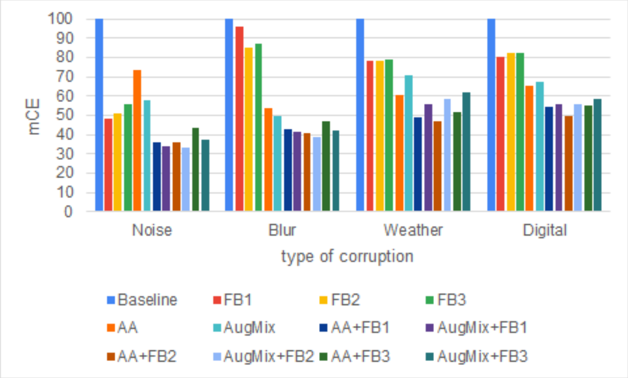


Figure 9: The mCE of models facing each type of corruption.

Digital: For the digital type, the results also demonstrate the positive impact of adding noise in improving robustness. AugMix combined with noise type **FB2** showed a reduction in error rates for elastic, pixelate, and jpeg from 68, 52, and 72 to 57, 36, and 49, respectively, making it the best model for these types of corruption. AA combined with **FB1** achieves the lowest CE for contrast and saturate. It should be noted that adding **FB2** and **FB3** causes a reduction of robustness to contrast for AugMix but not for AA. Other augmentations, such as AugMax and CutMix, are more sensitive in the face of digital corruption.

Overview: The mean corruption error for AA, AugMix, and noise-augmented models for each corruption type is presented in Figure 9. In general, the improvement in robustness from combining diverse augmentations can be observed for every corruption type. For example, combining Fourier-basis augmentation with AA and AugMix significantly reduced corruption errors for blur corruptions. Although the improvement is small for the other three types of corruption, it still demonstrates the positive impact of adding Fourier-basis noise.

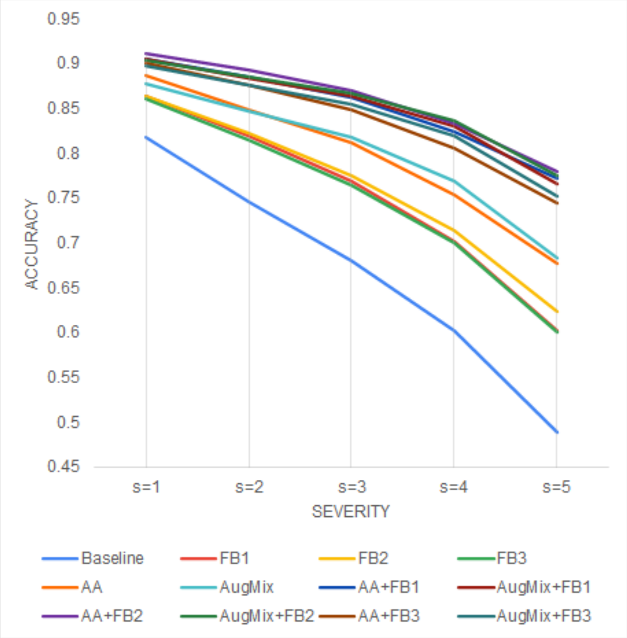


Figure 10: Average accuracy on CIFAR-10-C of all corruptions over all five severity levels for all models.

Corruption severity: The chart depicted in Figure 10 illustrates the decline in test accuracy on CIFAR-10-C for each model as the severity of corruption increases. The results indicate that the accuracy of the baseline model significantly drops with an increase in severity. Similarly, single augmentation models experience a decrease in accuracy, but AA and AugMix models show a slower decline compared to Fourier-basis augmentation. In contrast, mixed augmentation models exhibit higher accuracy levels across all severity levels, with a gradual reduction in accuracy. The AA+**FB2** model is the most robust among all models, with an accuracy rate of 78% even at the highest corruption severity level of 5.

model	acc(%)
AA+**FB2**	94
WRN	96
deep ensemble	96.6
hyper-deep ensembles	96.5

Table 6: Results for Wide ResNet-28-10 on CIFAR-10 using different methods.

Test accuracy: We search for studies that used Wide ResNet-28-10 as a reference model, and compare the classification accuracy on CIFAR-10 with our models. Among models using Fourier-basis augmentations, the AA+**FB2** model achieves the highest accuracy in our second experiment, reaching 94%. The Wide ResNet-28-10 model in [36] attains an accuracy of 96%, and according to [37], the Deep Ensembles model achieves an accuracy of 96.6% by aggregating predictions from several stochastically gradient descent trained models. Additionally, with a straightforward approach that involves random search over various hyperparameters, Hyper-deep ensembles [38] achieves an accuracy of 96.5%. These results indicate that our model has a lower test accuracy than the other models, so our future work could start by improving the accuracy.

6 DISCUSSION

In Experiment 1, different frequency ranges of Fourier-basis noise are used as augmentation to improve the model's robustness. The results indicate that adding high-frequency noise was most effective in improving the model's robustness against certain corruptions, while adding low-frequency noise was not helpful. The results in Experiment 1 can be further understood by analyzing Figure 11. The clean images have most of their energy concentrated in low frequencies; This is why **Low** has the highest text accuracy on CIFAR-10. The corruptions used in the experiment can be grouped into two categories: those that primarily concentrated on low frequencies (e.g., brightness, contrast, fog) and those that mainly focus on high frequencies (e.g., impulse, shot, speckle). This supports that low-frequency noise improves the model's performance against corruptions that focuse on low frequencies. In contrast, high-frequency noise is more effective against corruptions that concentrated on high frequencies. Specifically, **Low** has the lowest error rate for fog, contrast, and brightness corruptions, while **High** had the best performance for additive noise corruptions. The heat maps in Figure 2 show that the baseline model is highly sensitive to additive noise, particularly noise with high-frequencies. A possible explanation for the results is that the baseline model may be biased against the low-frequency components of the images and therefore ignores the high-frequency components. As a result, when the model encounters high-frequency noise in the test, it cannot handle it effectively, resulting in poor performance. This bias may explain why adding high-frequency noise to the training data improved the model's performance against high-frequency noise in testing.

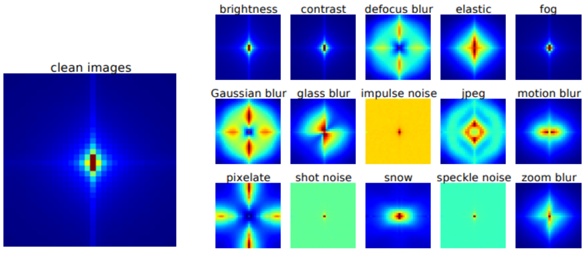


Figure 11: Fourier spectrum of natural images in CIFAR-10 and corruptions in CIFAR-10-C [24].

Experiment 1 shows that using a single augmentation technique does not make the model robust against low-frequency and high-frequency corruptions. This trade-off highlights the importance of using a diverse augmentation when training models to be strong against various types of corruption. In Experiment 2, Reinforcement Learning is used to explore the optimal augmentation policy by exploring combinations of Fourier-basis noise. The searched policy contains eight sub-policies, with each sub-policy adding two Fourier-basis noises of different frequencies. During augmentation, the model selects a sub-policy randomly for each image. Using the searched policy for data augmentation can mitigate the trade-off observed in Experiment 1. By combining multiple sub-policies that add noise in different frequencies, the model's robustness to high-frequency corruptions can be improved while reducing error rates for low-frequency corruptions. In addition, the augmentation techniques used in the sub-policies are tailored to different frequencies of noise, allowing for the improved overall performance of the model.

The model can become even more robust by combining different data augmentations, including AugMix, AutoAugment, and Fourier-basis augmentation. Table 4 demonstrates the effectiveness of Fourier-basis noise augmentation, as the average error rates for corruptions have significantly decreased after adding Fourier-basis noise. Additionally, the model's performance is not limited to high-frequency corruptions such as noise and blur but extends to low-frequency corruptions like fog, brightness, and contrast. The addition of Fourier-basis noise augmentation exposes the model to a wider range of variations in the data, further increasing its robustness to different types of corruption. As a result, Fourier-basis noise augmentation can help make the model's strengths more prominent while reducing its weaknesses, effectively mitigating the performance trade-off between high-frequencies and low-frequencies corruptions.

7 CONCLUSION

This section aims to provide a summary of the study and address the research questions, after which potential areas of future research is explored.

In summary, this study focuses on improving model robustness and aims to answer research questions on this topic. Regarding the first research question, the study investigates the effect of introducing noise of varying frequencies on model robustness. It was found that adding high-frequency noise is the most effective way to enhance model performance. This is because the original image is predominantly comprised of low frequencies, and high-frequency noise can add extra information to the image, making it more resilient against high-frequency corruptions.

The results of the second experiment answer the second research question. In experiment 2, a Fourier-basis augmentation is searched using the FAA searching algorithm. The algorithm applies Reinforcement Learning to investigate various frequencies, to identify the combination of noise that produces the highest classification performance. Combining Fourier-basis augmentation with other augmentation, such as AugMix and AutoAugment, can further strengthen the conclusion that adding different noise is an effective approach to enhance model robustness. The combination of diverse data augmentation can produce better results. Also, mixed augmentation mitigates the trade-off between performance in the presence of high-frequency and low-frequency corruptions.

However, there is always room for improvement and further research, such as exploring a more extensive search space and adjusting the structure of the augmentation policy. Furthermore, changing the parameters, such as optimizer, learning rate, during the training phase may increase the classification accuracy, which may further mitigate the trade-off of performance between clean set and corrupted set.

References

[1] Yu Su and Frédéric Jurie. Improving image classification using semantic attributes. *International journal of computer vision*, 100(1):59–77, 2012.

[2] Yuji Tokozume, Yoshitaka Ushiku, and Tatsuya Harada. Between-class learning for image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5486–5494, 2018.

[3] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *International Conference on Machine Learning*, pages 5389–5400. PMLR, 2019.

[4] Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of big data*, 6(1):1–48, 2019.

[5] Zeshan Hussain, Francisco Gimenez, Darvin Yi, and Daniel Rubin. Differential data augmentation techniques for medical imaging classification tasks. In *AMIA annual symposium proceedings*, volume 2017, page 979. American Medical Informatics Association, 2017.

[6] Maayan Frid-Adar, Idit Diamant, Eyal Klang, Michal Amitai, Jacob Goldberger, and Hayit Greenspan. Gan-based synthetic medical image augmentation for increased cnn performance in liver lesion classification. *Neurocomputing*, 321:321–331, 2018.

[7] Hoo-Chang Shin, Neil A Tenenholtz, Jameson K Rogers, Christopher G Schwarz, Matthew L Senjem, Jeffrey L Gunter, Katherine P Andriole, and Mark Michalski. Medical image synthesis for data augmentation

and anonymization using generative adversarial networks. In *International workshop on simulation and synthesis in medical imaging*, pages 1–11. Springer, 2018.

[8] Pieter M Blok, Frits K van Evert, Antonius PM Tielen, Eldert J van Henten, and Gert Kootstra. The effect of data augmentation and network simplification on the image-based detection of broccoli heads with mask r-cnn. *Journal of Field Robotics*, 38(1):85–104, 2021.

[9] Mei-Ling Huang, Tzu-Chin Chuang, and Yu-Chieh Liao. Application of transfer learning and image augmentation technology for tomato pest identification. *Sustainable Computing: Informatics and Systems*, 33:100646, 2022.

[10] MAA Ghaffar, A McKinstry, T Maul, and TT Vu. Data augmentation approaches for satellite image super-resolution. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 4:47–54, 2019.

[11] Mark Pritt and Gary Chern. Satellite image classification with deep learning. In *2017 IEEE Applied Imagery Pattern Recognition Workshop (AIPR)*, pages 1–7, 2017.

[12] Kosaku Fujita, Masayuki Kobayashi, and Tomoharu Nagao. Data augmentation using evolutionary image processing. In *2018 Digital Image Computing: Techniques and Applications (DICTA)*, pages 1–6, 2018.

[13] Logan Engstrom, Brandon Tran, Dimitris Tsipras, Ludwig Schmidt, and Aleksander Madry. A rotation and a translation suffice: Fooling cnns with simple transformations. 2018.

[14] Agnieszka Mikołajczyk and Michał Grochowski. Data augmentation for improving deep learning in image classification problem. In *2018 International Interdisciplinary PhD Workshop (IIPhDW)*, pages 117–122, 2018.

[15] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

[16] Zhi Zhang, Tong He, Hang Zhang, Zhongyue Zhang, Junyuan Xie, and Mu Li. Bag of freebies for training object detection neural networks. *arXiv preprint arXiv:1902.04103*, 2019.

[17] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6023–6032, 2019.

[18] Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation strategies from data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 113–123, 2019.

[19] Leslie Pack Kaelbling, Michael L Littman, and Andrew W Moore. Reinforcement learning: A survey. *Journal of artificial intelligence research*, 4:237–285, 1996.

[20] Vincent François-Lavet, Peter Henderson, Riashat Islam, Marc G Bellemare, Joelle Pineau, et al. An introduction to deep reinforcement learning. *Foundations and Trends® in Machine Learning*, 11(3-4):219–354, 2018.

[21] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019.

[22] Dan Hendrycks, Norman Mu, Ekin D Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. Augmix: A simple data processing method to improve robustness and uncertainty. *arXiv preprint arXiv:1912.02781*, 2019.

[23] Ryan Soklaski, Michael Yee, and Theodoros Tsiligkaridis. Fourier-based augmentations for improved robustness and uncertainty calibration. *arXiv preprint arXiv:2202.12412*, 2022.

[24] Dong Yin, Raphael Gontijo Lopes, Jon Shlens, Ekin Dogus Cubuk, and Justin Gilmer. A fourier perspective on model robustness in computer vision. *Advances in Neural Information Processing Systems*, 32, 2019.

[25] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil

Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

[26] Luis Perez and Jason Wang. The effectiveness of data augmentation in image classification using deep learning. *arXiv preprint arXiv:1712.04621*, 2017.

[27] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[28] Ali Shafahi, Mahyar Najibi, Mohammad Amin Ghiasi, Zheng Xu, John Dickerson, Christoph Studer, Larry S Davis, Gavin Taylor, and Tom Goldstein. Adversarial training for free! *Advances in Neural Information Processing Systems*, 32, 2019.

[29] Eric Wong, Leslie Rice, and J Zico Kolter. Fast is better than free: Revisiting adversarial training. *arXiv preprint arXiv:2001.03994*, 2020.

[30] Sungbin Lim, Ildoo Kim, Taesup Kim, Chiheon Kim, and Sungwoong Kim. Fast autoaugment. *Advances in Neural Information Processing Systems*, 32, 2019.

[31] Toan Tran, Trung Pham, Gustavo Carneiro, Lyle Palmer, and Ian Reid. A bayesian data augmentation approach for learning deep models. *Advances in neural information processing systems*, 30, 2017.

[32] Ronald Newbold Bracewell and Ronald N Bracewell. *The Fourier transform and its applications*, volume 31999. McGraw-Hill New York, 1986.

[33] Chuan Guo, Jared S Frank, and Kilian Q Weinberger. Low frequency adversarial perturbation. *arXiv preprint arXiv:1809.08758*, 2018.

[34] Tonmoy Saikia, Cordelia Schmid, and Thomas Brox. Improving robustness against common corruptions with frequency biased models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10211–10220, 2021.

[35] Daniel Hendrycks. Cifar-10-c and cifar-10-p, Jan 2019.

[36] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016.

[37] Michael Dusenberry, Ghassen Jerfel, Yeming Wen, Yian Ma, Jasper Snoek, Katherine Heller, Balaji Lakshminarayanan, and Dustin Tran. Efficient and scalable bayesian neural nets with rank-1 factors. In *International conference on machine learning*, pages 2782–2792. PMLR, 2020.

[38] Florian Wenzel, Jasper Snoek, Dustin Tran, and Rodolphe Jenatton. Hyperparameter ensembles for robustness and uncertainty quantification. *Advances in Neural Information Processing Systems*, 33:6514–6527, 2020.

A Searched policies

	Operation1	Operation2
1	FB_10_3(0.87,0.49)	FB_12_3(0.4,3.74)
2	FB_20_4(0.52,4.84)	FB_18_1(0.73,3.12)
3	FB_3_2(0.35,4.47)	FB_7_1(0.89,3.75)
4	FB_12_4(0.75,3.27)	FB_18_2(0.93,1.07)
5	FB_10_4(0.18,3.95)	FB_7_1(0.72,0.31)
6	FB_17_2(0.77,4.32)	FB_9_4(0.92,1.12)
7	FB_18_4(0.01,0.81)	FB_13_4(0.46,1.61)
8	FB_11_2(0.91,2.02)	FB_20_3(0.01,1.65)

Table 7: Searched augmentation policies FB2. The notation $FB\_10\_3(0.87, 0.49)$ refers to the addition of Fourier-basis noise with a probability of 0.87 and magnitude of 0.49 to group 10, where the phase ranges from 180° to 270°. The four phases, represented by the numbers 1, 2, 3, and 4, cover the ranges 0-90°, 90-180°, 180-270°, and 270-360°.

	Operation1	Operation2
1	FB_13_6(0.18,3.88)	FB_6_7(0.97,3.45)
2	FB_2_6(0.64,3.57)	FB_5_2(0.47,1.74)
3	FB_7_6(0.65,2.14)	FB_18_6(0.87,4.87)
4	FB_17_4(0.7,4)	FB_21_4(0.04,1.63)
5	FB_15_7(0.25,4.35)	FB_9_4(0.99,1.89)
6	FB_18_7(0.75,3.19)	FB_9_6(0.78,1.36)
7	FB_22_6(0.57,1.51)	FB_21_3(0.24,1.32)
8	FB_21_1(0.01,1.89)	FB_11_1(0.45,2.71)

Table 8: Searched augmentation policies FB3. The notation $FB\_13\_6(0.18, 3.88)$ refers to the addition of Fourier-basis noise with a probability of 0.18 and magnitude of 3.88 to group 13, where the phase ranges from 225° to 270°. The eight phases, represented by the numbers 1-8, cover the ranges 0-45°, 45-90°, 90-135°, 135-180°, 180-225°, 225-270°, 270-315°, and 315-360°.

B Heat maps of models

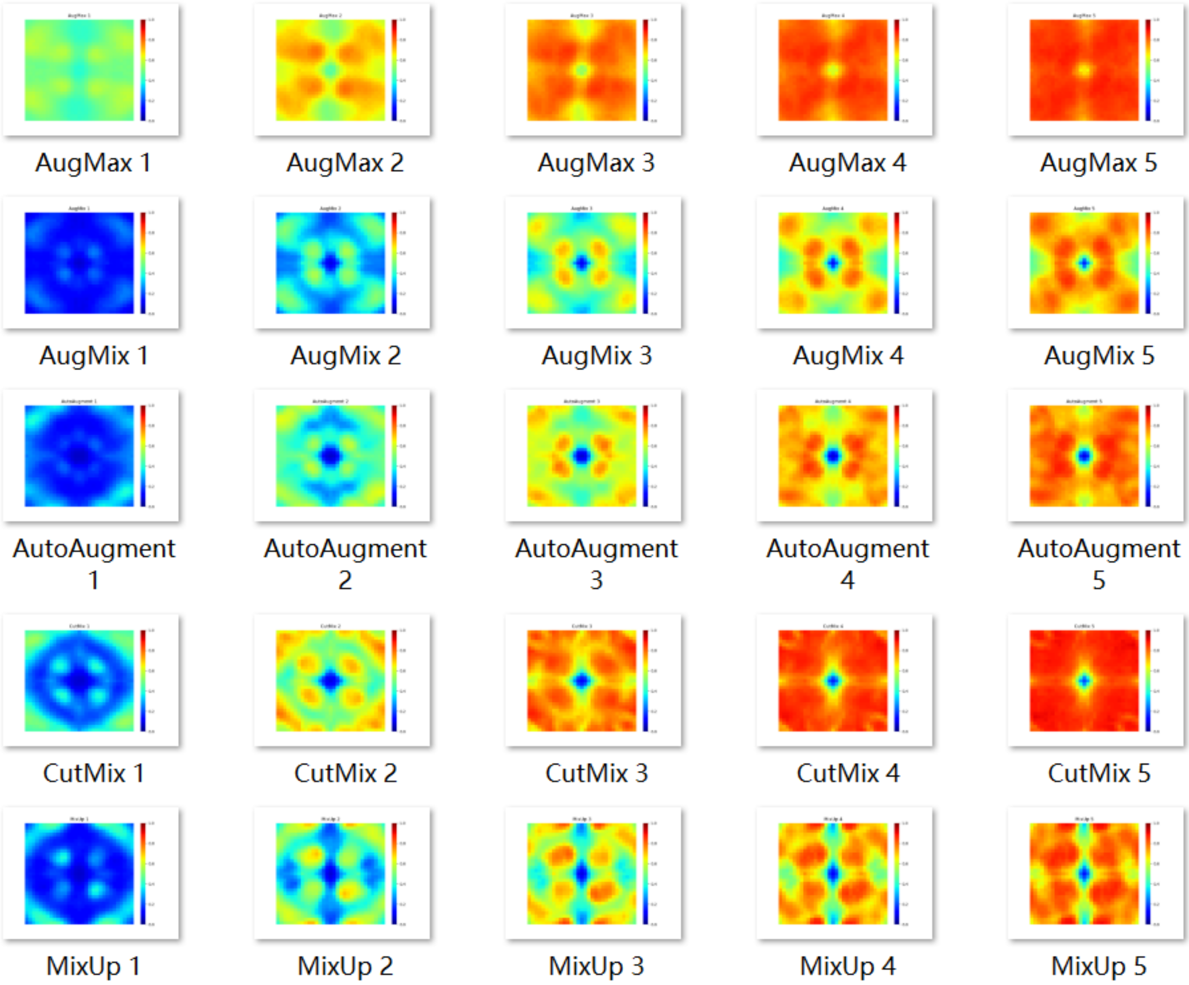


Figure 12: Fourier heat maps from AugMax, AugMix, Autougment, CutMix, and MaxUp. The numbers 1 to 5 indicate the magnitude of Fourier-basis noise. Except for AugMax, these models perform well when the magnitude is 1. However, as the noise magnitude increases, the models become less effective at handling medium and high frequency noise.

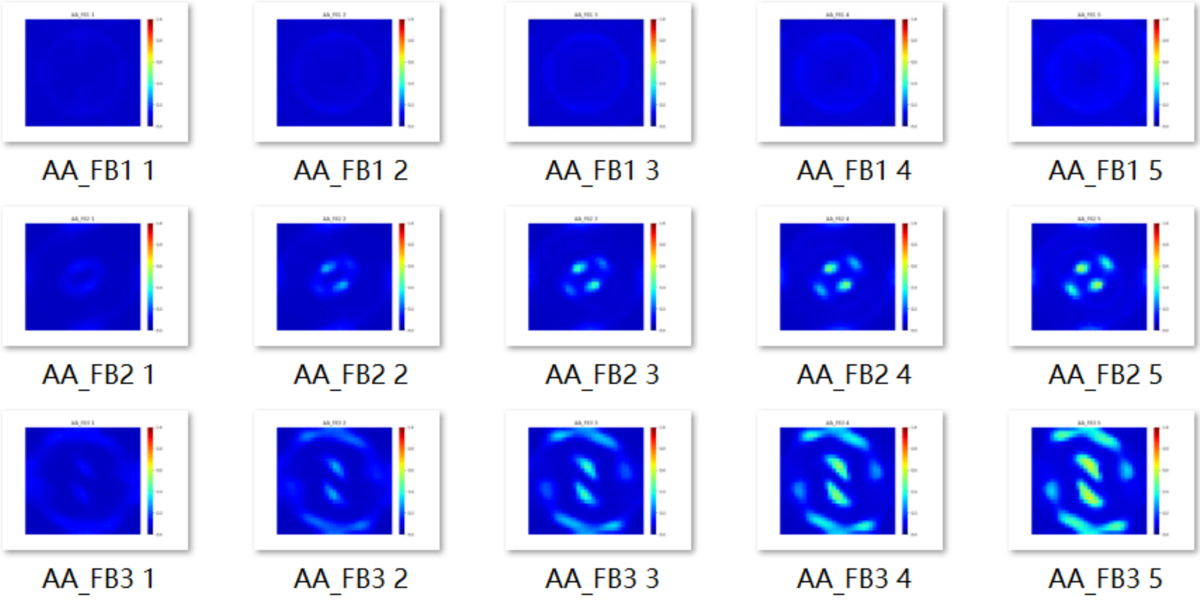


Figure 13: Fourier heat maps from AutoAugment(AA) combined with Fourier-basis noise. Overall AA+FB1 is robust to Fourier-basis noise attacks, except at middle frequencies, which may be related to the lack of noise in policy 1 that adds that frequency. AA+FB2 performs poorly in the face of low-frequency noise with phases in quadrants two and four, and AA+FB3 has poor performance on quadrants one and three. The performance was much better than AA, proving the positive effect of the combination.

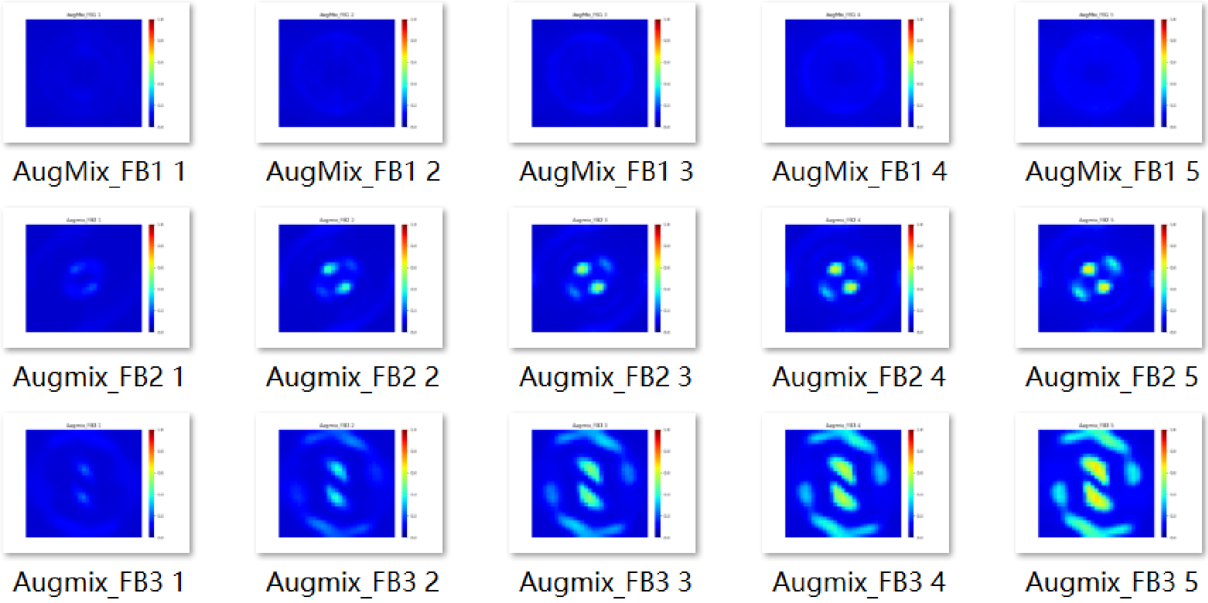


Figure 14: Fourier heat maps from Augmix combined with Fourier-basis noise. The performance similarly to AA+FB, and was much better than Augmix.