Data Augmentation using Fourier-Basis Noise

OMID FATTAHI MEHR, University of Twente, The Netherlands

Data augmentation has become an important tool to improve the robustness of a model against corruptions in data and adversarial attacks. Most of the previous research has focused on approaching data augmentation from the spatial domain. This paper utilizes Fourier-Basis noise to augment images. Fourier-basis noise consists of frequencies added to an image. We define a new selection method by creating predefined frequency sets on different criteria. These sets are simpler than other established methods and have many possible configurations in which different frequencies can be combined. We conduct experiments that are used to evaluate the effect of those sets on the robustness against common corruptions. The results show that high frequency noise augmentation provides a significant improvement in robustness against corruptions compared to the baseline model. This research shows positive results on the effect, Fourier-Basis noise can have on corruption robustness and suggests further exploration of the method for a better understanding of its impact.

Additional Key Words and Phrases: Data Augmentation, Noise Robustness, Fourier-Basis Noise

1 INTRODUCTION

Convolutional neural networks are commonly used for image recognition, but fail to generalize well outside of training data and are susceptible to even small common corruptions occurring in nature [2, 10]. Therefore, they cannot be employed in safety critical environments, such as autonomous driving [5, 27, 30, 31]. Additionally, in the medical domain, noise in X-ray images or histograms can cause unexpected false classifications [6]. Common corruptions can occur in nature, for example frosting on the camera or sunlight leading to high brightness in the image [14]. Various techniques, such as expanding the training data, are utilized to improve robustness against typical corruptions like noise or blur. Data augmentation is a simple and efficient method to prevent overfitting [20, 24]. In the past few years, it has been used more frequently to enhance robustness against common corruptions and adversarial attacks [28, 36]. Originally, simple transformations, such as flipping [20], cropping [12], and more have been utilized to extend the dataset. This approach is tedious, as it requires experts to select the appropriate transformations manually for each dataset. As a result, new strategies were created that adapt to various datasets more effectively.

One approach is the automation of the transformation selection, where parameters of different transformations are learned from the training data. The parameters can be a transformation's probability or intensity when applied to a single image. Different methods have been proposed [3, 16, 22, 32], with the most prominent being AutoAugment [3], developed by Cubuk et al., which uses policies that can adapt to different datasets. The learning process requires another DNN to determine the appropriate combination of transformations. Thus, these methods cannot scale well to larger datasets [4]. RandAug [4], based on AutoAugment, reduces the number of variables that need to be learned with randomization, which decreases the overall search space. Although it improves time efficiency and scalability, the number of transformations is still limited, making it not very flexible.

AugMix [15], proposed by Hendrycks et al., takes randomly different augmentations to apply to an image and creates a random mix, which is then again merged with the original image. This method has shown improvements in various benchmarks without losing accuracy to uncorrupted images [15, 36]. Another method, AugMax [35], by Wang et al., combines the approach of AugMix with adversarial training to achieve better coverage for weaknesses against adversarial examples.

The methods that have been presented so far are part of an active field of research that aims to develop robust machine learning systems. Most of the approaches are based on spatial transformations. This research uses Fourier-Basis noise in the frequency domain, which has been explored by Soklaski et al. [29]. They combine Fourier-Basis noise with AugMix to improve robustness against "Fourier-Basis attacks". These kinds of attacks consist of images added with frequency noise in the low, mid, or high frequency range. They can cause heavy degradation to model performance. Although this method shows success with regard to common corruptions, it uses random combinations of different frequencies, which can be either simple or complex.

This paper explores simple combinations of frequencies. We propose a new method that selects different sets of frequencies from a predefined limited set of all available frequencies that can be applied to an image. This approach is simpler, in regard to the implementation intricacy and computation time. The method consists of the manual creation of different sets of frequencies based on certain criteria that are going to be combined and evaluated in experiments. The advantages include easier training and more flexibility to precisely target the weaknesses of a model. We aim to not increase complexity, but rather see if simple predefined sets can have similar performance to already established methods. If it can provide similar improvements, then it will not only give an insight into the effects of different frequency ranges on images but also be much more efficient compared to other, more complex methods.

To be able to utilize the frequency domain and generate noise with different frequencies, we use a similar procedure as Yin et al. [36]. The dataset that is used for training and testing is the CIFAR-10 dataset [19] with 50000 training images and 10000 test images. The common corruptions that will be considered are categorized into Noise, Blur, Weather, and Digital corruptions, based on the CIFAR-10-C dataset by Hendrycks et al. [14]. In addition to selecting frequency sets and applying them to images, experiments will be conducted to test the capability and explore the different effects of Fourier-Basis noise on the robustness against common corruptions.

TScIT 37, July 8, 2022, Enschede, The Netherlands

The aims of this research can be formulated into three research questions.

- **RQ1**: How can frequencies be added to images as a means of augmenting them for training?
- **RQ2**: To what extent can Fourier-Basis noise augmentation improve the robustness to common corruptions of an image recognition model in comparison to a base model?
  - **RQ2.1**: Which frequencies contribute to an increased robustness?

The answer to **RQ1**, provides the basis for **RQ2** and **RQ2.1**. First, a baseline model will be used that consists of an image recognition model that is trained on the CIFAR-10 dataset without Fourier-Basis noise augmentation. The following experiments will then provide results that can be compared to the performance of the baseline model. Based on the results of **RQ2**, **RQ2.1** can be answered, depending on decreased or improved robustness.

2 RELATED WORK

This section looks at other methods that have been used for data augmentation and the link from common corruptions to adversarial attacks. This includes research that explores other data augmentation methods from the Fourier perspective. We also take a look at research that tries to improve robustness to corruptions by augmenting images with similar kinds of corruptions.

2.1 Data Augmentation Methods

Image Style Transfer [8] is a well-known method that can be used to merge a texture from one image with the shape of another to create a new image. Although it is more commonly used in art, it has also been employed as a data augmentation method [9]. Jackson et al. [17] use the shape of the original dataset together with random textures to augment images. It was found to be improving the robustness of the model when combined with other data augmentation methods. Adversarial examples are forms of noise injections that are used to perform an adversarial attack. In most cases, the noise cannot be picked up by the human eye. However, image recognition models are susceptible to it. This poses a risk, since one can inject noise into an image without changing the semantic content [28]. Consequently, it can be used to confuse a model to classify an image to the wrong category without the human eye being able to detect it from the input image [11]. The current state of literature does not have a complete agreement on whether robustness to common corruptions is related to robustness to adversarial examples. While some research finds evidence for a relation [7], others find the opposite [18, 21].

2.2 Frequency Domain

Previous research has tried to understand the relation between common corruptions and the frequency domain. Yin et al. [36] compared various currently used augmentation methods on frequency noise. The results showed that AutoAugment caused the model to be robust against more types of noise than other methods. In general, it has been found by Wang et al. [34] and others [37] that CNNs are able to capture the high frequency components of an image and are more vulnerable in that range. In other works, it has been also suggested to increase the sensitivity of a CNN to the low frequency range, since more robust models prefer low frequency information [26].

2.3 Augmentation with Corruptions

A simpler method is to directly add noise to images in order to improve robustness against similar types of noise [10]. Some literature suggests minor generalization [7]. Rusak et al. [25] used Gaussian and Speckle Noise to train a ResNet50 model on the ImageNet dataset, which showed good results for various corruptions in the ImageNet-C dataset. However, it did not perform well on certain blur corruptions. Lopes et al. [23] combined cutting and Gaussian noise to create Patch Gaussian noise, which is a scheme that applies noise to only certain parts of an image. In contrast, Vasiljevic et al. [33] showed that training on blur can improve the accuracy of the computer vision model when receiving blurred images as input.

3 METHODS

The general method used to augment images in this research can be divided into the noise generation process and the selection of frequencies. With the selection, noise is created that is applied to images. The noise generation step creates Fourier-Basis noise ranging from low to high frequency, which can be directly applied to images. The second step does not only include selecting but also layering frequencies, where multiple frequencies are combined in order to augment images with a greater variety of noise. Both methods are combined and used to augment images that are utilized to train a model. The experimental setup and the training are described in more detail in Section 4.

3.1 Noise Generation

The noise generation process is based on the method proposed in [36]. Yin et al. describe adding perturbations to images with $\widetilde{X}_{i,j} = X + rvU_{i,j}$, where X is the original image without noise. $U_{i,j}$ is a 2D Fourier-Basis function, generated from a spectrum matrix, with the frequency determined by the entry variables i and j. The variable r can have values randomly chosen between -1 and 1. The variable v determines the strength of the noise, which indicates how much noise is added, similar to the severity of the corruptions described in [14]. A visual representation of the noise generation process is shown in Figure 1. The sample space of the set of all 2D Fourier-Basis functions that can be selected contains 1024 functions because the size of the spectrum matrix is 32×32, similar to CIFAR-10 images. This means that for every entry in the matrix, there is one Fourier-Basis function. All of them are going to be used to evaluate the model's robustness to Fourier-Basis noise. The classification error rates are placed in an error matrix that will be used to select frequencies for augmentation. The Fourier heat map displays, based on the error matrix, the sensitivity of the model to low, mid, and high frequency noise. In the heat map, low frequencies are located in the center, while the highest frequencies are located at the edge of the map, corresponding to the spectrum matrix.

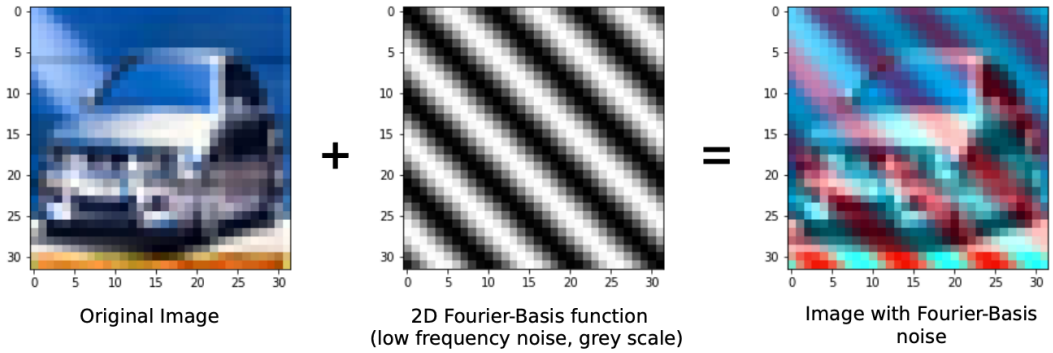
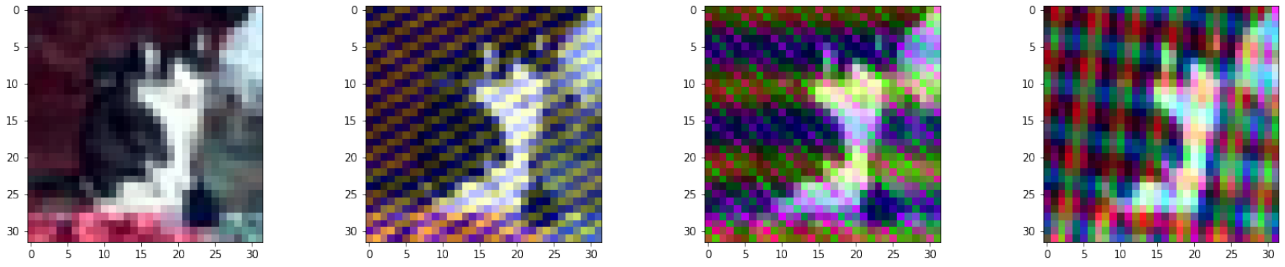


Fig. 1. The process of applying noise to an image, consists of the original image X and the 2D Fourier-Basis function (in this case, $4 * r * U_{4,4}$ with a strength of 4). Combining them results in an image augmented with Fourier-Basis noise $\widetilde{X}_{4,4}$. Generally, the 2D Fourier-Basis function is in RGB, but for visual purposes it is represented in grey scale.



(a) Natural (b) High Frequency (c) Small Low and High Frequency (d) Low, Mid and High Frequency

Fig. 2. An image from the CIFAR-10 dataset augmented with different Fourier-Basis noise. Image (b) is from experiment High(4) and adds high frequency noise with strength 4 (indicated by the experiment name) to image (a). Images (c) and (d) are from the experiments SLow(2.5, p=dec) & High(1.5) and L(0.5) & M(1) & H(2.5), which combine the Small Low and High frequency sets and Low, Mid, and High frequency sets, respectively.

3.2 Frequency Selection

Single frequencies can be chosen individually by using the spectrum matrix S. We denote the interval of integers between x and y, including x and y, as $\{x, y\}$. The spectrum matrix entries, denoted by $S(i, j)$, can have values ranging from 0 to 31, such that $i, j \in \{0, 31\}$. The lowest frequency is the zero frequency and is placed in the center of the matrix, corresponding to the entry $(16, 16)$. To simplify the indexing and selection of suitable frequencies, the range of indices is shifted such that the zero frequency lies in $(0, 0)$, so that $i, j \in \{-16, 15\}$.

Different criteria are required to select sets of frequencies that can be combined purposefully to create experiments. Five basic sets of frequencies have been created, denoted as high frequency, mid frequency, low frequency, small low frequency (SLow) and a set of frequencies based on the error matrix E, as previously mentioned in Section 3.1. The high frequency set includes all frequencies ranging between $i \in \{-16, -12\}$ and $i \in \{11, 15\}$, where $j \in \{-16, 15\}$. The mid-frequency set contains all frequencies ranging between $i \in \{-11, -6\}$ and $i \in \{5, 10\}$, where $j \in \{-11, 10\}$. Furthermore, the low frequency set contains the rest, such that $i, j \in \{-5, 4\}$. For all frequency sets, (i, j) and (j, i) are added to the resulting set. Duplicates of coordinates that occur in the set are removed, such that all frequencies only occur in a set once. The sets have been chosen in order to be flexible when creating experiments, while also not being too complex to combine and to use.

The small low frequency set is a modified low frequency set that consists of frequencies ranging between $i, j \in \{-1, 1\}$, so that it only covers the frequencies in the center. This is due to the sensitivity of many corruptions, such as Fog and Contrast from the CIFAR-10-C dataset [14], which are centered close to the zero frequency [36].

The set based on the error matrix is defined by setting an error threshold t, such that for every entry $(i, j) \in S$, if $E(i, j) >= t$, then (i, j) is used for augmentation. The basis functions contained in these sets can be utilized to add noise to the images. The strength of the noise can be set for each frequency or for each set of frequencies. For every epoch, a random frequency is selected from a set to augment a single image. Additionally, sets can be combined to add multiple frequencies of varying intensities to a single image. The probability of choosing a certain frequency for an image can be weighted separately for each set, but is by default uniformly random.

3.3 Evaluation Metrics

To evaluate the experiments, several metrics are utilized that reflect the performance of the resulting model against common corruptions. First, the accuracy of the natural test set is assessed with the classification accuracy in order to see the model's capability of classifying natural images, i.e., uncorrupted images. It measures the fraction of correctly classified images by the trained model. In addition to the test accuracy, a benchmark is needed that can measure the changes in robustness compared to the base model. In this regard, the *corruption error* with Equation 1 and *mean corruption error* with Equation 2, from [13] are used, referred to as **CE** and **mCE**. Both metrics are evaluated on the CIFAR-10-C dataset, which is a dataset that contains images with 15 different corruptions. They are categorized into Noise, Blur, Weather, and Digital. All 15 corruptions have a severity level s from 1 to 5, indicating the strength of the corruption. We define N to be a network trained with Fourier-Basis noise. The *corruption error* measures the error rate E summed over all five severity levels of a corruption c in comparison to the naturally trained ResNet18 model, referred to as *base*.

$$\mathrm{CE}_c^N = \sum_{s=1}^{5} \mathrm{E}_{s,c}^N / \sum_{s=1}^{5} \mathrm{E}_{s,c}^{base} \tag{1}$$

To be able to compare different augmentation strategies better, the *mean corruption error* is used, which is defined as the mean of all **CEs** from each corruption, where the number of corruptions is defined as R.

$$mCE^N = \frac{1}{R} \sum_{r=1}^{R} \mathrm{CE}_r^N \tag{2}$$

The Fourier heat map, seen in Figure 3, is the final metric and will be a visual aid. For each model, it provides information on the sensitive frequency areas and how the different augmentation techniques improve the robustness to high, mid, and low frequency range noise. The heat map will use a strength of 4.0 for the noise that is applied, and it will have a size of 31×31 to be able to center the zero frequency and therefore be symmetric about the origin.

4 EXPERIMENTS

Several experiments are conducted to investigate the effectiveness of Fourier-Basis noise on the robustness of a model. These experiments give an insight into what can improve robustness and what might be harmful and cause the opposite. A single experiment consists of three phases, namely the augmentation step, the training step, and the testing step.

4.1 Augmentation Setup

The augmentation step employs other methods in addition to Fourier-Basis noise. For all experiments, the augmentation procedure consists of padding (with four pixels), random horizontal flipping, and random cropping. Afterwards, the image is transformed to a tensor for further augmentation, with some form of Fourier-Basis noise, followed by normalization. The base model includes all augmentation procedures except for the addition of Fourier-Basis noise. Padding, flipping, and cropping are used to extend the training dataset and achieve better test accuracy. The experiments are assigned to categories based on the number of frequencies that are applied to a single image. For each image, up to three Fourier-Basis functions can be selected and applied, each with a different strength. We denote the three categories as *K1*, *K2*, and *K3*, indicating the number of frequencies that are layered on a single image.

4.1.1 K1. For the first category, *K1*, the four basic, previously discussed, *Low*, *Mid*, *High*, and *Error Matrix* based frequency sets are used to train the model. The Small Low frequency set is, in this case, not considered since it is designed to be used with other sets in K2 and K3. The noise of all four experiments has a strength of 4, which is based on the default value, proposed in [36], that is used to create the heat map. The error rate threshold is 0.5, which indicates that all frequencies with an error rate higher than or equal to 0.5 are considered for augmentation. An example of high frequency noise applied to an image can be seen in Figure 2(b). During the training, the model chooses uniformly at random one of the frequencies from the selected set. As a result, the augmentation process is not static, since each image can be applied with different frequency noise in each epoch.

4.1.2 K2. The second category, *K2*, combines two frequency sets. The selected sets are (the number following the frequency indicates the strength): *Low(2) & Mid(2)*, *High(2) & Error(2, t=0.5)*, and *Small Low(2.5,p=dec) & High(1.5)*. An example image applied with *Small Low(2.5,p=dec) & High(1.5)* frequency noise is displayed in Figure 2(c). All experiments add, in total, noise with a strength of 4. The first two experiments have both frequency sets contributing equally to the image. In contrast, the third experiment shifts the focus to the small low set, which has a strength of 2.5 while the high frequency part has a strength of 1.5. Also, the weighting of the frequencies in the small low set in experiment three decreases based on the sum of the coordinates. The probability of selecting (0,0) is 0.65. Selecting a frequency for which the absolute value of the coordinates sum to 1 have a probability of 0.4 and 0.2 when the absolute sum is 2. This is also indicated with p=dec in Small Low(2.5,p=dec), which indicates the probability weighting that is decreasing for higher frequencies. Based on the work of [26, 34] the high frequency set is combined once with the small low and once with the error matrix based frequency set. The combinations are created to see whether the model generalizes well to high frequency and low frequency corruptions. It is also investigated, whether adding low frequency noise balances the low frequency sensitivity of the model trained on high frequency only.

4.1.3 K3. In the third category, *K3*, three frequencies are combined and applied to an image. The selected sets are: *Low(0.5) & High(2) & Error(2, t=0.5)*, *Low(0.5) & Mid(1) & High(2.5)*, and *Small Low(2.5,p=dec) & Mid(0.5) & High(1.5)*. The total strength of experiments one and three is 4.5, while the second experiment has a strength of 4. The strengths are assigned to frequency sets based on the influence each set has. In the first experiment, for example, the emphasis is on the high frequency range (2.5), followed by the mid (1.0) and low (0.5) frequency range. The small low set has the same probability weighting as the corresponding experiment in *K2*. An example image of the first experiment can be seen in Figure 2(d).

Table 1. Natural Accuracy, Mean Corruption Error, and Corruption Error of every experiment, where the experiments are defined by Frequency(strength, t=threshold of the error rate from the error matrix, p=probability weighting). SLow stands for Small Low in the experiment SLow(2.5,p=dec) & High(1.5). The frequencies from K3 experiments have been shortened from Small Low, Low, Mid and High Frequency to SL, L, M, and H. The test accuracy for natural images is measured in percentage (%). The abbreviated corruption types Gauss., Bright., Cont. and Pixel. refer to Gaussian, Brightness, Contrast and Pixelation. The **CE** results that are less than 100 indicate a lower error rate than the base model, while values above 100 indicate a higher and therefore worse error rate. The values that are bold indicate the best result from that column. In this case, Base is only given as a reference, and the best results are highlighted for experiments with data augmentation.

			Noise			Blur				Weather				Digital			
Experiments	Natur.(%)	**mCE**	Gauss.	Shot	Impulse	Defocus	Glass	Motion	Zoom	Snow	Frost	Fog	Bright.	Cont.	Elastic	Pixel.	JPEG
Base	92	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
Low(4)	**91**	142	117	127	109	157	147	155	183	148	158	**117**	112	**107**	151	177	166
Mid(4)	85	119	41	51	65	140	58	139	145	110	119	211	180	155	145	99	119
High(4)	90	**81**	42	46	63	**79**	**43**	**86**	**72**	86	79	142	125	116	**95**	**52**	83
Error(4, t=0.5)	88	98	36	44	59	114	61	124	119	89	96	153	141	142	119	78	90
Low(2) & Mid(2)	90	133	80	90	104	167	120	146	192	123	143	179	114	147	141	124	124
High(2) & Error(2, t=0.5)	89	94	**27**	**34**	51	103	51	107	100	**83**	84	211	132	170	117	64	**76**
SLow(2.5,p=dec) & High(1.5)	**91**	86	52	57	79	100	60	108	96	85	**67**	121	**103**	113	115	55	82
L(0.5) & H(2) & E(2, t=0.5)	88	102	28	36	**49**	114	53	117	111	94	95	226	148	174	130	71	83
L(0.5) & M(1) & H(2.5)	88	99	**27**	**34**	55	113	49	113	107	91	88	216	143	157	131	68	87
SL(2.5,p=dec)&M(0.5)&H(1.5)	90	91	40	45	74	108	60	112	103	86	69	147	116	130	125	61	84

The sensitivity to various noise types, such as Gaussian Noise or Impulse Noise, ranges across the whole frequency spectrum [36]. Therefore, the high, mid, and low frequency combination is selected to see whether the noise resistance can be substantially improved. The *Small Low(2.5,p=dec) & Mid(0.5) & High(1.5)* experiment is a combination of the small low and high frequency set from *K2* to investigate whether the addition of the mid-range has a positive effect on the general robustness.

4.2 Training

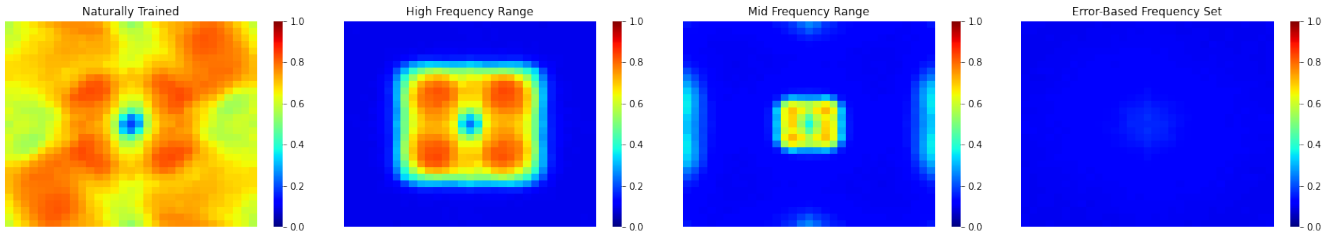
The training and testing procedures are implemented with PyTorch. Because of time limitations regarding the training procedure, the ResNet18 architecture [12] is used. Adam is employed as the optimizer with a learning rate of 0.0001 and a weight decay (L_2 regularization) of 1e-4 to prevent overfitting. The loss function utilized for the training is the Cross Entropy Loss. A scheduler reduces the learning rate by a factor of 0.2 every time the model stops improving. Every experiment consists of 100 epochs of training, with early stopping occurring after 30 epochs of no improvement in the validation loss. The training data of CIFAR-10 has a 90:10 split for the training and validation sets. Training and testing is performed on an Nvidia Tesla T4 and an Nvidia A10 GPU.

5 RESULTS

The results, together with the natural test set accuracy, are shown in Table 1. For better visualization, Figure 4 contains the average corruption error results for all four categories from the three experiments with the best **mCE** results. Training with high frequency noise has the best **mCE**. In contrast, training with low frequency noise has the highest error rates for almost all **CEs** and **mCE**. The **mCEs** of Low(4) and High(4) are 142 and 81, respectively, which is a significant difference. There is a general trend that experiments Low(4), Mid(4) and Low(2) & Mid(2) have the highest error rates of all corruption types, except for a few cases. In regard to the natural test accuracy, it is surprising that High(4) achieved 90% natural accuracy in comparison to the base model with 92%. Therefore, there is almost no tradeoff in natural test accuracy. The addition of the small low set to the high frequency set improved the accuracy by 1%. Overall, Mid(4) has the lowest natural test accuracy with 85%. In Figure 3 Fourier heatmaps have been plotted that show the error rates for each of the 31×31 frequencies. While the heatmap of High(4) shows low error rates for the range of frequencies that have been selected, Mid(4) has low error rates in high frequency areas that are not in the set. As expected, the Error(4, t=0.5) heatmap displays constant low error rates for every frequency range. The presentation of the results is separated into each corruption category, followed by the average corruption accuracy results based on the severity.

5.0.1 Noise. In the noise category, High(2) & Error(2, t=0.5) and the mix of all frequencies (low, mid, and high) achieved the lowest corruption error rate for Gaussian and Shot Noise. Concurrently, L(0.5) & H(2) & E(2,t=0.5) has the best error rate for Impulse noise. In general, Low(4) performed the worst, with each **CE** being significantly higher than all other experiments. In this particular case, Mid(4) performed better than Low(4). The two experiments that include the small low frequency set in K2 and K3 have slightly higher **CE** for Impulse noise than most of the other experiments. The average error rate for K1, K2 and K3 decreases from K1 to K3. **Overall, High(2) & Error(2, t=0.5) performed the best for all corruptions in the Noise category, with an average of 37.** This is closely followed by L(0.5) & H(2) & E(2,t=0.5) with an average of 38.

5.0.2 Blur. For the blur category, **High(4) has the best corruption error rates and outperformed the base model in all four**



(a) Natural (b) High Frequency (strength: 4) (c) Mid Frequency (strength: 4) (d) Error Based (strength: 4, t=0.5)

Fig. 3. Fourier Heat Maps (Section 3.3) from (a) base model, (b) High(4): adding high frequency noise, (c) Mid(4): adding mid frequency noise, and Error(4, t=0.5): adding noise based on the error matrix where any frequency is considered, where its error rate (displayed with color) is less than or equal to 0.5. The red color indicates a high error rate up to 1, and the blue color indicates a low error rate down to 0.

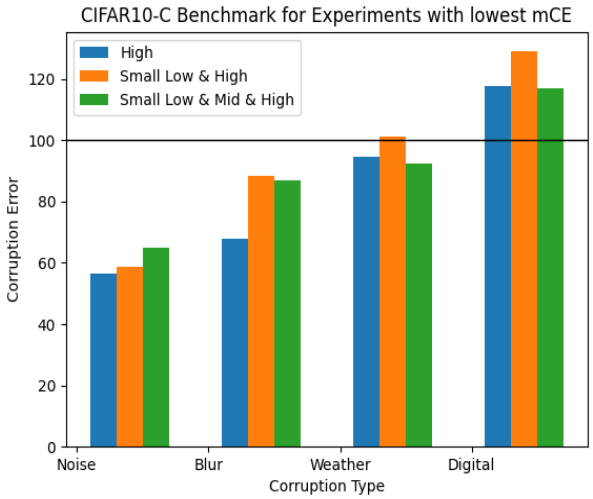


Fig. 4. The corruption error of the first three experiments with the lowest **mCE** values, averaged for each corruption type. The line at y=100 displays the base model reference value.

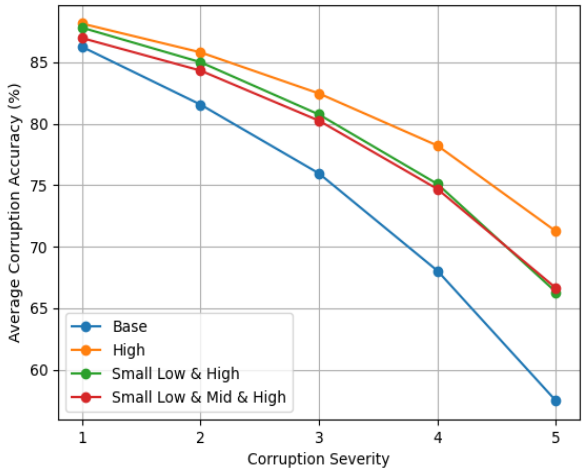


Fig. 5. Average CIFAR-10-C corruption accuracy of all 15 corruptions over all five severity levels for four experiments. The selection of the experiments is based on the lowest **mCE** results, similar to Figure 4.

blur types. Also, it was the only experiment that outperformed Base in Defocus and Motion blur. The improvement with respect to high frequency noise can also be observed in K2. For example, High(2) & Error(2, t=0.5) performed slightly better than other experiments, despite Error(2, t=0.5) having a high **CE** for all blur types. In contrast, Low(4) and Low(2) & Mid(2) are outperformed by Base and have very high **CEs**, again with a large difference from the rest. Mid(4) has similar high **CEs** for all blur types except for Glass, which is 58. In general, the **CEs** of Glass appear to be significantly better than all other blur types. Comparing High(2) & Error(2, t=0.5) and L(0.5) & H(2) & E(2, t=0.5) shows that by adding low frequency with low strength, the error rates of Defocus, Motion and Zoom increased by 10, while Glass blur increased slightly by 2.

5.0.3 Weather. In this category, there are two types of results. Snow and Frost have mostly enhanced robustness after applying Fourier-Basis noise, but Fog and Brightness did not see any improvement. **SLow(2.5,p=dec) & High(1.5) is on average performing the best in the Weather category.** In addition, combining the small low frequency set with other sets has a positive effect on the performance regarding Frost and Brightness. Low(4) performed the best in Fog with 117, which is still worse than Base. The good results of Base are also reflected, with the performance of L(0.5) & H(2) & E(2, t=0.5) being more than two times worse than base model. Also, middle frequencies seem to be damaging for robustness to Fog and overall not well performing in the category.

5.0.4 Digital. Similar to Weather, there are again two types of results. Fourier-Basis noise does not improve robustness against Contrast and Elastic corruptions. However, robustness against Pixelation and JPEG is substantially enhanced. Low(4) has the best **CE** in Contrast, while Error(4, t=0.5) and High(2) & Error(2, t=0.5) have a **CE** of 170 and 174. **High(4) has on average the lowest CE in the Digital category.** It also has the lowest **CE** in Elastic and

Pixelation. In general, Fog, Contrast and Brightness turned out to be very challenging for models with Fourier-Basis noise augmentation.

5.0.5 Corruption Severity. In Figure 5 the accuracy of the base model and the first three experiments with the best **mCE** are selected and plotted based on the five corruption severities. The base model shows a considerable decrease in accuracy from the second severity level on. The same happens for the other experiments, but to a much lesser degree. High(4) has, again, the best results and the highest accuracy in all five severity levels compared to the rest.

6 DISCUSSION

First, the method described in Section 3 has been successful at generating noise and applying it as a data augmentation method. As the following analysis will show, it improved the base model's robustness to common corruptions from the CIFAR-10-C dataset. The results in Table 1 and Figure 4 suggest a vast improvement in corruption robustness for the high frequency experiment. At the same time, the low frequency experiment and combinations with mid frequency showcase the worst performance overall. This can be explained by the results of the base model. Base has low error rates around the zero frequency, which can also be observed in Figure 3(a). As a result, the base model performed well on low frequency corruptions such as Fog, Contrast or Brightness. Consequently, Low(4) has relatively low **CE** in those corruptions, since it targets the extended area around the zero frequency. However, the **CEs** are still worse than the base model, which indicates that the corruptions are closely centered around the zero frequency. This is also confirmed by the experiment combining the small low frequency set with the high frequency set that results in similar values for Fog, Contrast, and Brightness.

The base model is very sensitive to high frequency noise, as indicated in Figure 3. Therefore, training it on high frequency noise improves the robustness against noise and blur corruptions. A reason could be that the model ignores the relative high frequency component of the image and could develop a bias towards the low frequency component. Hence, the low frequency corruptions would then perform worse, which explains the results obtained for Fog, Brightness and Contrast for High(4). There is a performance tradeoff for High(4) between Noise & Blur and Weather & Digital. With the addition of the small low set, this tradeoff is weakened. Although the combination improves robustness against Weather and Digital compared to High(4), the improvements in robustness against Blur are diminished. Thus, combinations with small low or other frequencies decrease the impact of the high frequency set on Blur. This has a considerable effect, since High(4) is the only experiment that performs better in Blur than the base model.

The heat maps and CIFAR-10-C results reveal an interesting link between common corruptions and Fourier-Basis noise. Figure 3(d) displays the heatmap from Error(4, t=0.5), which shows low error rates for all frequencies. Nonetheless, it still performs average in the benchmark. This could be explained by the impact some corruptions have on images. For example, when blur is applied to an image, high frequency information is to some extent removed from it. With our method, we can only add frequencies to images. Therefore, reducing the overall error rate of Fourier-Basis noise does not necessarily imply an equal reduction in the error rate of the corruptions in CIFAR-10-C.

As already found out by Yin et al. [36], different corruptions have different frequency distributions, which support the good results for Noise in K2 and K3. Ford et al. achieved improvement in robustness to Noise and Blur corruptions by using Gaussian noise data augmentation [7]. Since Gaussian noise is very similar to high frequency noise, it confirms the results we got with the high frequency set. Other research [34, 37] suggests that the baseline model is prone to high frequency noise. In addition, Saikia et al. [26] confirm that robust models prefer low frequency information, which explains the overall good performance of high frequency noise.

7 CONCLUSIONS AND FUTURE WORK

This paper utilized a new method for data augmentation using Fourier-Basis noise to investigate the robustness to common corruptions on the ResNet18 model [12]. The dataset that was used to train the model is the CIFAR-10 dataset. Several experiments have been designed that use different frequency sets to augment images. For each image, there were up to three different frequencies of noise that were applied. The corruption benchmark CIFAR-10-C was used to evaluate the effects on the robustness against 15 different corruption types, while heat maps were created to visually display the robustness against Fourier-Basis noise. The results indicate that high frequency noise improves the base model's corruption robustness considerably, while low frequency noise worsens the overall performance. The combination of different frequencies led to no further improvement. It was deduced that training on high frequency leads to a low frequency bias that shows more corruption robustness than the base model. The potential of this method can be quite strong considering its simplicity and the results it has achieved in the robustness against corruptions that can be comparable to state-of-the-art results.

Nonetheless, this was just the first step, and many other aspects can be explored. This research only provides a limited view on the application possibilities. Therefore, further research should use this method with various other datasets and models to be able to make better conclusions as to how comparable this method performs regarding other established methods. Especially with regard to common corruptions, more benchmarks, such as CIFAR-10-P [13] should be utilized to assess the performance of the method. Also, using real-world corruptions could give valuable insights on the practicality of potential applications, such as in medicine. This is especially important, considering that Abello et al. [1] found that frequency bias differs for every dataset. In this research, only ten experiments have been conducted. Therefore, there is still room for more detailed and better-informed frequency choices that can incorporate the findings from this study. Another interesting aspect that could turn out to be an advantage compared to other methods is the training time. Since the overall augmentation process is simple and the set of frequencies is predefined, it could be more efficient to use and improve overall training time.

REFERENCES

[1] Antonio A. Abello, Roberto Hirata, and Zhangyang Wang. 2021. Dissecting the High-Frequency Bias in Convolutional Neural Networks. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. 863–871. https://doi.org/10.1109/CVPRW53098.2021.00096 ISSN: 2160-7516.

[2] Aharon Azulay and Yair Weiss. 2018. Why do deep convolutional networks generalize so poorly to small image transformations? (2018). https://doi.org/10.48550/ARXIV.1805.12177 Publisher: arXiv Version Number: 4.

[3] Ekin D. Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V. Le. 2018. AutoAugment: Learning Augmentation Policies from Data. (2018). https://doi.org/10.48550/ARXIV.1805.09501 Publisher: arXiv Version Number: 3.

[4] Ekin D. Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V. Le. 2019. RandAugment: Practical automated data augmentation with a reduced search space. (2019). https://doi.org/10.48550/ARXIV.1909.13719 Publisher: arXiv Version Number: 2.

[5] Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song. 2017. Robust Physical-World Attacks on Deep Learning Models. (2017). https://doi.org/10.48550/ARXIV.1707.08945 Publisher: arXiv Version Number: 5.

[6] Samuel G. Finlayson, John D. Bowers, Joichi Ito, Jonathan L. Zittrain, Andrew L. Beam, and Isaac S. Kohane. 2019. Adversarial attacks on medical machine learning. *Science (New York, N.Y.)* 363, 6433 (March 2019), 1287–1289. https://doi.org/10.1126/science.aaw4399

[7] Nicolas Ford, Justin Gilmer, Nicholas Carlini, and Ekin Cubuk. 2019. Adversarial Examples Are a Natural Consequence of Test Error in Noise. In *Proceedings of the 36th International Conference on Machine Learning*. PMLR, 2280–2289. https://proceedings.mlr.press/v97/gilmer19a.html ISSN: 2640-3498.

[8] Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. 2016. Image Style Transfer Using Convolutional Neural Networks. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2414–2423. https://doi.org/10.1109/CVPR.2016.265 ISSN: 1063-6919.

[9] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. 2018. ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. (2018). https://doi.org/10.48550/ARXIV.1811.12231 Publisher: arXiv Version Number: 2.

[10] Robert Geirhos, Carlos R. Medina Temme, Jonas Rauber, Heiko H. Schütt, Matthias Bethge, and Felix A. Wichmann. 2018. Generalisation in humans and deep neural networks. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems (NIPS'18)*. Curran Associates Inc., Red Hook, NY, USA, 7549–7561.

[11] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. 2014. Explaining and Harnessing Adversarial Examples. (2014). https://doi.org/10.48550/ARXIV.1412.6572 Publisher: arXiv Version Number: 3.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 770–778. https://doi.org/10.1109/CVPR.2016.90 ISSN: 1063-6919.

[13] Daniel Hendrycks. 2019. CIFAR-10-C and CIFAR-10-P. (Jan. 2019). https://doi.org/10.5281/zenodo.2535967

[14] Dan Hendrycks and Thomas Dietterich. 2019. Benchmarking Neural Network Robustness to Common Corruptions and Perturbations. (2019). https://doi.org/10.48550/ARXIV.1903.12261 Publisher: arXiv Version Number: 1.

[15] Dan Hendrycks, Norman Mu, Ekin D. Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. 2019. AugMix: A Simple Data Processing Method to Improve Robustness and Uncertainty. (2019). https://doi.org/10.48550/ARXIV.1912.02781 Publisher: arXiv Version Number: 2.

[16] Daniel Ho, Eric Liang, Xi Chen, Ion Stoica, and Pieter Abbeel. 2019. Population Based Augmentation: Efficient Learning of Augmentation Policy Schedules. In *Proceedings of the 36th International Conference on Machine Learning*. PMLR, 2731–2741. https://proceedings.mlr.press/v97/ho19b.html ISSN: 2640-3498.

[17] Philip Jackson, Amir Atapour-Abarghouei, Stephen Bonner, Toby Breckon, and Boguslaw Obara. 2019. Style augmentation : data augmentation via style randomization.. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, Deep Vision, Long Beach, CA, USA, 16-20 June 2019 [Conference proceedings]*. DU, Long Beach, CA, USA. http://cvpr2019.thecvf.com/

[18] Matt Jordan, Naren Manoj, Surbhi Goel, and Alexandros G. Dimakis. 2019. Quantifying Perceptual Distortion of Adversarial Examples. (2019). https://doi.org/10.48550/ARXIV.1902.08265 Publisher: arXiv Version Number: 1.

[19] Alex Krizhevsky. 2009. *Learning multiple layers of features from tiny images*. Technical Report.

[20] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. 2017. ImageNet classification with deep convolutional neural networks. *Commun. ACM* 60, 6 (May 2017), 84–90. https://doi.org/10.1145/3065386

[21] A. Laugros, A. Caplier, and M. Ospici. 2019. Are adversarial robustness and common perturbation robustness independent attributes? 1045–1054. https://doi.org/10.1109/ICCVW.2019.00134

[22] Sungbin Lim, Ildoo Kim, Taesup Kim, Chiheon Kim, and Sungwoong Kim. 2019. Fast AutoAugment. (2019). https://doi.org/10.48550/ARXIV.1905.00397 Publisher: arXiv Version Number: 2.

[23] Raphael Gontijo Lopes, Dong Yin, Ben Poole, Justin Gilmer, and Ekin D. Cubuk. 2019. Improving Robustness Without Sacrificing Accuracy with Patch Gaussian Augmentation. (2019). https://doi.org/10.48550/ARXIV.1906.02611 Publisher: arXiv Version Number: 1.

[24] Luis Perez and Jason Wang. 2017. The Effectiveness of Data Augmentation in Image Classification using Deep Learning. (2017). https://doi.org/10.48550/ARXIV.1712.04621 Publisher: arXiv Version Number: 1.

[25] Evgenia Rusak, Lukas Schott, Roland S. Zimmermann, Julian Bitterwolf, Oliver Bringmann, Matthias Bethge, and Wieland Brendel. 2020. A Simple Way to Make Neural Networks Robust Against Diverse Image Corruptions. In *Computer Vision – ECCV 2020 (Lecture Notes in Computer Science)*, Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm (Eds.). Springer International Publishing, Cham, 53–69. https://doi.org/10.1007/978-3-030-58580-8_4

[26] Tonmoy Saikia, Cordelia Schmid, and Thomas Brox. 2021. Improving robustness against common corruptions with frequency biased models. (2021). https://doi.org/10.48550/ARXIV.2103.16241 Publisher: arXiv Version Number: 1.

[27] Yu Shen, Laura Zheng, Manli Shu, Weizi Li, Tom Goldstein, and Ming Lin. 2021. Gradient-Free Adversarial Training Against Image Corruption for Learning-based Steering. In *Advances in Neural Information Processing Systems*, Vol. 34. Curran Associates, Inc., 26250–26263. https://proceedings.neurips.cc/paper/2021/hash/dce8af15f064d1accb98887a21029b08-Abstract.html

[28] Connor Shorten and Taghi M. Khoshgoftaar. 2019. A survey on Image Data Augmentation for Deep Learning. *Journal of Big Data* 6, 1 (July 2019), 60. https://doi.org/10.1186/s40537-019-0197-0

[29] Ryan Soklaski, Michael Yee, and Theodoros Tsiligkaridis. 2022. Fourier-Based Augmentations for Improved Robustness and Uncertainty Calibration. (2022). https://doi.org/10.48550/ARXIV.2202.12412 Publisher: arXiv Version Number: 1.

[30] Chuwen Song. 2020. Assessing the Impact of Various Image Corruptions on the Recognition of Traffic Signs by Machine Learning. In *Proceedings of the 12th International Conference on Computer Modeling and Simulation (ICCMS '20)*. Association for Computing Machinery, New York, NY, USA, 47–50. https://doi.org/10.1145/3408066.3408071

[31] Yuchi Tian, Kexin Pei, Suman Jana, and Baishakhi Ray. 2018. DeepTest: automated testing of deep-neural-network-driven autonomous cars. In *Proceedings of the 40th International Conference on Software Engineering (ICSE '18)*. Association for Computing Machinery, New York, NY, USA, 303–314. https://doi.org/10.1145/3180155.3180220

[32] Toan Tran, Trung Pham, Gustavo Carneiro, Lyle Palmer, and Ian Reid. 2017. A Bayesian Data Augmentation Approach for Learning Deep Models. (2017). https://doi.org/10.48550/ARXIV.1710.10564 Publisher: arXiv Version Number: 1.

[33] Igor Vasiljevic, Ayan Chakrabarti, and Gregory Shakhnarovich. 2016. Examining the Impact of Blur on Recognition by Convolutional Networks. (2016). https://doi.org/10.48550/ARXIV.1611.05760 Publisher: arXiv Version Number: 2.

[34] Haohan Wang, Xindi Wu, Zeyi Huang, and Eric P. Xing. 2020. High-Frequency Component Helps Explain the Generalization of Convolutional Neural Networks. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 8681–8691. https://doi.org/10.1109/CVPR42600.2020.00871 ISSN: 2575-7075.

[35] Haotao Wang, Chaowei Xiao, Jean Kossaifi, Zhiding Yu, Anima Anandkumar, and Zhangyang Wang. 2021. AugMax: Adversarial Composition of Random Augmentations for Robust Training. (2021). https://doi.org/10.48550/ARXIV.2110.13771 Publisher: arXiv Version Number: 3.

[36] Dong Yin, Raphael Gontijo Lopes, Jonathon Shlens, Ekin D. Cubuk, and Justin Gilmer. 2019. A Fourier Perspective on Model Robustness in Computer Vision. (2019). https://doi.org/10.48550/ARXIV.1906.08988 Publisher: arXiv Version Number: 3.

[37] Z. Zhang, D. Meng, L. Zhang, W. Xiao, and W. Tian. 2022. The range of harmful frequency for DNN corruption robustness. *Neurocomputing* 481 (2022), 294–309. https://doi.org/10.1016/j.neucom.2022.01.087