Harnessing Edge Information for Improved Robustness in Vision Transformers

Yanxi Li, Chengbin Du, Chang Xu

School of Computer Science, University of Sydney, Australia
yali0722@uni.sydney.edu.au, chdu5632@uni.sydney.edu.au, c.xu@sydney.edu.au

Abstract

Deep Neural Networks (DNNs) have demonstrated remarkable accuracy in vision classification tasks. However, they exhibit vulnerability to additional noises known as adversarial attacks. Previous studies hypothesize that this vulnerability might stem from the fact that high-accuracy DNNs heavily rely on irrelevant and non-robust features, such as textures and the background. In this work, we reveal that edge information extracted from images can provide relevant and robust features related to shapes and the foreground. These features assist pretrained DNNs in achieving improved adversarial robustness without compromising their accuracy on clean images. A lightweight and plug-and-play **EdgeNet** is proposed, which can be seamlessly integrated into existing pretrained DNNs, including Vision Transformers, a recent family of state-of-the-art models for vision classification. Our EdgeNet can process edges derived from either clean nature images or noisy adversarial images, yielding robust features which can be injected into the intermediate layers of the frozen backbone DNNs. The cost of obtaining such edges using conventional edge detection algorithms (e.g., Canny edge detector) is marginal, and the cost of training the EdgeNet is equivalent to that of fine-tuning the backbone network with techniques such as Adapter.

Introduction

Deep Neural Networks (DNNs) have attracted significant attention for their impressive performance in vision classification tasks (LeCun et al. 1989, 1995; Krizhevsky, Sutskever, and Hinton 2012; He et al. 2016; Zagoruyko and Komodakis 2016), demonstrating exceptional accuracy. However, their vulnerability to adversarial attacks has been a subject of concern (Goodfellow, Shlens, and Szegedy 2014; Madry et al. 2017; Hendrycks et al. 2021b,a; Hendrycks and Dietterich 2019). Adversarial attacks targeting classification models involve introducing subtle perturbations into input data, leading to misclassification by the models.

Previous research (Geirhos et al. 2018; Li and Xu 2023) suggests that the vulnerability of high-accuracy DNNs to these attacks might be rooted in their heavy reliance on *irrelevant and non-robust features* such as textures and backgrounds. In contrast, robust DNNs should instead base their predictions on *relevant and robust features* that pertain to shapes and foreground elements within the images.

However, Tsipras et al. (2018) point out that these moderately correlated features, while robust, can adversely affect accurate predictions, making them both robust and non-predictive. Conversely, the key to improving natural accuracy lies in utilizing weakly correlated and non-robust features, which, despite lacking adversarial robustness, exhibit predictive capability. Therefore, improving the adversarial robustness of a DNN without compromising its natural accuracy is challenging.

Based on the aforementioned theorem and the existence of high-accuracy large-scale pretrained models, the enhancement of adversarial robustness in naturally pretrained models has emerged as a prominent subject. Recently, TORA-ViTs (Li and Xu 2023) leverage the capabilities of a fine-tuning technique known as Adapter (Houlsby et al. 2019), effectively enhancing adversarial robustness with an affordable training cost. However, a drawback is also obvious. Through the incorporation of a fusion module to balance predictive and robust features, their model requires tuning a hyper-parameter to manage a trade-off. In certain scenarios, natural accuracy is compromised to enhance robustness, and vice versa.

In this paper, we present an alternative approach wherein, rather than directly augmenting parameters to the backbone network, we introduce a mechanism for integrating specific information extracted from the original images into the intermediate layers of the backbone network. To be specific, our novel approach highlights the potential of edge information extracted from images. This edge information holds the capability to furnish relevant and robust features pertaining to shapes and foreground elements within the images. These features, when integrated, assist pretrained DNNs in achieving improved adversarial robustness without compromising their natural accuracy in classifying clean images.

To achieve this objective, we propose the incorporation of a side branch named **EdgeNet**. This lightweight, plug-and-play network can seamlessly integrate into existing pretrained deep models, including the state-of-the-art models such as Vision Transformers (ViTs) (Dosovitskiy et al. 2020). Our EdgeNet operates by processing edge information extracted from input images. This process yields a set of robust features that can be strategically injected into the intermediary layers of the frozen backbone DNNs. This augmentation empowers the network to boost its defenses against adversarial perturbations while sustaining its accuracy in recognizing unaltered clean images.

The building blocks feature a “sandwich” architecture, comprising two zero convolutions (Zhang and Agrawala 2023) at both the input and output, sandwiching a ViT block in the middle. These two zero convolutions selectively transmit relevant inputs to the intermediate block and inject relevant outputs into the pretrained backbone. Furthermore, the zero convolution at the output position ensures that the information injected into the backbone initiates from a zero point, thereby ensuring the stability and trainability of our method.

Our approach incurs minimal additional computational overhead, comparable to using Adapters for fine-tuning ViT. Firstly, obtaining edge information through conventional edge detection algorithms, such as the well-known Canny edge detector (Canny 1986), incurs only marginal computational costs compared to DNNs. Furthermore, our EdgeNet-ViT-B/16, which incorporates 4 new blocks, is composed of 119.9M parameters and involves 24.37G floating-point operations (FLOPs). In contrast, the TORA-ViT-B/16 model, relying on Adapters, consists of 111.2M parameters and requires 26.0G FLOPs (Li and Xu 2023). Our approach achieves reduced computational overhead while slightly increasing memory consumption. This affordability, combined with its effectiveness, positions EdgeNet as a compelling tool for enhancing DNN robustness in a resource-efficient manner.

Our experiments cover a wide range of robust benchmarks, including white-box and black-box adversarial attacks on ImageNet-1K, employing FGSM (Goodfellow, Shlens, and Szegedy 2014) and PGD (Madry et al. 2017). The robustness of our EdgeNet extends beyond adversarial attacks to encompass scenarios that involve natural adversarial examples in ImageNet-A (Hendrycks et al. 2021b), out-of-distribution data in ImageNet-R (Hendrycks and Dietterich 2019), and common corruptions in ImageNet-C (Hendrycks et al. 2021a). In particular, our EdgeNet demonstrates slightly superior or comparable performance to the most balanced configuration ($\lambda = 0.5$) of TORA-ViTs across clean ImageNet-1K and ImageNet-A/R/C datasets. Furthermore, it achieves significantly improved accuracy under FGSM and PGD attacks (69.8% compared to 54.7% and 48.8% compared to 38.0%, respectively). The results reveal that our EdgeNet effectively enhances the robustness of pretrained ViTs.

Related Works

Adversarial Robustness

FGSM (Goodfellow, Shlens, and Szegedy 2014) claims that the vulnerability of neural networks to adversarial attacks stems from their linear characteristics, rather than the previously assumed factors of nonlinearity and overfitting. In line with this understanding, the authors present a simple and efficient method to generate adversarial examples for adversarial training proposed based on such a perspective to reduce adversarial error. PGD (Madry et al. 2017) studies the adversarial robustness from the view of robust optimization. A first-order gradient-based method for iterative adversarial is proposed. This method utilizes PGD on the negative loss function as a universal “first-order adversary”, signifying the strongest attack utilizing this approach.

Robustness to Other Perturbations

More recently, perturbations beyond adversarial attacks are gaining increasing interests. ImageNet-A (Hendrycks et al. 2021b) considers natural adversarial examples, which place objects in unusual contexts or orientations. By using a simple adversarial filtration technique, the dataset ensures that real-world, unmodified examples transfer to various unseen models reliably, highlighting shared weaknesses in computer vision models. ImageNet-C (Hendrycks and Dietterich 2019) considers common corruptions, which applies a series of 19 common visual corruptions in 5 categories to images. This benchmark standardizes and expands on the topic of corruption robustness, aiming to show which classifiers are preferable in safety-critical applications. ImageNet-R (Hendrycks et al. 2021a) considers out-of-distribution data, which contains abstract or rendered versions of objects. The authors critically evaluate previously proposed methods for improving out-of-distribution robustness, revealing that larger models and artificial data augmentations can enhance real-world robustness. Contrary to some claims in prior work, the findings emphasize that these techniques are effective and that improvements in artificial robustness benchmarks can indeed transfer to real-world distribution shifts.

Robust ViTs

ViTs, or Vision Transformers, represent an emerging family of new architectures for vision models. Several empirical studies (Bhojanapalli et al. 2021; Mahmood, Mahmood, and Van Dijk 2021; Paul and Chen 2022) have discovered that ViTs exhibit robustness against various types of perturbations. To enhance the robustness of ViTs, the Robust Vision Transformer (RVT) (Mao et al. 2022) has been introduced, which restructures the building blocks of ViTs and introduces two plug-and-play methods: position-aware attention scaling and patch-wise augmentation. In contrast to this approach, pyramid adversarial training (PyramidAT) (Herrmann et al. 2022) does not alter the network architecture but instead devises pyramid attacks to create adversarial examples by disturbing the input image on multiple scales.

Extending Backbones for Robustness

Lately, there have been efforts aimed at enhancing the robustness of existing visual model backbones through improvements. Li et al. (2021) explore the adversarial robustness of convolutional neural networks (CNNs) from a Neural Architecture Search (NAS) perspective, identifying a critical vulnerability to adversarial attacks despite their remarkable performance in certain tasks. Recognizing the trade-off between standard accuracy and robustness, they propose "Neural Architecture Dilation" as a method to enhance the resilience of CNNs. This approach aims to improve the backbone CNNs' robustness without significantly compromising their accuracy. Li and Xu (2023) explore the vulnerability of deep neural networks (DNNs) to input perturbations, focusing on the trade-off between natural accuracy and robustness in Vision Transformers (ViTs). They find that despite inherent robustness to various perturbations, ViTs still exhibit a trade-off between accuracy and robustness. Therefore, they propose a "trade-off" between the robustness and accuracy of Vision Transformers (TORA-ViTs), aiming to efficiently transfer pre-trained ViT models to balance both accuracy and robustness.

Methodology

Firstly, we provide a brief overview of adversarial training as a preliminary. Next, we illustrate the integration of edge information into the backbone and introduce the architecture of building blocks in our EdgeNet. Lastly, we provide necessary details of the edge detection algorithm and establish our joint optimization objective.

Preliminary: Adversarial Training

The common method for achieving robustness of a DNN $\hat{y} = f(\boldsymbol{x})$ against adversarial attacks is adversarial training. This method involves formulating the training objective in a minimax form, wherein the goal is to minimize the loss to discover the optimal model while concurrently maximizing the loss to identify the optimal adversarial examples:

$$f^* := \operatorname*{argmin}_f \mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}} \left[\max_{\boldsymbol{x}'\in B_p(\boldsymbol{x},\varepsilon)} \ell\left(f(\boldsymbol{x}'),y\right)\right], \qquad (1)$$

where f^* is the robust model resulting from adversarial training, $\boldsymbol{x}$ and y represent images and labels sampled from a training distribution $\mathcal{D}$, and $B_p(\boldsymbol{x},\varepsilon) = \{\boldsymbol{x}' : \|\boldsymbol{x}-\boldsymbol{x}'\|_p \leq \varepsilon\}$ defines a ball covering all allowed adversarial examples $\boldsymbol{x}'$, with the clean image $\boldsymbol{x}$ as its center, the allowed magnitude of perturbation ε as its radius, and the l_p-norm serving as a measure of distance.

Integration of Edge Information

In Eq. 1, the model $f(\cdot)$ solely considers the images $\boldsymbol{x}$ (for clean examples) or $\boldsymbol{x}'$ (when subjected to an attack). We propose to integrate edge information into the model, enhancing the performance of the model:

$$\hat{\boldsymbol{y}} = f(\boldsymbol{x}, \boldsymbol{e}), \qquad (2)$$

where $\boldsymbol{e}$ is the edge obtained by $\boldsymbol{e} = \mathrm{Edge}(\boldsymbol{x})$, and $\mathrm{Edge}(\cdot)$ is an edge detection algorithm, such as the Canny edge detector.

We start with a backbone network, composed of L building blocks as expressed in the following equation:

$$f_b = \left\{ f_b^{(l)}, l = 1, \ldots, L \right\}, \qquad (3)$$

where each building block $f_b^{(l)} : \boldsymbol{h}_{l-1} \mapsto \boldsymbol{h}_l$ maps the previous layer's representation $\boldsymbol{h}_{l-1}$ to the subsequent one $\boldsymbol{h}_l$.

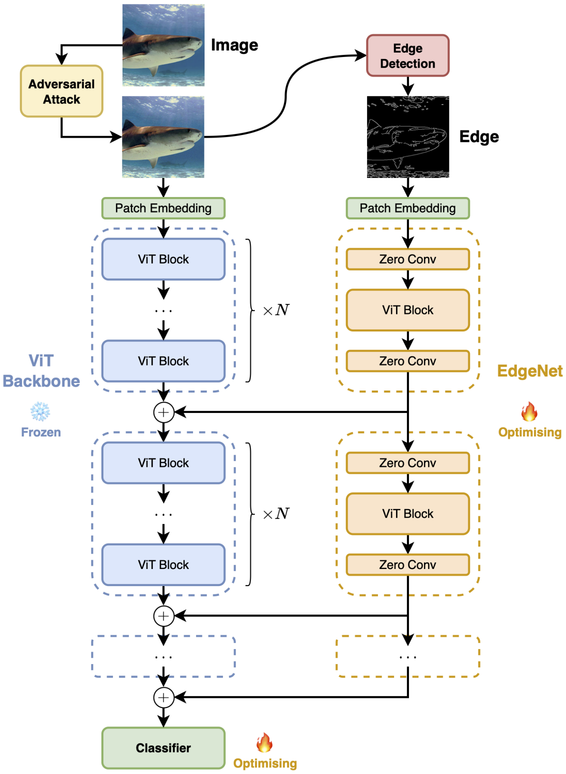

Figure 1: The architecture of our EdgeNet with ViT as the backbone. We employ an interval of N, signifying the addition of one EdgeNet block for every $N\times$ ViT blocks. Each EdgeNet block features a "sandwich" architecture, commencing with zero convolutions at both the input and output to initialize them with zeros. The output of each EdgeNet block is integrated into the intermediate layer of the ViT backbone through element-wise addition. Throughout the optimization process, the backbone remains frozen, while the EdgeNet and classification head undergoes training.

We enhance the backbone architecture by introducing an additional set of L_e blocks capable of processing edge information, which we refer to as **EdgeNet**:

$$f_e = \left\{ f_e^{(l)}, l \in 1, \ldots, L_e \right\}, \qquad (4)$$

where each building block $f_e^{(l)} : \boldsymbol{e}_{l-1} \mapsto \boldsymbol{e}_l$ are mappings similar to $f_b^{(l)}$ but deals with edge-related features $\boldsymbol{e}_l$.

In order to control the scale of the new blocks, we introduce a hyper-parameter N, which determines the insertion interval for incorporating these additional blocks. This is achieved through the relationship $L_e = L/N$. More specifically, when considering each building block indexed by $l = 1, \ldots, L$, we have

$$\begin{cases} \boldsymbol{h}_l' = \boldsymbol{h}_l + \boldsymbol{e}_{l/N} & \text{if } l \bmod N \text{ is } 0, \\ \boldsymbol{h}_l' = \boldsymbol{h}_l & \text{otherwise.} \end{cases} \qquad (5)$$

$\boldsymbol{h}_l'$ is then used as the input to the $l+1$ block $f_b^{(l+1)}$. Fig. 1 demonstrate the overall architecture of this framework.

EdgeNet Building Blocks

We implement a "sandwich" architecture for each building block in our EdgeNet framework, as depicted in Fig. 1. To be specific, we add zero convolutions $\mathcal{Z}(\cdot)$ (Zhang and Agrawala 2023) to both the input and output of each block. Nestled between the two zero convolutions, we place a ViT block $\mathcal{T}(\cdot)$ with randomized initialization, maintaining the same architecture to those found in the backbone:

$$\boldsymbol{e}_l = \mathcal{Z}^{(l)}_{\textbf{out}}\left(\mathcal{T}^{(l)}\left(\mathcal{Z}^{(l)}_{\textbf{in}}\left(\boldsymbol{e}_{l-1}\right)\right)\right). \tag{6}$$

Zero convolutions are defined as 1×1 convolution layer with both weight and bias initialized with zeros. Therefore, the input to the intermediate ViT block and the output of the EdgeNet building block are both start with zero.

Utilizing zero inputs, $\mathcal{Z}^{(l)}_{\textbf{in}}(\cdot)$ functions as a filter for extracting information related to the optimization objective. Employing zero outputs, $\mathcal{Z}^{(l)}_{\textbf{out}}(\cdot)$ functions as a filter for determining information to be integrated into the backbone. Furthermore, the addition of zeros to the backbone at the beginning ensures that the information flow within the backbone remains unaffected. Consequently, the subsequent fine-tuning of EdgeNet is significantly streamlined.

Edge Detection

We utilize the Canny edge detector (Canny 1986) for edge detection. Firstly, the image is processed with a Gaussian filter to reduce noise and smooth the intensity variations. Subsequently, the gradient magnitude and direction are computed using convolution with Sobel filters. The gradient direction helps determine the orientation of the edges. Non-maximum suppression is then applied to thin out the edges by retaining only the local maxima in the gradient magnitude along the gradient direction. Finally, a double thresholding step categorizes the edge pixels as strong, weak, or non-edges. Strong edges are retained, while weak edges are subjected to connectivity analysis to determine if they should be preserved.

Within the double thresholding phase, we employ the following equations to automatically determine the lower and upper thresholds:

$$\text{lower} = \max(0, 0.7 \times \text{median\_value}), \tag{7}$$

$$\text{upper} = \min(255, 1.3 \times \text{median\_value}), \tag{8}$$

where median_value is the median value of pixels obtained from the previous step.

Joint Optimization

During the training process, the pre-existing ViT blocks and the patch embedding layer within the backbone remain fixed, undergoing no updates. The optimization objective solely focuses on the new ViT blocks and patch embedding layer introduced for edge features, in addition to the classification head within the backbone.

Considering that our primary focus is not directed towards balancing the trade-off between accuracy and robustness, we adopt a simplified joint optimization objective:

$$\min_f \mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}\Big[\alpha \cdot \ell\left(f(\boldsymbol{x}, \text{Edge}(\boldsymbol{x})), y\right) + \beta \cdot \max_{\boldsymbol{x}' \in B_p(\boldsymbol{x},\varepsilon)} \ell\left(f(\boldsymbol{x}', \text{Edge}(\boldsymbol{x}')), y\right)\Big], \tag{9}$$

where α is the weight for accuracy and β is the weight for robustness. The cross-entropy loss is used for $\ell(\cdot,\cdot)$. Through the adjusting of α and β, we can fine-tune our EdgeNet in a manner that enhances its robustness, meanwhile ensuring that the accuracy won't drop significantly.

Experiments

Settings

Pretrained ViTs. In our experiments, we adopt the vanilla ViT architecture introduced by Dosovitskiy et al. (2020). In our specific approach, we employ the ViT-B/16 variant, which is characterized by several key parameters. This variant encompasses an input size of 224×224 pixels, with each image divided into patches of dimensions 16×16. The embedding dimension is set at 768, and the architecture is comprised of a total of 12 blocks. To initialize the network, we employ pretrained parameters made available by Steiner et al. (2021).

Training. For the joint training objective in Eq. 9, we set the hyper-parameters $\alpha = 1.2$ and $\beta = 0.8$. We use FGSM with l_∞-norm for adversarial training and adopt a perturbation magnitude of $\varepsilon = 1/255$. We use the SGD optimizer, with a fixed learning rate of 1×10^{-4}, a momentum of 0.9, and a weight decay of 2×10^{-5}.

Evaluations. Our evaluations cover 5 distinct settings.

1. We initiate our analysis by addressing ***white-box attacks***. To investigate the robustness of our model, we employ both single-step FGSM (Goodfellow, Shlens, and Szegedy 2014) and multi-step PGD (Madry et al. 2017) on the ImageNet-1K dataset. Consistent with Mao et al. (2022), we adopt a l_∞-norm and a perturbation magnitude of $\varepsilon = 1/255$ for both FGSM and PGD. For PGD, we execute it for 5 steps, using a step size of $0.5/255$.

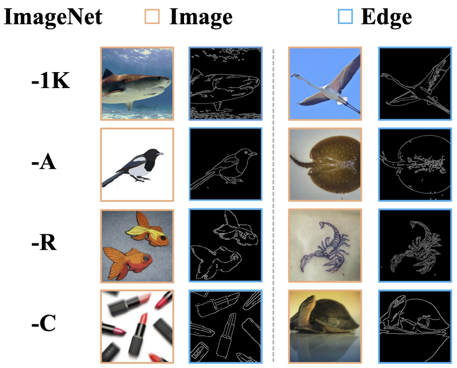

Figure 2: Instances selected from ImageNet-1K, -A, -R, and -C, accompanied by their respective edges extracted by the Canny edge detector.

# Intervals	# New Blocks	FLOPs (G)	Params (M)	Throughput (Images/Sec)	Clean	Attacks		ImageNet Variants		
						FGSM	PGD	A	R	C (↓)
1	12	37.88	186.14	375.16	83.4	69.0	48.0	39.5	56.8	**34.3**
3	4	24.37	119.99	543.40	**83.7**	**69.8**	**48.8**	**39.6**	56.9	34.4
6	2	21.00	103.45	601.64	83.3	66.8	46.3	37.6	**57.2**	35.0
-	0	17.60	88.1	635.81	80.2	41.1	15.5	22.1	42.0	56.9

Table 1: The performance of EdgeNet across varying scales. The "# Intervals" determines the frequency of adding a new block in relation to existing ones, while "# New Blocks" denotes the total number of added blocks. We also include results achieved by fine-tuning the classification head of the backbone for comparison (the last row).

2. Moving on, we delve into the realm of ***black-box attacks***. Initially, the ViT backbone is used to generate adversarial perturbations to attack our EdgeNet-ViT. Subsequently, we employ a ResNet-50 model to generate adversarial perturbations to attack both the ViT backbone and our EdgeNet-ViT.

Expanding the scope beyond adversarial attacks, we extend our evaluations to assess the robustness of our EdgeNet-ViT in broader scenarios.

3. In the domain of ***natural adversarial examples***, we use the ImageNet-A dataset (Hendrycks et al. 2021b). This dataset places the ImageNet objects in unusual contexts or orientations, challenging the model's adaptability to unconventional scenarios.
4. In the domain of ***out-of-distribution data***, we use the ImageNet-R dataset (Hendrycks et al. 2021a). This dataset contains abstract or rendered versions of objects, probing the model's capacity to generalize beyond its trained data distribution.
5. In the domain of ***common corruptions***, we use the ImageNet-C dataset (Hendrycks and Dietterich 2019), which applies 19 common corruptions categorized into 5 groups (e.g., motion blur, Gaussian noise, fog, JPEG compression, etc.), mimicking real-world distortions that a model might encounter.

Illustrations of samples sourced from ImageNet-1K, -A, -R, and -C, along with their corresponding edges extracted by the Canny edge detector, are presented in Fig. 2.

Different Scales of EdgeNet

As we introduce an interval hyper-parameter, we manipulate its value to adjust the scale of EdgeNet. We present the performance of EdgeNet across different scales on the aforementioned benchmarks, alongside reporting metrics such as the count of floating-point operations (FLOPs), the number of parameters, and the inference throughput (measured in images per second). We assess the throughput using a single NVIDIA RTX4090 GPU with 24GB of memory. As we maintain the backbone blocks in a frozen state and solely optimize our newly introduced blocks while fine-tuning the classification head, we establish a baseline by including the ViT-B/16 backbone. In this baseline, no new blocks are added, but the classification head is fine-tuned. The results are reported in Table 1.

When incorporating a total of 12 new blocks into the model, a substantial increase in computational overhead is observed. Additionally, the convergence of the model becomes challenging under these circumstances. While the inclusion of a larger number of new blocks results in improved performance compared to inserting only 2 new blocks, it falls short in performance when compared to the outcome of inserting 4 new blocks. In contrast, introducing 4 new blocks emerges as the most optimal configuration for EdgeNet, yielding its peak performance. This configuration does exhibit a slightly elevated computational overhead, yet it retains a commendable throughput, albeit slightly lower than the setup with only 2 new blocks (approximately 58.24 images/second lower). When incorporating a mere 2 new blocks, the achieved enhancement is not as pronounced as what is observed when inserting 12 or 4 new blocks. However, this configuration still outperforms the scenario of fine-tuning the classification head in isolation.

Taking into account both classification performance and computational considerations, we identify the configuration with # Intervals = 3 as the optimal setting. In this configuration, EdgeNet achieves significantly improved clean accuracy and robustness compared to the baseline, albeit at the expense of approximately 14.5% reduction in throughput. It strikes a balanced compromise between classification performance, computational requirements, and robustness. This configuration demonstrates substantial gains in clean accuracy and robustness over the baseline while maintaining a reasonable trade-off in terms of computational efficiency.

Comparison to SOTA Methods

Table 2 presents a comprehensive comparison between our proposed EdgeNet and 5 distinct categories of state-of-the-art (SOTA) methods. These categories encompass naturally trained and robust CNNs, naturally trained and robust ViTs, along with robust fine-tuned ViTs, evaluated across various benchmarks. The reported metrics include accuracy under adversarial attacks (FGSM and PGD), on ImageNet-A, and on ImageNet-R. Additionally, the mean Corruption Error (mCE) is reported for ImageNet-C, with lower values indicating better performace. As can be seen, our method showcases superior performance when subjected to both FGSM and PGD attacks. Meanwhile, our approach attains similar levels of performance on the clean ImageNet-1K dataset and its variants when compared to SOTA methods from previous

Categories	**Models**	**Clean**	**Attacks**		**ImageNet Variants**		
			FGSM	**PGD**	**A**	**R**	**C** (↓)
CNNs	ResNet-50 (He et al. 2016)	76.1	12.2	0.9	0.0	36.1	76.7
	ResNeXt50-32x4d (Xie et al. 2017)	79.8	34.7	13.5	10.7	41.5	64.7
	EfficientNet-B4 (Tan and Le 2019)	83.0	44.6	18.5	26.3	47.1	71.1
	ConvNeXt-B (Liu et al. 2022)	83.8	-	-	36.7	51.3	46.8
Robust CNNs	ANT (Rusak et al. 2020)	76.1	17.8	3.1	1.1	39.0	63.0
	AugMix (Hendrycks et al. 2019)	77.5	20.2	3.8	3.8	41.0	65.3
	Debiased CNN (Li et al. 2020)	76.9	20.4	5.5	3.5	40.8	67.5
	DeepAugment (Hendrycks et al. 2021a)	75.8	27.1	9.5	3.9	46.7	53.6
	Anti-Aliased CNN (Zhang 2019)	79.3	32.9	13.5	8.2	41.1	68.1
ViTs	ViT-B/16 (Dosovitskiy et al. 2020)	72.8	-	-	8.0	27.1	74.8
	ViT-B/16 + CutMix (Dosovitskiy et al. 2020)	75.5	-	-	14.8	28.5	64.1
	ViT-B/16 + MixUp (Dosovitskiy et al. 2020)	77.8	-	-	12.2	34.9	61.8
	ViT-B/16 + AugReg (Steiner et al. 2021)	79.9	-	-	17.5	38.2	52.5
	ViT-B/16-384 + AugReg (Steiner et al. 2021)	81.4	-	-	26.2	38.2	58.2
	PVT-Large (Wang et al. 2021)	81.7	33.1	7.3	26.6	42.7	59.8
	ConViT-B (d'Ascoli et al. 2021)	82.4	45.4	20.8	29.0	48.4	46.9
	DeiT-B/16 (Touvron et al. 2021)	82.0	46.4	21.3	27.4	44.9	48.5
	T2T-ViT$_t$-24 (Yuan et al. 2021)	82.6	46.7	17.5	28.9	47.9	48.0
	Swin-B (Liu et al. 2021)	83.4	49.2	21.3	35.8	46.6	54.4
	PiT-B (Heo et al. 2021)	82.4	49.3	23.7	33.9	43.7	48.2
Robust ViTs	PyramidAT (Herrmann et al. 2022)	81.7	-	-	23.0	47.7	45.0
	PyramidAT-384 (Herrmann et al. 2022)	83.3	-	-	36.4	46.7	47.8
	RVT-B (Mao et al. 2022)	82.5	52.3	27.4	27.7	48.2	47.3
	RVT-B* (Mao et al. 2022)	82.7	53.0	29.9	28.5	48.7	46.8
	MAE-ViT-B (He et al. 2022)	83.6	-	-	35.9	48.3	51.7
	FAN-L-ViT (Zhou et al. 2022)	83.9	-	-	34.2	53.1	43.3
Robust Fine-tuning	TORA-ViT-B/16 ($\lambda = 0.1$) (Li and Xu 2023)	84.1	48.4	23.3	46.5	57.6	31.7
	TORA-ViT-B/16 ($\lambda = 0.5$) (Li and Xu 2023)	83.7	54.7	38.0	39.2	56.3	34.4
	TORA-ViT-B/16 ($\lambda = 0.9$) (Li and Xu 2023)	80.3	74.2	57.5	22.2	53.7	41.6
	EdgeNet-ViT-B/16 **(Ours)**	83.7	69.8	48.8	39.6	56.9	34.4

Table 2: Evaluation of SOTA methods on ImageNet-1K and its variants (A, R and C). The top-1 accuracy is used to assess performance on clean ImageNet-1K, under adversarial attacks (FGSM and PGD), on ImageNet-A, and -R. In the case of ImageNet-C, the focus is on the mean Corruption Error (mCE), where lower values indicate better performance (marked by ↓). "ViT-B/16-384 + AugReg" and "PyramidAT-384" employ input dimensions of 384×384 inputs, while the remaining models utilize input dimensions of 224×224.

research.

We commence by comparing our EdgeNet with the robust fine-tuning method. When compared to the most balanced setting of TORA-ViT-B/16, indicated by $\lambda = 0.5$, we observe remarkable enhancements in accuracy under FGSM and PGD attacks, registering improvements of 15.1% and 10.8%, respectively. This performance augmentation is achieved while maintaining the same level of clean accuracy (83.7%). Furthermore, when considering ImageNet variants, our EdgeNet exhibits accuracy gains of 0.4% for ImageNet-A and 0.6% for ImageNet-R, while consistently preserving the identical mCE for ImageNet-C. When compared to TORA-ViT-B/16 with $\lambda = 0.1$, we have slightly lower clean accuracy (0.4%). This is because this model is fine-tuned for better performance on natural images. Therefore, our improvements in terms of adversarial robustness is even larger. We improve accuracy under FGSM and PGD attacks by 21.4% and 25.5%. We also have slightly lower performance on ImageNet variants, this is because they find their performance on ImageNet variants is correlated to clean accuracy instead of adversarial robustness.

In comparison to TORA-ViT-B/16 employing $\lambda = 0.1$, our clean accuracy exhibits a minor decrease of 0.4%. This diminishment can be attributed to the fact that this version of TORA has been fine-tuned for optimized performance on natural images. Consequently, our pronounced advancements in terms of adversarial robustness are even more notable. Under FGSM and PGD attacks, our approach displays substantial improvements, improving accuracy by 21.4% and 25.5%, respectively. Additionally, our performance is slightly lower than theirs when assessed on ImageNet variants. This can be attributed to the observation that their performance on ImageNet variants is closely associated with clean accuracy rather than adversarial robustness.

In the final setting of TORA-ViT-B/16, denoted by $\lambda = 0.9$, which is their most robust setting. Although their accuracy against FGSM and PGD attacks sees an increase of 4.4% and 8.7% respectively, this progress comes at the expense of a 3.4% reduction in clean accuracy. Additionally, in comparison to our approach, their performance on Ima-

Source Model	Defense Model	Valid Acc. (%)	
		FGSM	PGD
ViT-B/16	ViT-B/16	35.03	14.26
ViT-B/16	EdgeNet-ViT-B/16	74.41	70.32
ViT-S/16	ViT-B/16	74.09	75.59
ViT-S/16	EdgeNet-ViT-B/16	79.34	80.09
ViT-L/16	ViT-B/16	78.31	77.29
ViT-L/16	EdgeNet-ViT-B/16	80.62	80.18
Swin-B	ViT-B/16	82.94	82.40
Swin-B	EdgeNet-ViT-B/16	83.24	82.96

Table 3: The validation accuracy under black-box attacks on ImageNet-1K. Using ViT-B/16 as both source model and defense model is equivalent to a white-box attack, included here solely for the purpose of comparison.

geNet variants experiences relative drops of 17.4%, 3.2%, and 7.2%. Finally, we would like to emphasize once again that TORA controls a trade-off by introducing a specialized module into the backbone network to control the balance between robust features and predictive features. In contrast, our method aims to enhance robustness by introducing edge information without altering the backbone network itself. Therefore, in a fair comparison against their most balanced setting ($\lambda = 0.5$), our improvements are even more significant. However, even when compared to their favorably biased models, it is evident that our performance gap in their advantageous metrics is minimal, while our enhancements are more pronounced in their weaker aspects. In summary, our approach represents a more comprehensive, unbiased, and balanced model.

In addition to the robust fine-tuning, our EdgeNet outperforms all the other previous approaches under adversarial attacks and on the ImageNet variants. In terms of clean performance, our performance is only slightly lower than ConvNext-B4 and FAN-L-ViT for 0.1% and 0.2%, respectively. These differences are very marginal. Furthermore, our clean performance surpasses that of other previous methods.

Black-box Attacks

In the previous experiments, white-box attacks are investigated, involving scenarios where the attacker possesses access to the parameters of target models. In Table 3, we extend our analysis to a more realistic black-box attack scenario, where the assumption is made that the attacker lacks access to the parameters of the target models. We consider various models as the source model for generating adversarial perturbations. These models encompass the backbone ViT-B/16, as well as two of its size variants, namely ViT-S/16 (a smaller version) and ViT-L/16 (a larger version). Furthermore, we include another Vision Transformer architecture known as Swin-B in our considerations.

Initially, we consider attacks using ViT-B/16, the backbone itself, as the source model. The results show that when EdgeNet is incorporated as an additional component, attacks originating from the backbone no longer successfully compromise our model, increasing the classification accuracy from 35.03% to 74.41% under FGSM and from 14.26% to 70.32% under PGD respectively.

Input	Clean	Attacks		ImageNet Variants		
		FGSM	PGD	A	R	C (↓)
Image	82.7	64.4	47.0	32.2	56.1	37.2
Edge	83.7	69.8	48.8	39.6	56.9	34.4

Table 4: The performance of integrating image or edge information into the backbone.

When utilizing other models as the source model, it becomes evident that our EdgeNet demonstrates effective defense against these attacks, showcasing stronger robustness compared to the ViT-B/16 backbone itself. Furthermore, it is noteworthy that even when employing the Swin-B with a different architecture as the source model, both the ViT-B/16 backbone and our method exhibit substantial robustness. However, even in this scenario, our approach manages to further enhance the backbone's robustness.

Integrating Image or Edge Information

In order to illustrate the effectiveness of incorporating edge information, we conduct an experiment by replacing the inputs to EdgeNet with images. For this configuration, we maintain the exact same architecture and hyper-parameters for the new blocks, opting for the optimal # Intervals = 3 setting. As shown in Table 4, both the integration of images and edge information yield performance improvements compared to the classification head fine-tuning method presented in Table 1. Furthermore, it is noteworthy that the integration of edge information consistently outperforms the integration of image information. This is because integrating image information again may have redundancy in relation to the image features already present within the backbone.

Conclusion

In this work, we have uncovered a significant pathway to enhance the robustness of Deep Neural Networks, specifically Vision Transformers, against adversarial attacks. By leveraging edge information extracted from images, we developed EdgeNet, a lightweight and seamlessly integrable module that brings about improved adversarial robustness. The efficiency of EdgeNet, demonstrated through minimal additional computational overhead and wide applicability across various robust benchmarks, makes it a compelling advancement in the field. The experiment results, including superior performance against different types of adversarial attacks and maintained accuracy on clean images, underline the potential of edge information as a robust and relevant feature in vision classification tasks. Notably, the robustness of EdgeNet extends beyond adversarial attacks to scenarios involving natural adversarial examples (ImageNet-A), out-of-distribution data (ImageNet-R), and common corruptions (ImageNet-C). This broader application underlines EdgeNet's versatility and its potential as a comprehensive solution for diverse challenges in vision classification tasks.

Acknowledgements

This work was supported in part by the Australian Research Council under Projects DP210101859 and FT230100549.

References

Bhojanapalli, S.; Chakrabarti, A.; Glasner, D.; Li, D.; Unterthiner, T.; and Veit, A. 2021. Understanding robustness of transformers for image classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 10231–10241.

Canny, J. 1986. A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence*, 679–698.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

d'Ascoli, S.; Touvron, H.; Leavitt, M. L.; Morcos, A. S.; Biroli, G.; and Sagun, L. 2021. Convit: Improving vision transformers with soft convolutional inductive biases. In *International Conference on Machine Learning*, 2286–2296. PMLR.

Geirhos, R.; Rubisch, P.; Michaelis, C.; Bethge, M.; Wichmann, F. A.; and Brendel, W. 2018. ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. *arXiv preprint arXiv:1811.12231*.

Goodfellow, I. J.; Shlens, J.; and Szegedy, C. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

He, K.; Chen, X.; Xie, S.; Li, Y.; Dollár, P.; and Girshick, R. 2022. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 16000–16009.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Hendrycks, D.; Basart, S.; Mu, N.; Kadavath, S.; Wang, F.; Dorundo, E.; Desai, R.; Zhu, T.; Parajuli, S.; Guo, M.; et al. 2021a. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 8340–8349.

Hendrycks, D.; and Dietterich, T. 2019. Benchmarking Neural Network Robustness to Common Corruptions and Perturbations. *Proceedings of the International Conference on Learning Representations*.

Hendrycks, D.; Mu, N.; Cubuk, E. D.; Zoph, B.; Gilmer, J.; and Lakshminarayanan, B. 2019. Augmix: A simple data processing method to improve robustness and uncertainty. *arXiv preprint arXiv:1912.02781*.

Hendrycks, D.; Zhao, K.; Basart, S.; Steinhardt, J.; and Song, D. 2021b. Natural Adversarial Examples. *CVPR*.

Heo, B.; Yun, S.; Han, D.; Chun, S.; Choe, J.; and Oh, S. J. 2021. Rethinking spatial dimensions of vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 11936–11945.

Herrmann, C.; Sargent, K.; Jiang, L.; Zabih, R.; Chang, H.; Liu, C.; Krishnan, D.; and Sun, D. 2022. Pyramid adversarial training improves vit performance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 13419–13429.

Houlsby, N.; Giurgiu, A.; Jastrzebski, S.; Morrone, B.; De Laroussilhe, Q.; Gesmundo, A.; Attariyan, M.; and Gelly, S. 2019. Parameter-efficient transfer learning for NLP. In *International Conference on Machine Learning*, 2790–2799. PMLR.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25: 1097–1105.

LeCun, Y.; Boser, B.; Denker, J. S.; Henderson, D.; Howard, R. E.; Hubbard, W.; and Jackel, L. D. 1989. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4): 541–551.

LeCun, Y.; Jackel, L.; Bottou, L.; Cortes, C.; Denker, J. S.; Drucker, H.; Guyon, I.; Muller, U. A.; Sackinger, E.; Simard, P.; et al. 1995. Learning algorithms for classification: A comparison on handwritten digit recognition. *Neural networks: the statistical mechanics perspective*, 261: 276.

Li, Y.; and Xu, C. 2023. Trade-Off Between Robustness and Accuracy of Vision Transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7558–7568.

Li, Y.; Yang, Z.; Wang, Y.; and Xu, C. 2021. Neural architecture dilation for adversarial robustness. *Advances in Neural Information Processing Systems*, 34: 29578–29589.

Li, Y.; Yu, Q.; Tan, M.; Mei, J.; Tang, P.; Shen, W.; Yuille, A.; and Xie, C. 2020. Shape-texture debiased neural network training. *arXiv preprint arXiv:2010.05981*.

Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; and Guo, B. 2021. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 10012–10022.

Liu, Z.; Mao, H.; Wu, C.-Y.; Feichtenhofer, C.; Darrell, T.; and Xie, S. 2022. A convnet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11976–11986.

Madry, A.; Makelov, A.; Schmidt, L.; Tsipras, D.; and Vladu, A. 2017. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*.

Mahmood, K.; Mahmood, R.; and Van Dijk, M. 2021. On the robustness of vision transformers to adversarial examples. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 7838–7847.

Mao, X.; Qi, G.; Chen, Y.; Li, X.; Duan, R.; Ye, S.; He, Y.; and Xue, H. 2022. Towards robust vision transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12042–12051.

Paul, S.; and Chen, P.-Y. 2022. Vision transformers are robust learners. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 2071–2081.

Rusak, E.; Schott, L.; Zimmermann, R. S.; Bitterwolf, J.; Bringmann, O.; Bethge, M.; and Brendel, W. 2020. A simple way to make neural networks robust against diverse image corruptions. In *European Conference on Computer Vision*, 53–69. Springer.

Steiner, A.; Kolesnikov, A.; Zhai, X.; Wightman, R.; Uszkoreit, J.; and Beyer, L. 2021. How to train your vit? data, augmentation, and regularization in vision transformers. *arXiv preprint arXiv:2106.10270*.

Tan, M.; and Le, Q. 2019. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, 6105–6114. PMLR.

Touvron, H.; Cord, M.; Douze, M.; Massa, F.; Sablayrolles, A.; and Jégou, H. 2021. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, 10347–10357. PMLR.

Tsipras, D.; Santurkar, S.; Engstrom, L.; Turner, A.; and Madry, A. 2018. Robustness may be at odds with accuracy. *arXiv preprint arXiv:1805.12152*.

Wang, W.; Xie, E.; Li, X.; Fan, D.-P.; Song, K.; Liang, D.; Lu, T.; Luo, P.; and Shao, L. 2021. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 568–578.

Xie, S.; Girshick, R.; Dollár, P.; Tu, Z.; and He, K. 2017. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1492–1500.

Yuan, L.; Chen, Y.; Wang, T.; Yu, W.; Shi, Y.; Jiang, Z.-H.; Tay, F. E.; Feng, J.; and Yan, S. 2021. Tokens-to-token vit: Training vision transformers from scratch on imagenet. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 558–567.

Zagoruyko, S.; and Komodakis, N. 2016. Wide residual networks. *arXiv preprint arXiv:1605.07146*.

Zhang, L.; and Agrawala, M. 2023. Adding conditional control to text-to-image diffusion models. *arXiv preprint arXiv:2302.05543*.

Zhang, R. 2019. Making convolutional networks shift-invariant again. In *International conference on machine learning*, 7324–7334. PMLR.

Zhou, D.; Yu, Z.; Xie, E.; Xiao, C.; Anandkumar, A.; Feng, J.; and Alvarez, J. M. 2022. Understanding the robustness in vision transformers. In *International Conference on Machine Learning*, 27378–27394. PMLR.