ASK-ViT: A Model with Improved ViT Robustness through Incorporating SK Modules Using Adversarial Training

Youkang Chang *, Hong Zhao and Weijie Wang

School of Computer and Communications, Lanzhou University of Technology, Lanzhou 730050, China
* Correspondence: changyk@lut.edu.cn

Abstract: The ViTs model has been widely used since it was proposed, and its performance on large-scale datasets has surpassed that of CNN models. In order to deploy the ViTs model safely in practical application scenarios, its robustness needs to be investigated. There are few studies on the robustness of ViT model, therefore, this study investigates the robustness of the ViT model in the face of adversarial example attacks, and proposes the ASK-ViT model with improving robustness by introducing the SK module. The SK module consists of three steps, Split, Fuse and Select, which adaptively select the size of the perceptual field according to the input of multi-scale information, and extracts the features that help the model to classify examples. In addition, adversarial training is used in the training process. Experimental results show that the accuracy of the proposed defense method against C&W, DI2FGSM, MDI2FGSM, ImageNet-A, and ImageNet-R attacks is 30.104%, 29.146%, 28.885%, 29.573%, and 48.867%, respectively. Compared with other methods, the method in this study shows stronger robustness.

Keywords: deep neural network; robustness; ViT; SK module; adversarial training

1. Introduction

Following the convolutional neural networks (CNN) with wide applications in many fields [1–4], the Transformer [5] model has become a popular research topic for many scholars, thanks to the multi-headed self-attentive mechanism in the Transformer model, which can effectively extract features from multiple different locations in the input data. Several experimental results show that this mechanism can better extract global features. Later, Dosovitskiy et al. [6] used the Transformer structure for computer vision (CV) tasks and proposed the Vision Transformer (ViT) model. Since being proposed, the ViT model has been widely used in image semantic segmentation [7], target recognition [8], medical image detection [9], video super-resolution [10] and other fields, and it has shown good performance. Considering the wide application of ViT models in several fields, in order to deploy them in real life, the security of ViT models becomes particularly important, especially for the perturbation of small, counterattacks that are not easily observed by human vision. Therefore, it is necessary to investigate the robustness of ViT models.

The robustness of the ViT model has been extensively and thoroughly investigated for the impact of adversarial attacks. Benz et al. [11] compared the ViT model, the CNN model and the MLP-Mixer model, where the MLP-Mixer model uses a Multi-Layer perceptron (MLP) instead of a self-attentive module. They pointed out that the CNN model is able to extract local features better than the ViT model, which extracts global features but is more vulnerable to adversarial example attacks. In addition, the MLP-Mixer model is highly vulnerable to universal adversarial perturbation (UAP) attacks. Bhojanapalli et al. [12] tested the robustness of the ViT model on different datasets and compared it with the ResNet baseline model and found that the ViT model can be as robust as the ResNet baseline when pre-trained on larger datasets, and that attention blocks appear redundant for the ViT model. They noted that any single block in the model can be removed except...
for the first attention block without significantly degrading the robustness of the model. Fu et al. [13] proposed the Patch-Fool attack method to attack ViT models using attention-aware optimization techniques, and this method makes ViT models more vulnerable to the attack of adversarial examples than CNN models. Shao et al. [14] showed that the robustness of neural network models can be improved when the network structure is composed of Transformer modules and CNN modules; for a Transformer-only structure, however, simply increasing the Transformer module size or adding more Transformer layers does not guarantee the improvement of ViT model robustness. They point out that adversarial training can also effectively improve the robustness of ViT models.

Based on the above-mentioned research for the robustness of ViT models, this work proposes the ASK-ViT defense mechanism to enhance the robustness of the ViT model. Specifically, the SK module is introduced in the Attention mechanism of the ViT model. The SK module consists of three steps, Split, Fuse and Select. Firstly, the Split step has multiple network paths with different-sized convolutional kernels corresponding to different-sized perceptual fields of neurons; secondly, the Fuse operation fuses and aggregates the information of multiple network paths to obtain global representation weights; finally, the Select operation aggregates the feature maps of different-sized convolutional kernels based on the weight information. The SK module adaptively selects the size of the perceptual field based on the input multi-scale information, then uses a dataset consisting of clean and adversarial examples to train the model adversarially to enhance the robustness of the ViT model. The main work of this paper is as follows:

- Proposing an ASK-ViT model that adaptively adjusts the size of the perceptual field, extracts the multi-scale spatial information of the features and enhances the robustness of the model in the face of different adversarial attacks.
- Adversarial training of the ASK-RVT model using different adversarial examples, which can effectively defend against white-box and black-box attacks.
- By comparing with other ViT models, the proposed method exhibits strong robustness.

This paper is organized as follows: The second section reviews related work on commonly used defense methods, ViT models, and research on robustness against ViT models. The third section describes the proposed ASK-ViT defense method. The fourth section shows the experimental design as well as the analysis of the results. Finally, the full paper is summarized in the fifth section.

2. Related Work

This section will first introduce the defense method against attacks, then introduce the ViT model, and finally introduce the study carried out by the researchers on the robustness of the ViT model for attacks against examples.

2.1. Defense against Attacks

Adversarial attacks bring enormous security risks for the security deployment of various applications. In response to this problem, many scholars have proposed different methods for defense, and the defense methods at this stage are mainly divided into three categories.

- Preprocessing the adversarial examples before they are fed into the neural network.
- Adding more network layers or adding sub-networks to the network model to enhance the robustness of the neural network.
- Using other networks as additional networks to detect adversarial examples.

**Data pre-processing:** Before feeding the adversarial examples into the neural network, the adversarial examples are processed by data pre-processing methods to eliminate the effects of adversarial perturbations. This method is applicable to various types of adversarial attacks and enhances the robustness of the network model without affecting the classification accuracy of the network model for clean examples. The main methods are image denoising [15], data compression [16] and pixel shifting [17].
The data preprocessing method does not require the modification of the structure of the deep neural network and is fast in computation, but it causes a loss of high-frequency information when preprocessing the input data, which leads to the network extracting incorrect features and eventually outputting the wrong prediction results. Therefore, when using preprocessing methods for the defense of the adversarial examples, it should be remembered that the high-frequency information is not lost.

**Enhanced robustness of the neural network:** Unlike the pre-training processing method, this method modifies the structure of the neural network to enhance the robustness of the model.

Adversarial training [18], as one of the most effective defense methods, is used to train the neural network by adding the adversarial examples generated by the attack method to the clean examples and training them together as the training set. Papernot et al. [19] proposed a defense distillation method based on the distillation method by first training the original network at a distillation temperature of T to obtain the prior knowledge, and then adding the a priori knowledge into the distillation network to continue the training and obtain the final output. This method can effectively defend against white-box attacks FGSM and JSMA, but its defense performance is low for black-box attacks.

The robustness of neural networks can be effectively enhanced by improving their randomness and cognitive properties, but when faced with new examples, the neural networks need to be retrained, and the defense is less effective in the face of specific adversarial attack methods.

**Detecting adversarial examples:** Other neural networks are used to detect adversarial examples, distinguishing clean examples from adversarial examples by a threshold strategy, and feeding the data directly into the neural network if it is a clean example, otherwise, using an adversarial attack defense method to mitigate the impact caused by adversarial examples. For example, Samangouei et al. [20] proposed a defense method based on the Generative Adversarial Network (GAN) network, but its effect on improving the robustness of neural networks is not very obvious. Meng et al. [21] proposed a defense method based on MagNet network, which can better defend against black-box attacks, but cannot defend against white-box attacks. Nesti et al. [22] proposed a defense method based on Defense Perturbation, which is simple and computationally less expensive and does not need to consider complex models.

### 2.2. Vision Transformer Model

The ViT model achieves excellent performance on larger datasets, such as ImageNet-21K and JFT-300M, and consists of three main components: Linear Projection, Transformer Encoder block, and Multilayer Perceptron (MLP) layer. The image is first segmented into \( P \times P \)-sized patched sequence blocks, then the sequence blocks are mapped into one-dimensional vectors by Linear Projection, and the [CLS] token and Position Embedding are inserted and sent to the Transformer Encoder block together. In the Encoder block, the Multilayer Self Attention Mechanism (MHSA) is used to jointly learn the information from different regions in the Encoder block. Finally, MLP is used for image classification.

Based on the ViT model, researchers have proposed a series of variants of the ViT model to improve the performance of vision tasks. The main approaches include enhanced localization, improved relevance of self-attentive mechanisms, and network structure design.

Liu et al. [23] proposed the Shifted windows (Swin) Transformer structure to generate a hierarchical feature map by merging deep image blocks and calculating the self-attention in each non-overlapping local window, while allowing cross-window connections for efficiency. This layered structure is flexible and can be modeled at multiple scales, and its computational complexity is linearly related to the input image size. Huang et al. [24] proposed the Shuffle Transformer network model and investigated spatial shuffle as an efficient way to connect different windows, as they considered cross-window connectivity to be a key factor for improving the neural network model representation capability. The ViT model is able to capture the global dependencies of the input data but lacks the ability...
to extract local information. Graham et al. [25] applied convolutional neural networks to Transformer and proposed the LeViT network structure while introducing attention bias to integrate the location information of the input data. Srinivas et al. [26] modified the last three bottleneck blocks of ResNet and used global self-attention to replace the spatial convolution to propose the BoTNet network model. There was a significant performance improvement in both instance segmentation and target detection.

2.3. Study of Vision Transformer Model Robustness

ViT models have a wide range of applications in several fields, and the robustness of ViT models needs to be investigated in order to deploy them safely in real life.

Kim et al. [27] focused on the difference between the structural robustness of CNN and ViT models, and pointed out that Positional embedding in Patch embedding does not work properly when the color scale of the data changes. For this reason, they proposed an improved Patch embedding structure, called PreLayerNorm, to ensure the dimensional invariance of the ViT model. Experimental results show that the ViT structure with PreLayerNorm shows strong robustness in a wide range of corrupted data. Bai et al. [28] found that the reason why ViT models are more difficult to train compared to CNN networks is that ViT models are not as good as CNN models in capturing high-frequency information in images. To compensate for the shortcomings of ViT models, they proposed the HAT method, which directly enhances the high-frequency component of the image by adversarial training. Experimental results show that HAT can improve the performance of the ViT model and its variants. Heo et al. [29] investigated the role of spatial dimensional transformation and its effectiveness on Transformer-based structures, and pointed out that increasing the size of channels and reducing the spatial dimension helps the ViT model to improve robustness. Therefore, they proposed the Pooling-based Vision Transformer (PiT) model based on the ViT model. Experimental results show that PiT exhibits better generalization and better robustness in image classification than the baseline model compared to ViT. Popovic et al. [30] introduced a token-consistent stochastic layer in a multilayer perceptron to improve the adversarial robustness and functional privacy of the network model without changing the Transformer structure.

For the robustness of ViT models, many scholars have conducted in-depth research, focusing on the differences between the robustness of ViT models and CNN models and the characteristics of the ViT model structure. In this paper, the SK module is introduced into the ViT model, and the ViT model is further trained using the adversarial training defense method to enhance the robustness of the ViT model.

3. SK-Vision Transformer

The proposed network model is shown in Figure 1. The SE module is introduced in Embed-to-token and the SK module is introduced in ViT module. The model structure is shown in Figure 2a,b.

![Figure 1. Network diagram of SK-ViT model.](image-url)
that, the tokens are input into the Pooling layer to fuse features in different domains into new features while reducing the time complexity. After that, the tokens are fed into the ASK-ViT module, which can capture the global features by using the attention mechanism and adaptively adjust the perceptual field according to the feature information at different scales by using the features of the SK module. After that, the tokens are input into the Pooling layer to fuse features in different domains into new features while reducing the time complexity. After that, the tokens are fed into the Global average pooling layer to finally output the correct classification results.

3.1. Selective Kernel Module

In order to enable neurons to adaptively adjust the size of the receptive field according to different input information when extracting features, the Selective Kernel (SK) module is introduced in feature extraction. As a dynamic selection mechanism, the SK module has multiple branches with convolutional kernels of different sizes, and the feature information in each branch determines the softmax weights, and, guided by this information, fusion is performed using softmax attention. Figure 3 shows the SK module with a two-branch structure.

**Figure 2.** Embed to Token and SK-ViT modules.

The model adopts a multi-level hierarchy. Firstly, the input data is partitioned into patches of the same size and each patch is fed into the Embed-to-token module, which is shown in Figure 2a, through the SE-Module, which is used to highlight important features, then through the Convolutional Token Embedding layer, then each patch is flattened into a one-dimensional vector through the Convolutional Token Embedding layer. The tokens are then fed into the ASK-ViT module, which can capture the global features by using the attention mechanism and adaptively adjust the perceptual field according to the feature information at different scales by using the features of the SK module. After that, the tokens are input into the Pooling layer to fuse features in different domains into new features while reducing the time complexity. After that, the tokens are fed into the Global average pooling layer to finally output the correct classification results.

**Figure 3.** SK module structure.

**Split:** The input feature map \( X \in \mathbb{R}^{H \times W \times C} \) is first convolved by group convolution, and the size of the convolution kernels used is \( 3 \times 3 \) and \( 5 \times 5 \) to obtain two feature maps of the same size \( \hat{F}: X \rightarrow \hat{U} \in \mathbb{R}^{H \times W \times C} \) and \( \hat{F}: X \rightarrow \hat{U} \in \mathbb{R}^{H \times W \times C} \). Then the two sets of feature maps are subjected to batch normalization and ReLU function operations.

**Fuse:** Multiple branches are generated in the Split stage, and these branches carry information of different sizes to the next layer of neurons. To enable the neurons to adaptively adjust the size of the receptive field according to the content, the information
flow of the branches is controlled using a gate mechanism. The results of the different branches are first aggregated by summing the elements, as shown in Figure 3. The equation is shown in (1).

\[
U = \tilde{U} + \hat{U}
\]  

(1)

The channel statistics \( s \in \mathbb{R}^C \) are generated by embedding the global information using global average pooling, where \( C \) denotes the feature dimension of \( s \). The formula is shown in (2).

\[
s_c = \mathcal{F}_{gp}(U_c) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} U_c(i,j)
\]  

(2)

The features \( z \in \mathbb{R}^{d \times 1} \) are then computed by the fully connected (fc) layer, with \( z \) being equivalent to the squeeze operation in the SE module, to enable accurate and adaptive feature selection to improve efficiency while reducing dimensionality. The formula is shown in (3).

\[
z = \mathcal{F}_{fc}(s) = \delta(B(Ws))
\]  

(3)

where \( \delta \) is the ReLU function, \( B \) is the batch normalization, \( W \in \mathbb{R}^{d \times C} \). To investigate the effect of the parameter \( d \) on the model efficiency, a decay rate \( r \) is used to control its value, as shown in Equation (4).

\[
d = \max(C/r, L)
\]  

(4)

where \( L \) denotes the minimum value of \( d \). The default \( L = 32 \).

**Select:** After the **Fuse** step, the information on different spatial scales is selected adaptively using cross-channel soft attention with feature \( z \) as a guide. The softmax operation is taken for the channels to obtain Equation (5).

\[
a_c = \frac{e^{A_c z}}{\sum_i e^{A_i z}}, b_c = \frac{e^{B_c z}}{\sum_i e^{B_i z}}
\]  

(5)

where \( A, B \in \mathbb{R}^{C \times d}, a, b \) denote the soft attention vectors of the two branches \( \tilde{U} \) and \( \hat{U} \), respectively. \( A_c \in \mathbb{R}^{1 \times d} \) denotes the \( c \)-th row of \( A \), \( a_c \) denotes the \( c \)-th element of \( a \), and \( B_c, b_c \) do the same. In the case of two branches, \( B \) denotes the redundancy matrix, \( a_c + b_c = 1 \). The feature map \( V \) in the final is obtained by adding attention weights to the convolution kernel, as shown in Equation (6).

\[
V_c = a_c \cdot \tilde{U}_c + b_c \cdot \hat{U}_c, a_c + b_c = 1
\]  

(6)

where \( V = [V_1, V_2, \cdots, V_C], V_C \in \mathbb{R}^{H \times W} \).

**3.2. Embed to Token**

It was pointed out [31] that the low-level features of patches help to enhance the robustness of the model. However, the ViT model simply segments the images into patches of the same size, and this simple segmentation does not allow the model to better extract low-level features such as edges, curves, and contours. To extract the low-level features of patches, the CeiT [32], LeViT [25], and TNT [33] models use convolution operations instead of linear mapping in the ViT model. Unlike in these methods, in the model described in this paper, patches are firstly input into the SE module, as shown in Figure 2a, which can effectively highlight the important features of the feature map, reduce and suppress the redundant features, and help the model to extract the key features. Secondly, the extracted features are input into the Convolutional Token Embedding module, and the convolutional operation effectively utilizes the low-level features of patches; extracting more visual features can improve the robustness of the model.
3.3. Global Average Pooling

The ViT model uses Classification Token (CLS) for the final image classification, but studies have pointed out that CLS tokens cannot exploit the translation invariance of images and can impair the robustness of the model when classifying examples. To address this problem, CPVT [34] and LeViT [25] used averaging pooling instead of CLS tokens in the last layer, and the results showed that this method can improve classification accuracy. Therefore, the model proposed in this paper uses global average pooling instead of CLS labeling to improve the robustness of the model by fusing the visual features of different spatial regions.

3.4. Adversarial Training

In addition to introducing the SK module in the RVT model, the robustness of deep neural networks is enhanced using adversarial training methods. Adversarial training, as one of the most promising defense methods available, can regularize the parameters of deep neural networks by fusing adversarial examples with clean examples for training, adapting the network model to adversarial attacks while enhancing the defense against emerging adversarial examples.

The Min-Max problem was first defined by Huang et al. [35]. Later, Shaham et al. [36] further optimized the Min-Max problem and proposed a framework for adversarial training, as shown in Equation (7).

$$\min_\theta \mathbb{E}_{(Z,y) \sim D} \left[ \max_{||\delta|| \leq \varepsilon} L(f_{\theta}(X + \delta), y) \right]$$  \hspace{1cm} (7)

where \(\max()\) denotes maximization, \(X\) denotes the input example, \(\delta\) denotes the perturbation added to the input example, \(f_{\theta}()\) denotes the deep neural network, \(y\) denotes the true label of the clean example, and \(L(f_{\theta}(X + \delta), y)\) denotes the loss between the output label of the adversarial example \(X + \delta\) passing through the deep neural network and the true label. \(\max(L)\) denotes the optimization objective, which aims to find the perturbation that maximizes the loss function so that the added perturbation has to disturb the deep neural network as much as possible.

\(\min()\) is a formula for optimizing a deep neural network when the adversarial perturbation has been determined, and training the deep neural network to minimize its loss value on the training set makes the model robust to the adversarial perturbation. Equation (7) describes the idea of adversarial training by definition, but does not describe how to generate adversarial perturbations with stronger attack performance \(\delta\). To address this problem, researchers have proposed multiple attack methods to generate perturbations \(\delta\). In fact, for adversarial training defense methods, generating perturbations with stronger attack performance \(\delta\) can make the deep neural network more robust.

4. Experimental Design and Analysis of Results

4.1. Experimental Platform

The experimental platform of this study is based on ubuntu 18.04; the hardware device is NVIDIA Tesla A100 GPU with 40G of video memory; the experimental environment is PyTorch deep learning framework supporting GPU acceleration, and the cuda environment is configured with NVIDIA CUDA 11.3 and cuDNN V8.2.1 deep learning acceleration library.

4.2. Dataset Setup and Evaluation Metrics

This experiment uses the miniImageNet dataset to verify the validity of the model. miniImageNet contains a total of 100 classes, including 64 classes in the training set, 16 classes in the validation set, and 20 classes in the test set, each containing 600 images, for a total of 60,000 data examples of size \(84 \times 84\). In the experimental process, the data is first preprocessed by upsampling the examples to \(224 \times 224\) pixel size, and then the adversarial examples are generated using the white-box adversarial attack methods C&W,
DI²FGSM, MDI²FGSM, and the black-box adversarial attack methods P-RGF and RGF, Parsimonious. In addition to using the miniImageNet dataset, two naturally generated datasets, ImageNet-A and ImageNet-R, are used, where the ImageNet-A dataset consists of real-world, unmodified and naturally occurring examples that are misclassified by the ResNet50 neural network, which contains 200 categories with a total of 7500 examples. The ImageNet-R dataset is a subset of the ImageNet dataset, which renders examples from the ImageNet dataset in 16 different styles such as art, cartoon, graffiti and sketch, and contains a total of 30,000 renderings of images from 200 ImageNet categories. Accuracy is used as an evaluation metric in evaluating the model. Specifically, the adversarial examples generated are fed into the ViT classification model and the output labels of the ViT model are recorded. The ViT model is considered to be correctly classified if the output labels are the same as the clean example labels. Therefore, the accuracy rate is the proportion of the number of correctly classified adversarial examples by the ViT model to the whole number of examples. A higher accuracy rate represents a better robust performance of the model.

4.3. Parameter Setting

The cross-entropy loss function is used during training and the parameters of the model are optimized using the Adam optimizer, which is suitable for situations where the objective function is unstable and prevents the parameters from falling into local minima, and is also suitable for scenarios with large datasets and can accelerate the training of the neural network; the learning rate is set to $\beta = 0.0005$; the momentum size is 0.9; the weight decay is 0.05. The parameters are updated using the SoftMax loss function, 20% of the adversarial examples are used in the adversarial training process with a clean mixture for adversarial training. The parameter settings are shown in Table 1.

| Parameter          | Value          |
|--------------------|----------------|
| Learning Rate      | 0.0005         |
| Epoch              | 100            |
| Decay Epochs       | 30             |
| Weight Decay       | 0.05           |
| Optimizer Epsilon  | $1 \times 10^{-8}$ |
| Batch Size         | 32             |
| Momentum           | 0.9            |
| Learning Rate Decay| 0.1            |

4.4. Analysis of Experimental Results

4.4.1. Comparison with Different Network Structures

For the proposed defense method, Transformer models with different scales of TNT networks, PyramidTNT structure, T2T-ViT structure, and DeiT structure are selected and compared with the proposed defense method to verify the robustness of the proposed defense method against different adversarial examples, and the experimental results are shown in Table 2. Table 2 shows the comparison of the defense performance of the proposed defense method with several other Transformer models and their variants. It can be found that the proposed defense method has a higher accuracy rate in the face of different adversarial example attack methods compared to several other Transformer structures. For example, the defense performance of the proposed method can reach 30.104% in the face of C&W attacks, which is much higher than the 11.325% of the DeiT-base model, showing a strong robustness; the accuracy of the proposed defense method against the MDI²FGSM attack method with stronger attack performance is 28.885%, while the accuracy of TNT-B is only 13.596%. The reason for this is that the SK module introduced as a dynamic selection mechanism can adaptively select the size of the perceptual field based on the input multi-
scale information in order to capture effective features. Secondly, the training process uses the adversarial training method to make the deep neural network fit the distribution of the data and further improve the robustness of the deep neural network. Figure 4 shows the comparison of the proposed defense method with other Transformer structures.

Table 2. Comparison of different network structures.

| Method Type | C&W | DF²FGSM | MDF²FGSM | ImageNet-A | ImageNet-R |
|-------------|-----|---------|----------|------------|------------|
| **Transformer** |     |         |          |            |            |
| TNT-S       | 11.066 | 11.245 | 12.464   | 23.160     | 44.067     |
| TNT-B       | 13.562 | 11.914 | 13.596   | 25.587     | 46.173     |
| ConViT-Ti   | 11.562 | 8.152  | 9.826    | 8.373      | 36.037     |
| ConViT-S    | 12.179 | 10.654 | 10.288   | 23.987     | 43.303     |
| ConViT-B    | 11.129 | 13.868 | 12.568   | 28.334     | 47.073     |
| T2T-ViT-14  | 10.219 | 10.893 | 10.161   | 27.187     | 42.523     |
| T2T-ViT-24  | 9.383  | 8.067  | 9.908    | 7.300      | 32.210     |
| DeiT-tiny   | 10.242 | 9.983  | 10.967   | 14.387     | 42.487     |
| DeiT-small  | 11.325 | 10.867 | 9.042    | 23.640     | 44.630     |

| **ASK-ViT** |     |         |          |            |            |
| Tiny        | 20.042 | 22.833 | 19.792   | 14.773     | 44.760     |
| Tiny-plus   | 26.219 | 26.917 | 26.417   | 15.586     | 45.267     |
| Small       | 26.020 | 27.823 | 26.344   | 25.053     | 46.853     |
| Small-plus  | 26.771 | 28.385 | 27.208   | 26.667     | 47.640     |
| Base        | 28.479 | 27.094 | 26.260   | 28.480     | 48.587     |
| Base-plus   | **30.104** | **29.146** | **28.885** | **29.573** | **48.867** |

Figure 4. Comparison of different Transformer structures with SK-ViT.

As can be seen in Figure 4, the feature maps extracted by SK-ViT are most similar to those of Clean compared to the other four structures, which indicates that the SK module can capture minute features better and is able to remap the adversarial examples back to the clean example space. It also further validates the results in Table 2.
4.4.2. Comparison of ASK-Vision Transformer with SE-Vision Transformer and Vision Transformer Defense Performance

The proposed defense method is an improvement on the ViT model, which achieves better performance in the defense against attacks, in terms of accuracy, and robustness against adversarial examples. In order to verify that the proposed defense method has strong robustness, this section will study the defense performance of ASK-ViT and SE-ViT, two ViT defense methods, in the face of different white-box and black-box attacks, and investigate in depth the impact of SK module on the robustness performance of neural network. The experimental results are shown in Tables 3 and 4.

Table 3. Performance of ASK-RVT with SE-ViT in defending against white-box attacks.

|               | ASK-ViT | SE-ViT | ViT (Base-Plus) |
|---------------|---------|--------|-----------------|
| C&W           | 28.677  | 17.406 | 12.031          |
| DF²FGSM       | 28.417  | 21.667 | 11.281          |
| MDI²FGSM      | 27.375  | 18.635 | 10.250          |
| ImageNet-A    | 28.720  | 24.640 | 14.667          |
| ImageNet-R    | 48.693  | 35.533 | 28.733          |

Table 4. Performance of ASK-RVT with SE-ViT and ViT in defending against black box attacks.

|               | ASK-ViT | SE-ViT | ViT       |
|---------------|---------|--------|-----------|
| RGF           | 29.979  | 19.406 | 17.021    |
| P-RGF         | 29.896  | 19.667 | 18.833    |
| Parsimonious  | 29.469  | 18.865 | 17.948    |

The results in Tables 3 and 4 show that the proposed defense method achieves high accuracy and shows strong robustness for both white-box and black-box attacks, indicating the effectiveness of ASK-ViT. By analyzing ASK-ViT with SE-ViT and ViT, it is found that the SE module can highlight the important features of the channel, effectively extract increasingly complex features, and suppress secondary information. Therefore, the robustness of the SE-ViT model is better than that of the ViT model; compared with the SE module, the role of the SK module is to select the size of the receptive field adaptively according to the features on different scales, and to capture the classification features automatically using the receptive fields of different sizes. Therefore, the robustness of the ASK-ViT model is better than that of the SE-ViT model.

Figure 5 shows the attention maps of ASK-ViT, SE-ViT and ViT models in extracting features, where X denotes the input features, X + Attention denotes the feature map after the Attention mechanism of ViT model; X + SE denotes the feature information extracted after feature X has gone through the SE module; X + SK denotes the feature information output after feature X has gone through the SK module.

As shown in Figure 5, the output feature map is X + Attention after inputting the feature information of the Adversarial example X into the Attention mechanism, which shows that the feature map does not highlight the key information, and the deep neural network cannot extract useful features, resulting in a poor robustness of the network model. The feature map after SE module is X + SE. Compared with the X + Attention feature map, X + SE can highlight the information of the channel and highlight the key feature information, which helps the network model to improve the robustness. X + SK is the feature map after SK model. Compared with the Attention mechanism and SE module, SK adaptively selects the feeling field according to the feature information, and the X + SK feature map emphasizes more on the content understanding while highlighting the feature information. Therefore, the introduction of the SK module can make the deep neural network more robust compared with the other two mechanisms.
4.4.3. Effect of Adversarial Training on Robustness

After verifying the effectiveness of the SK module, the effect of Adversarial Training on the robustness of the network model is further verified. Therefore, two different sets of experiments are designed in this section to compare SK-ViT and ASK-ViT. The experimental results are shown in Table 5. It can be seen that the accuracy of ASK-ViT is 30.104%, while SK-ViT is 28.677% in the face of C&W attacks; the accuracy of ASK-ViT is 0.853% and 0.174% higher than SK-ViT in the face of natural datasets ImageNet-A and ImageNet-R, respectively. It shows that the robustness is better than SK-ViT model in defending against the adversarial examples.

4.4.4. Effect of Patch_Augmentation on Robustness

To investigate the effects of different image enhancement methods on the robustness of neural networks, three image enhancement methods, Random Resized Crop (RC), Random...
Gaussian Noise (GN) and Random Horizontal Flip (HF), are used. For the RC image enhancement method, the size of scale is set to [0.80, 1.0] and the size of ratio is set to [1.0, 1.0]. For the GN image enhancement method, the mean and standard deviation were set to 0 and 0.01, respectively. The experimental results are shown in Table 6, where Acc denotes the accuracy of clean examples and Rob.Acc denotes the accuracy of the proposed defense method in the face of four different adversarial attack methods C&W, MDF²FGSM, and ImageNet-A. The last row of the table denotes the accuracy of the proposed defense method without Patch_Augmentation.

Table 6. Effect of image enhancement methods on the robustness of ASK-ViT.

| Augmentations | RC | GN | HF | Acc  | C&W    | Rob.Acc | ImageNet-A |
|---------------|----|----|----|------|--------|---------|------------|
| √             | √  | √  | √  | 77.740 | 28.615 | 28.604  | 30.177     |
| √             | √  | √  | √  | 77.230 | 28.552 | 29.662  | 29.563     |
| √             | √  | √  | √  | 77.427 | 29.458 | 29.656  | 29.417     |
| √             | √  | √  | √  | 78.000 | 25.156 | 30.125  | 29.813     |
| √             | √  | √  | √  | 78.052 | 30.156 | 29.073  | 29.917     |
| √             | √  | √  | √  | 77.510 | 29.354 | 29.448  | 29.688     |
| √             | √  | √  | √  | 78.781 | 30.104 | 28.885  | 29.573     |
| √             | √  | √  | √  | 76.656 | 29.427 | 27.010  | 30.302     |

Table 6 shows the impact of different image enhancement methods on the robustness performance of the proposed defense method. It can be seen that all three image enhancement methods, RC, GN and HF, help to improve the robustness of the deep neural network in the face of different adversarial sample attack methods. In addition, the combination of different image enhancement methods is also compared, and it can be seen that the image enhancement after the combination does not improve much in terms of accuracy compared to a single image enhancement method; it even decreases the accuracy, for example, after using the combination of RC + HF, the accuracy is only 25.156% in the face of the C&W attack. The last row in the table indicates that no image enhancement is used. The results show that the accuracy rate decreases. Therefore, after weighing the experimental results and robustness, the image enhancement combination method of RC + GN + HF is used.

Figure 6 shows the attention mapping of different image enhancement methods in extracting features.
As shown in Figure 6, several different image enhancement methods in Table 6 are visualized and it is observed that when extracting features, each image enhancement method differs so little in showing important features that the deep neural network cannot extract them effectively. Therefore, several image enhancement methods and their different combinations are not effective in improving the robustness of the deep neural network in the face of adversarial sample attacks. Figure 6 also further validates the experimental results in Table 6.

4.4.5. Comparison of Cifar10 and Cifar100 Defense Performance

In addition to the validation on miniImageNet, ImageNet-A and ImageNet-R data, the robustness is also investigated on the Cifar10 and Cifar100 datasets. The Cifar10 dataset is a subset of the Tiny Images dataset, which consists of 60,000 $32 \times 32$ color images, with 10 categories, each consisting of 5000 training images and 1000 test images. The Cifar10 dataset is a subset of the Tiny Images dataset, which consists of 60,000 $32 \times 32$ color images, with 10 categories, each consisting of 5000 training images and 1000 test images.

As shown in Table 7, the ViT model with the ResNet model and its data variants using the literature [37] show poor robustness against FGSM and PGD attacks; especially, the Cifar10 dataset is successfully attacked with ViT-B-16 models, ResNet-56 and ResNet-164 models with an accuracy of 0%, against PGD attack. In contrast, the SK-ViT method shows stronger robustness against both attack methods, for example, the accuracy of SK-ViT-Base is 40.9% and 22.7% for the Cifar10 and Cifar100 datasets, respectively, in the face of PGD attacks. It shows that the SK module adaptively extracts features that help the model to perform classification and can improve the robustness of the deep neural network.

Table 7. Experimental results of Cifar10 and Cifar100 datasets.

|         | Cifar10 | Cifar100 |
|---------|---------|----------|
|         | FGSM    | PGD      | FGSM    | PGD      |
| ViT-B-16| 39.5    | 0.0      | 20.4    | 0.0      |
| ViT-B-32| 37.9    | 1.8      | 20.8    | 1.9      |
| ViT-R50 | 40.8    | 0.1      | 22.0    | 0.2      |
| ResNet-56| 23.0    | 0.0      | 6.0     | 0.2      |
| ResNet-164| 29.0  | 0.0      | 7.6     | 0.3      |
| SK-ViT  | Tiny    | 51.6     | 39.8    | 30.5    | 20.2    |
|         | Small   | 54.5     | 32.8    | 34.2    | 23.7    |
|         | Base    | 50.9     | 40.9    | 35.1    | 22.7    |

5. Discussion

In this work, the SK-RVT model is proposed based on the RVT model, which is an effective defense method. First, the SK module is introduced into the Transformer structure, and the multi-scale spatial information of the features is extracted using the SK module to adaptively adjust the size of the perceptual field, which helps the model to perform the extraction of key features. A training set consisting of a mixture of clean and adversarial examples is then used to train the network model adversarially to further enhance the robustness of the deep neural network. The experimental results show that the proposed defense method exhibits strong robustness under a variety of adversarial attacks.

Author Contributions: Y.C.: Conceptualization, methodology, and writing original draft; H.Z. and W.W.: writing assistance. All authors have read and agreed to the published version of the manuscript.

Funding: This research is supported by the National Natural Science Foundations of China (62166025); Science and Technology Project of Gansu Province (21YF5GA073).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.
Data Availability Statement: The experimental data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest: The authors declare that they have no conflict of interest.

References

1. Kim, S.-H.; Nam, H.; Park, Y.-H. Decomposed Temporal Dynamic CNN: Efficient Time-Adaptive Network for Text-Independent Speaker Verification Explained with Speaker Activation Map. arXiv 2022, arXiv:2203.15277.

2. Kim, H.; Jung, W.-K.; Park, Y.-C.; Lee, J.-W.; Ahn, S.-H. Broken stitch detection method for sewing operation using CNN feature map and image-processing techniques. Expert Syst. Appl. 2022, 188, 116014. [CrossRef]

3. Messina, N.; Amato, G.; Esuli, A.; Falchi, F.; Gennaro, C.; Marchand-Maillet, S. Fine-grained visual textual alignment for cross-modal retrieval using transformer encoders. ACM Trans. Multimed. Comput. Commun. Appl. 2021, 17, 1–23. [CrossRef]

4. Zhang, W.; Wu, Y.; Yang, B.; Hu, S.; Wu, L.; Dhelim, S. Overview of multi-modal brain tumor mr image segmentation. Proc. Healthc. 2021, 9, 1051. [CrossRef] [PubMed]

5. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need.

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

7. Strudel, R.; Garcia, R.; Laptev, I.; Schmid, C. Segmentor: Transformer for semantic segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 11–17 October 2021; pp. 7262–7272.

8. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 11–17 October 2021; pp. 10012–10022.

9. Fu, Y.; Zhang, C.; Kanaju, A.; Kweon, I.S. Adversarial robustness comparison of vision transformer and mlp-mixer to cnns. arXiv 2021, arXiv:2110.02797.

10. Bhojanapalli, S.; Chakrabarti, A.; Glasner, D.; Li, D.; Unterthiner, T.; Veit, A. Understanding robustness of transformers for image classification. IEEE Trans. Multimed. 2022. [CrossRef]

11. Chen, H.; Li, C.; Wang, G.; Li, X.; Rahaman, M.M.; Sun, H.; Hu, W.; Li, Y.; Liu, W.; Sun, C. GasHis-Transformer: A multi-scale visual transformer approach for gastric histopathological image detection. Pattern Recognit. 2022, 130, 108827. [CrossRef]

12. Liu, C.; Yang, H.; Fu, J.; Qian, X. Learning Trajectory-Aware Transformer for Video Super-Resolution. arXiv 2022, arXiv:2204.04216.

13. Benz, P.; Ham, S.; Zhang, C.; Kanaju, A.; Kweon, I.S. Adversarial robustness of vision transformer and mlp-mixer to cnns. arXiv 2021, arXiv:2110.02797.

14. Shao, R.; Shi, Z.; Yi, J.; Chen, P.-Y.; Hsieh, C.-J. On the adversarial robustness of visual transformers. arXiv 2021, arXiv:2103.15670.

15. Xie, C.; Wu, Y.; Maaten, L.v.d.; Yuille, A.L.; He, K. Feature denoising for improving adversarial robustness. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 501–509.

16. Das, N.; Shanbhogue, M.; Chen, S.-T.; Hoffman, F.; Chen, L.; Kounavis, M.E.; Chau, D.H. Keeping the bad guys out: Protecting and vaccinating deep learning with jpeg compression. arXiv 2017, 1705.02900.

17. Prakash, A.; Moran, N.; Garber, S.; DiLillo, A.; Storer, J. Deflecting adversarial attacks with pixel deflection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 8571–8580.

18. Zhang, H.; Wang, J. Defense against adversarial attacks using feature scattering-based adversarial training. In Proceedings of the 33rd International Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019; Volume 32, pp. 1831–1841.

19. Paperoni, N.; McDaniel, P.; Wu, X.; Jha, S.; Swami, A. Distillation as a defense to adversarial perturbations against deep neural networks. In Proceedings of the 2016 IEEE Symposium on Security and Privacy (SP), San Jose, CA, USA, 22–26 May 2016; pp. 582–597.

20. Samangouei, P.; Kabkab, M.; Challappa, R. Defense-gan: Protecting classifiers against adversarial attacks using generative models. arXiv 2018, arXiv:1805.06605.

21. Meng, D.; Chen, H. Magnet: A two-pronged defense against adversarial examples. In Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, Dallas, TX, USA, 30 October–3 November 2017; pp. 135–147.

22. Nesti, F.; Biondi, A.; Buttazzo, G. Detecting Adversarial Examples by Input Transformations, Defense Perturbations, and Voting. arXiv 2021, arXiv:2101.11466. [CrossRef] [PubMed]

23. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Shuffle transformer: Rethinking spatial shuffle for vision transformer. arXiv 2021, arXiv:2106.03650.
25. Graham, B.; El-Nouby, A.; Touvron, H.; Stock, P.; Joulin, A.; Jégou, H.; Douze, M. LeViT: A Vision Transformer in ConvNet’s Clothing for Faster Inference. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 11–17 October 2021; pp. 12259–12269.

26. Srinivas, A.; Lin, T.-Y.; Parmar, N.; Shlens, J.; Abbeel, P.; Vaswani, A. Bottleneck transformers for visual recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 16519–16529.

27. Kim, B.J.; Choi, H.; Jang, H.; Lee, D.G.; Jeong, W.; Kim, S.W. Improved Robustness of Vision Transformer via PreLayerNorm in Patch Embedding. arXiv 2021, arXiv:2111.08413.

28. Bai, J.; Yuan, L.; Xia, S.-T.; Yan, S.; Li, Z.; Liu, W. Improving Vision Transformers by Revisiting High-frequency Components. arXiv 2022, arXiv:2204.00993.

29. Heo, B.; Yun, S.; Han, D.; Chun, S.; Choe, J.; Oh, S.J. Rethinking spatial dimensions of vision transformers. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 11–17 October 2021; pp. 11936–11945.

30. Popovic, N.; Paudel, D.P.; Probst, T.; Van Gool, L. Improving the Behaviour of Vision Transformers with Token-consistent Stochastic Layers. arXiv 2021, arXiv:2112.15111. [CrossRef]

31. Mao, X.; Qi, G.; Chen, Y.; Li, X.; Duan, R.; Ye, S.; He, Y.; Xue, H. Towards robust vision transformer. arXiv 2021, arXiv:2105.07926.

32. Yuan, K.; Guo, S.; Liu, Z.; Zhou, A.; Yu, F.; Wu, W. Incorporating convolution designs into visual transformers. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 11–17 October 2021; pp. 579–588.

33. Han, K.; Xiao, A.; Wu, E.; Guo, J.; Xu, C.; Wang, Y. Transformer in transformer. In Proceedings of the 35th Conference on Neural Information Processing Systems, virtual, 6–14 December 2021; Volume 34, pp. 15908–15919.

34. Chu, X.; Zhang, B.; Tian, Z.; Wei, X.; Xia, H. Do We Really Need Explicit Position Encodings for Vision Transformers? arXiv 2021, arXiv:2102.10882.

35. Huang, R.; Xu, B.; Schuurmans, D.; Szepesvári, C. Learning with a strong adversary. arXiv 2015, arXiv:1511.03034.

36. Shaham, U.; Yamada, Y.; Negahban, S. Understanding adversarial training: Increasing local stability of supervised models through robust optimization. Neurocomputing 2018, 307, 195–204. [CrossRef]

37. Mahmood, K.; Mahmood, R.; Van Dijk, M. On the robustness of vision transformers to adversarial examples. arXiv 2021, arXiv:2104.02610.