Neural architecture search (NAS) could help search for robust network architectures, where defining robustness evaluation metrics is the important procedure. However, current robustness evaluations
in NAS are not sufficiently comprehensive and reliable. In particular, the common practice only considers adversarial noise and quantified metrics such as the Jacobian matrix, whereas, some studies indicated that the models are also vulnerable to other types of noises such as natural noise. In addition, existing methods taking adversarial noise as the evaluation just use the robust accuracy of the FGSM or PGD, but these adversarial attacks could not provide the adequately reliable evaluation, leading to the vulnerability of the models under stronger attacks. To alleviate the above problems, we propose a novel framework, called Auto Adversarial Attack and Defense (AAAD), where we employ neural architecture search methods, and four types of robustness evaluations are considered, including adversarial noise, natural noise, system noise and quantified metrics, thereby assisting in finding more robust architectures. Also, among the adversarial noise, we use the composite adversarial attack obtained by random search as the new metric to evaluate the robustness of the model architectures. The empirical results on the CIFAR10 dataset show that the searched efficient attack could help find more robust architectures.

**Keywords** Neural architecture search, adversarial attack and defense, robustness evaluation

### 1 Introduction

Deep neural networks (DNN) have been applied widely in various tasks such as image classification [Chen et al. (2022)], object detection [Zhou et al. (2021)], and image segmentation [Cai et al. (2022)] in the real world. However, they have been proved to be vulnerable against the adversarial example (AE), that is, the images with subtle perturbations may cause the wrong prediction of the model. Currently, various adversarial attack algorithms are developed, including the fast gradient sign method (FGSM) [Goodfellow et al. (2015)] and projected gradient descent (PGD) [Madry et al. (2018)], which cause many serious safety issues. Thus, designing robust model architecture against AE has received increasing attention in adversarial defense.

Neural architecture search (NAS) is one of the effective approaches to alleviate the above problem, which can obtain a robust model [Guo et al. (2020)]. Roughly, the process of NAS includes defining search space, conducting search strategy and defining evaluation metrics, where the robustness evaluation metrics is the important procedure. Existing work about robustness metrics in NAS can be categorized into two types: robust accuracy of adversarial training and quantified metrics. In the robust accuracy of adversarial training, the designers usually use the accuracy of the adversarially trained model under different adversarial noises to evaluate the robustness of each model architecture. Typically, Chen et al. [Chen et al. (2022)] develop ABanditNAS, which employs FGSM to perform adversarial training as the robust accuracy. Unlike ABanditNAS, Liu et al. [Liu and Jin (2021a)] apply multiple types of adversarial attacks to execute adversarial training. In the quantified metric, the researchers generally combine the attack-agnostic metric into the total loss of differentiable architecture search (DARTS) [Liu et al. (2018)], greatly decreasing the evaluation cost. Hosseini et al. [Hosseini et al. (2021)] incorporate two robustness metrics, including Jacobian matrix and certified lower bound into partial channel connections for memory-efficient architecture search (PC-DARTS) [Xu et al. (2019)]. Similarly, Mok et al. [Mok et al. (2021)] use the Hessian matrix information of input data as the robustness evaluation to guide the process of DARTS [Liu et al. (2018)].

![Figure 1: The motivation of our method. We list the robust accuracy of five models under 7 adversarial attacks. The robustness of five models ranked by FGSM or PGD are not consistent with stronger attacks. It causes that the evaluation of FGSM or PGD is not reliable in NAS for robust architectures.](image-url)
From the above work, it can be seen that a small number of algorithms have been proposed to design a variety of metrics for the robust architecture. However, there are two problems remaining unexplored. On the one hand, the robustness evaluation is not comprehensive. Current work evaluate the robustness under different adversarial attack and quantified metrics, but the robustness of the model also needs to consider the robust accuracy under other types of noises such as natural noise [Hendrycks et al. (2021a)] and system noise [Wang et al. (2021a); Tang et al. (2021)]. On the other hand, under adversarial noise, the robustness evaluation is not reliable. The reason is that the metrics with adversarial training have been proved to bring overestimation of the robustness, resulting in the searched model by NAS not being robust enough. For example, the robust accuracy of five searched model architectures is presented in Fig 1, which can be obtained from the report of AdvRush [Mok et al. (2021)]. If the evaluation just considers the robust accuracy of FGSM or PGD, it could cause the misjudgment of robust architectures in NAS. Similar phenomena can also be seen in other related work [Liu and Jin (2021a); Hosseini et al. (2021)].

To alleviate the above two problems, we propose a novel framework called auto adversarial attack and defense (AAAD), as shown in Fig 2. Specially, to solve the uncomprehensive evaluation, we realize four types of robustness evaluations, including adversarial noise, system noise, natural noise and quantified metric in NAS, and design the corresponding loss functions to search for robust architectures. Different from the adversarial noise and quantified metric in the above work, the natural noise is constructed using the style transfer technique, and the system noise is synthesized using inconsistent implementations of decoding and resize. To introduce the reliable evaluation, we propose an improved composite adversarial attack as the robustness metric. In detail, after training the architecture sampled randomly in the model search space, we conduct the random search algorithm to search for the near-optimal configuration, including the attack operations, attack magnitude and attack step. Compared with existing attack methods, the searched composite adversarial attack could have more efficient performance and contribute to finding more robust architectures. The four above types of robustness metrics and the composite adversarial attack are combined into a unified framework to search for robust architectures. Extensive experiments on the CIFAR10 dataset are conducted to investigate the performance of different metrics and demonstrate the effectiveness of our proposed framework. In this paper, our contributions could be summarized as follows:

1) To alleviate the uncomprehensive robustness evaluation in NAS, we provide four types of metrics, namely adversarial noise, natural noise, system noise and quantified metrics. According to each type of metric characteristic, we design the corresponding loss function to search for robust architectures.

2) To alleviate the unreliable robustness evaluation in NAS, we incorporate the improved composite adversarial attack as the evaluation, resulting in finding more robust architectures.

3) Four types of evaluation metrics and the composite adversarial attack are formed into a unified framework. Experiments on the CIFAR10 dataset show that our proposed framework can help find more robust architectures than existing methods.

2 Background and Related Work

2.1 Adversarial Attack

The adversarial attack could deceive DNN models by adding the specific perturbation in the original image, and it can be expressed as below:

$$\begin{align*}
\arg \max_{\Delta x} & \quad \mathcal{L}(x_{adv}, y; F) \\
x_{adv} & = x + \Delta x \\
\text{s.t.} & \quad \|\Delta x\| \leq \epsilon,
\end{align*}$$

(1)

where $x$ represents the original image and $\Delta x$ is the perturbation added on the clean image, subject to a magnitude $\epsilon$, $y$ is the original label, and $F$ is the model. $x_{adv}$ refers to an adversarial example. $\mathcal{L}$ is the loss function between the output of the model and the original label $y$ in image classification. To solve Eq. 1, a majority of adversarial attack algorithms are developed.

FGSM updates the perturbation on the original image $x$ by the single step along the direction of the gradient of the cross entropy loss function, which can be expressed as Eq. 2

$$x_{adv} = \text{clip}_{[0,1]} \left\{ x + \epsilon \cdot \text{sign} (\nabla_x \mathcal{L}(x, y; F)) \right\}.$$  

(2)

Different from updating the image using the single step, PGD performs multiple update under the random start. The update process of PGD is illustrated in Eq. 3

$$x_{l+1} = \text{project} \left\{ x_l + \epsilon_{\text{step}} \cdot \text{sign} (\nabla_x \mathcal{L}(x_l, y; F)) \right\}.$$  

(3)
where projecting can keep \( x_l \) within the range of the predefined magnitude.

Subsequently, to enhance the performance of adversarial attack, a majority of attack algorithms are developed such as Carlini & Wagner (CW) attack \( \text{Carlini and Wagner (2017)} \), Momentum Iterative (MI) attack \( \text{Dong et al. (2018)} \), DDN attack \( \text{Rony et al. (2019)} \), Spatial attack \( \text{Engstrom et al. (2019)} \) and MultiTargeted attack \( \text{Gowal et al. (2019)} \), leading to much threat to robustness of the DNN models.

### 2.2 Adversarial Defense

To enhance the robustness of the DNN against the noises generated by the adversarial attacks, a variety of adversarial defense methods have been developed, including the adversarial training \( \text{Madry et al. (2018)} \), the feature denoising \( \text{Xie et al. (2019)} \), the image preprocessing \( \text{Song et al. (2018)} \), the model ensemble \( \text{Pang et al. (2019)} \), the regularization \( \text{Lin et al. (2019)} \), the random smoothing \( \text{Chen et al. (2021)} \) and the certified based techniques \( \text{Boopathy et al. (2019)} \). Apart from the adversarial noises, some efforts have also been devoted to studying the robustness of the DNN under the system and natural noise \( \text{Hendrycks et al. (2021a); Wang et al. (2021a); Tang et al. (2021)} \). System noise stands for the examples brought by inconsistent implementations of image decoding and resize. Natural noise refers to the examples that look natural but fool the models, which are naturally existed or generated by the style transfer technique.

In addition to the above adversarial defense methods, recent work explores the intrinsic robustness of the model architectures selves. Consequently, designing robust model architectures has been an important problem attracting much attention. NAS is one of the effective methods of designing model architectures, which intends to automatically search for a better neural architecture for the specific task \( \text{Liu et al. (2018)} \).

NAS could be modeled into a bi-level optimization problem. The inner problem is the gradient-based weight optimization, while the outer one is the model architecture parameters optimization. The mathematical model of NAS can be modeled as Eq. 4.

\[
\min_{\alpha} \quad L_{\text{val}}(w^*(\alpha), \alpha) \\
\text{s.t.} \quad w^*(\alpha) = \arg\min_w L_{\text{train}}(w, \alpha), \quad (4)
\]

where \( \alpha \) denotes the searched architecture, \( w^* \) stands for the optimal weights for the selected architecture. To solve the above optimization problem, a majority of NAS algorithms are proposed, which are mainly divided into differentiable and non-differentiable methods \( \text{Yao et al. (2020)} \). Currently, both of two types of NAS methods have been applied to search for robust architectures. The non-differentiable methods adopt the robust accuracy of FGSM or PGD adversarial training as the evaluation, which is relatively time-consuming \( \text{Guo et al. (2020); Liu and Jin (2021b); Vargas et al. (2019)} \). Further, the differentiable methods take the Jacobian or Hessian matrix to measure the robustness of the models, resulting in the great acceleration of the search procedure \( \text{Mok et al. (2021); Hosseini et al. (2021)} \).

### 3 The Framework of Auto Adversarial Attack and Defense

In this section, the proposed auto adversarial attack and defense framework is introduced. On the one hand, to provide a more comprehensive evaluation for NAS, four types of robustness metrics are developed. On the other hand, to provide a more reliable evaluation for NAS, the composite adversarial attack with large search space (LCAA) is proposed. Our proposed framework is illustrated in Fig. 2. In the AAAD framework, the bottom module is neural architecture search for robust network architectures. In the robustness evaluation, we provide four types of evaluations comprehensively, including adversarial noise, system noise, natural noise and quantified metrics. The top module is the evolutionary search for the efficient attack schemes, where the efficient composite attack scheme could be found under the case that the threat model architecture is randomly generated in the model search space. The searched attack scheme could be used as the reliable robustness evaluation in adversarial noise to guide the process of searching for robust architectures. Then we describe the AAAD framework in detail.

#### 3.1 Robustness Evaluation

In this section, four types of robustness evaluations, including adversarial noises, natural noise, system noise and quantified metrics are introduced.
3.1 Adversarial Noises:

Adversarial noises are generated by various adversarial attack algorithms. The accuracy of the generated adversarial noise is used to evaluate the robustness of the tested model, which is expressed as Eq. 5:

$$R_A \left( w^* (\alpha), \alpha, x_i^{(\text{adv})} \right),$$

where $x_i^{(\text{adv})}$ is the adversarial noise generated by adversarial attack algorithms such as FGSM and PGD. In detail, the value of $R_A$ is calculated as described in Eq. 12.

3.1.2 Natural Noises:

Unlike adversarial noises generated by gradient-based adversarial attack, natural noises do not have the limitation of predefined magnitude, which aim to make the final image look natural and meanwhile fool the models. On the one hand, natural noises could exist naturally in the real world, as presented in [Hendrycks et al., 2021b]. On the other hand, natural noises could be manually generated by style transfer techniques. Since existing natural noise dataset such as ImageNet-C are created for more complex classification tasks, we use the style transfer technique [Johnson et al., 2016] to create natural noise for the CIFAR10 dataset. In detail, a style image is randomly selected from the COCO dataset [Lin et al., 2014], and other images are adopted as the content images. Based on the content and style image, a style transfer transformer is trained. Then we use the transformer to transform the original CIFAR10 dataset to the natural noise dataset. The accuracy of the tested model on the natural noises is expressed as Eq. 6:

$$R_N \left( w^* (\alpha), \alpha, x_i^{(\text{nat})} \right),$$

where $x_i^{(\text{nat})}$ is the natural noise sample.
3.1.3 System Noises:

System noises refer to the inaccuracy inherent to a system due to the inconsistent implementations of decoding and resize, such as ImageNet-S [Wang et al. (2021b); Tang et al. (2021)]. The system noise adapted for the CIFAR10 dataset is created in a similar way in our framework. In detail, we utilize the Pillow package for image decoding and resize. The images in the CIFAR10 dataset are resized as $32 \times 32$ after being decoded to $16 \times 16$. The accuracy of the tested model on the system noise is expressed as Eq. 7.

$$R_S \left( w^* (\alpha), \alpha, x_{i}^{(sys)} \right),$$

(7)

where $x_{i}^{(sys)}$ is the system noise sample.

3.1.4 Quantified metric:

Apart from the three above game-based robustness evaluation metrics, we also include the quantified metric such as the Frobenius ($F$) norms of Jacobian matrix of the input data, which does not rely on adversarial noise generated by various adversarial attacks, resulting in an easy combination with differentiable neural architecture search methods [Hosseini et al. (2021); Mok et al. (2021)]. The $F$ norms of Jacobian matrix is calculated as Eq. 8.

$$R_{J} \left( w^* (\alpha), \alpha, x_{i} \right) = \| J(x_{i}) \|_{F}^{2},$$

(8)

where $J$ denotes the Jacobian matrix of the output vector of 10 classifier with respect to input data. $i$ represents the number of input sample. Similarly, the calculation of the $F$ norms of Hessian matrix can be seen in Mok et al. (2021).

3.2 Differentiable NAS for Robust Network Architecture

In the model search space, we follow the adopted popularly cell-based structure, which consists of various operations, including none, max_pool_3x3, avg_pool_3x3, skip_connect, sep_conv_3x3, sep_conv_5x5, dil_conv_3x3 and dil_conv_5x5. The differentiable method is conducted to search for the near-optimal configuration of these eight operations. In the searching phase, the supernet consisting of all of the operations is constructed and trained. Based on the strategies of updating the architecture parameters, the near-optimal architecture on the supernet is obtained. In the retraining phase, the searched architecture is retrained. To realize the process of differentiable architecture search considering different robustness evaluation metrics, we design the total loss according to the characteristic of each type of evaluation.

In the adversarial noise, different from using the robust accuracy of adversarial attacks as the evaluation in the non-differentiable search strategy, we directly combine the adversarial attack into the total loss of the differentiable search strategy. The total loss considering adversarial accuracy is expressed as Eq. 9.

$$L_{total} = L (w^* (\alpha), \alpha; D_{val}) + \gamma L (w^* (\alpha), \alpha; D_{adv}),$$

(9)

where $\gamma$ is the coefficient to control the balance of clean accuracy and robust accuracy of the model. $D_{adv}$ is the adversarial example of $D_{val}$, which is generated by the specific adversarial attack. $L$ is the loss function of the cross entropy.

In the system noise and natural noise, we directly add them into the original image dataset. The mixture dataset is denoted as $D_{mix,val}$. The total loss function considering the system noise or natural noise is expressed as Eq. 10.

$$L_{total} = L (w^* (\alpha), \alpha; D_{mix,val}).$$

(10)

In the quantified metrics, we just directly consider them as the regularizer of the original loss. Take the Jacobian matrix as an example, the total loss is expressed as Eq. 11.

$$L_{total} = L (w^* (\alpha), \alpha; D_{val}) + \gamma R_{J} \left( w^* (\alpha), \alpha, x_{i} \right).$$

(11)

Three types of above total losses are substituted into Eq. 4 respectively to realize the purpose of automatically searching for robust architectures.
3.3 Searching for Attack Scheme

In this section, in adversarial noises, the LCAA is developed to provide more reliable robustness evaluation in searching for robust architectures, which aims to generate stronger noises to evaluate the robustness of the model. Inspired by the composite adversarial attack [Mao et al., 2021], we propose the composite adversarial attack with large search space. In their original work, the search space of attack magnitude and optimization step are discretized into 8 uniform parts, and then the search strategy is performed to find the efficient attack schemes. We enlarge the search space, where the attack magnitude is set as the continuous values and the step of attacker is set as the consecutive inters. The illustration of the comparison of CAA and LCAA can be seen in Fig 3.

![Figure 3: The illustration of attack magnitude and step in LCAA.](image)

In LCAA, the search space of attack operations under the infinite norm $S_{\infty}$ is composed of MI-Attack, MT-Attack, PGD, Identity-Attack, CW and FGSM [Mao et al., 2021], which are called the attacker operations. The attack perturbation is set to the continuous value from 0 to the predefined maximum magnitude $\epsilon$. The optimization step is set to the random inter from 0 to the predefined maximum step size $t$. The aim of CAA and LCAA are to search for the near-optimal order and configuration of attacker operations, the value of attack magnitude and the optimization step. An attack sequence example is expressed as [(‘CW’, $\epsilon =0.5$, t=50), (‘FGSM’, $\epsilon =0.4375$, t=125), (‘MI-Attack’, $\epsilon =0.2355$, t=25)].

4 Experiments

To verify the necessities and effectiveness of our proposed AAAD, we conduct experiments on CIFA10 dataset for image classification task.

4.1 Experiment Protocol

**In the phase of searching for architectures:** In following sections, we select three NAS methods including DARTS, PC-DARTS and random search. The supernets constructed by three methods are trained using the identical setting. The search epoch is set to 50. The batch sizes of data is set to 64. The initial channel is set to 16. The layer of the total network is set to 8. The original samples of 50,000 in the training data are spilted halfly for updating the weight parameters and architecture parameters respectively. For updating the weight parameters, the SGD optimizer with the initial learning rate of 0.1 and the weight decay factor of $3 \times 10^{-3}$ is adopted, which is annealed to zero through cosine scheduling. For updating the architecture parameters, the Adam optimizer with the learning rate of $6 \times 10^{-4}$ and the weight decay factor of $1 \times 10^{-3}$. As for the coefficients $\gamma$ of all experiments in the total loss are all set to 1. In addition, both of the percentages of natural noise and system noise are set to 0.5.

**In the phase of retraining the searched architectures:** The searched cell would be repeated 20 times to form a whole network. The initial channel is set to 36, and the total layer is set to 20. For the fair evaluation of each architecture’s intrinsic robustness, all the tested architectures are trained for 50 epochs under the identical training settings. The SGD optimizer with the initial learning rate of 0.025 and the weight decay factor of $3 \times 10^{-4}$ is adopted.

**Robustness evaluation:** The robustness of the trained model is evaluated using four types of metrics. In the adversarial noise, the attack magnitude of all attack algorithms is set to 0.01. The robust accuracy of the trained model under adversarial attack is calculated as Eq. [12]

$$Robust\ Accuracy = \frac{n_{t,\ adv}}{n_{t,\ total}},$$

where $n_{t,\ total}$ stands for the number of test samples that are predicted rightly by the classifier model. $n_{t,\ adv}$ denotes the number of adversarial example that are predicted rightly. The number of the test samples $n_{t,\ total}$ is set to 500. In the system and natural noise, the classification accuracy is calculated based on 500 samples respectively. In the quantified metric, we just implement the Jacobian matrix as this type of evaluation. 500 input samples are used to calculate the mean value of the $F$ norm of Jacobian matrix. The smaller the $F$ norm of Jacobian matrix means the more robust architectures.
Searching for composite adversarial attack: In the process of searching the efficient composite adversarial attack, we limit the length of attacker sequence to 3. The max step of each attack operation is set to 50. To accelerate the evaluation of the attack success rate, the number of test samples are set to 200.

The above experiments are conducted under the same environment using a single NVIDIA GTX 3090 Ti GPU. The visualizations of the searched model architectures and composite adversarial attack are presented in Appendix.

4.2 The performance of different robustness evaluations on NAS methods

To investigate the performance of different robustness evaluation metrics, we conduct experiments using three NAS methods, including DARTS [Liu et al. (2018)], PC-DARTS [Xu et al. (2019)] and NASP [Yao et al. (2020)]. The searched models using four types of evaluations are denoted as DARTS-FGSM, DARTS-Jacobian, DARTS-System and DARTS-Natural respectively. DARTS-Jacobian is implemented by us using the method in Hosseini et al. [2021]. The searched models just considering clean accuracy by three NAS methods are denoted as DARTS-Clean, PC-DARTS-Clean and NASP-Clean. All of the NAS methods are implemented using the fair search space. DARTS-Clean, PC-DARTS-Clean, NASP-Clean and the searched models using the Jacobian matrix are set as our baselines. To evaluate the robustness of the searched model in adversarial noise, we select four types of popular attack algorithms, including PGD, CW, FGSM and DDNL2Attack. The optimization step size of PGD is set to 3 and that of CW is set to 5. The statistical result is illustrated in Table 1. In other types of robustness evaluation, the comparisons of the searched models using three NAS methods are presented in Fig 4, Fig 5 and Fig 6 respectively.

![Figure 4: The robustness performance of DARTS with different types of evaluations.](image)

From Table 1 it could be seen that all of NAS methods could find more robust network when considering robustness metrics. Take DARTS-Clean as an example, the robustness accuracy under PGD attack is 39.4%. The robustness accuracy of the searched model using four types of evaluations can reach to 40.2%, 39.8%, 42.6% and 44.4% respectively. The improvement of the robustness of models is also similar in other attack algorithms. Compared with existing NAS method for robust architectures using Jacobian matrix, our proposed framework using three other types of evaluation can find more robust architectures on three NAS methods, which have the improvement of 4.2%, 4% and 3% under PGD attack respectively at most. From Fig 4, Fig 5 and Fig 6 in quantified metrics, we could see that even though the DARTS-Jacobian, PC-DARTS-Jacobian and NASP-Jacobian have smaller $R_J$, the accuracy under the natural and system noise is not competitive. In contrast, the searched model considering the accuracy of the system and natural noise are more robust than Jacobian matrix under the comprehensive evaluation. Take DARTS as an example, the accuracy of the searched model using Jacobian under the system noise is only 35.2%, while that of both of DARTS-System and DARTS-Natural can reach 37.2%. Similarly, there is also an improvement of about 7% in the natural noise. The improvement of the robustness under multiple types of evaluation can also be seen in PC-DARTS and NASP methods, which illustrates the effectiveness of our proposed method.

4.3 The performance of composite adversarial attack with large search space

To illustrate the effectiveness of LCAA, we conduct experiments under $l_{\infty}$ attack. The maximum magnitude is set to 0.01 and 0.031. The threat model architecture is randomly selected from the model search space. We use the components of the attacker search space, including PGD, MT-Attack, CW, MI-Attack, EnsembleAttack and CAA to make the comparison with LCAA. In each single adversarial attack, the attack steps of them are set as 150. In EnsembleAttack, we ensemble the PGD, MT-Attack and CW together, and the steps of each attack are set to 50. CAA
Searching for Robust Neural Architectures via Comprehensive and Reliable Evaluation  

Figure 5: The robustness performance of PC-DARTS with different types of evaluations.

Figure 6: The robustness performance of NASP with different types of evaluations.

Table 1: The robustness accuracy of the searched network architectures.

| Models                  | Clean | PGD | CW | FGSM | DDNL2Attack |
|-------------------------|-------|-----|----|------|-------------|
| DARTS-Clean Liu et al. (2018) | 94.52 | 39.4 | 35.8 | 74.8 | 57.0        |
| DARTS-Jacobian Hosseini et al. (2021) | 91.41 | 40.2 | 39.8 | 78.8 | **67.0**    |
| DARTS-FGSM (Ours)       | 94.17 | 39.8 | 38.8 | 77.8 | 60.0*       |
| DARTS-System (Ours)    | 93.63 | 42.6* | 41.8* | **81.0** | 57.2       |
| DARTS-Natural (Ours)   | 91.74 | **44.4** | **43.2** | 80.6* | 58.0        |
| PC-DARTS-Clean Xu et al. (2019) | 94.34 | 39.2 | 37.2 | 76.8 | 59.4        |
| PC-DARTS-Jacobian      | 93.28 | 39.6 | 38.8 | 77.8 | **63.8**    |
| PC-DARTS-FGSM (Ours)   | 94.34 | 41.0 | 39.2 | 79.4* | 60.8        |
| PC-DARTS-System (Ours) | 92.76 | 42.4* | 41.6* | 79.0 | 62.6*       |
| PC-DARTS-Natural(Ours) | 93.63 | **43.6** | **42.0** | **80.6** | 59.0       |
| NASP-Clean Yao et al. (2020) | 94.16 | 38.6 | 36.6 | 76.6 | **60.8**    |
| NASP-Jacobian           | 92.42 | 41.6 | 40.8 | 81.0* | 59.8        |
| NASP-FGSM (Ours)       | 92.42 | 43.6 | 43.2 | 81.0* | **60.8**    |
| NASP-System (Ours)     | 93.80 | **44.2** | **44.2** | 79.0 | 59.6        |
| NASP-Natural (Ours)    | 92.08 | **44.6** | **44.2** | **82.2** | 57.8        |

and LCAA are the best searched attacker with highest attacking success rate under random search with 80 individuals. The attack success rate and the number of gradient calculation of these attack algorithms are presented in Table 2.

From Table 2, it could be seen that both of CAA and LCAA could seek more stronger attacker configuration compared with the single attack operation. Besides, the gradient calculation numbers of them are less. Under the same search strategy, LCAA could be better than CAA. The attack success rate could be higher with fewer gradient calculation.
Table 2: The performance of LCAA compared with other attack schemes.

| Attacker  | PGD  | MT-Attack | CW   | MI-Attack Ensemble | Attack | CAA   | LCAA |
|----------|------|-----------|------|--------------------|--------|-------|------|
| $\ell_{\infty}$-0.01 | 55.5/150 | 55.2/150 | 54.9/150 | 55.5/150 | 55.5/150 | 66.5/20 | 75.0/55 |
| $\ell_{\infty}$-0.031 | 82.1/150 | 82.5/150 | 81.2/150 | 82.5/150 | 82.5/150 | 93.2/22 | 99.4/44 |

In a/b, a stands for the attack success rate, and b represents the number of gradient calculation.

4.4 The performance of LCAA evaluation on NAS methods

We further conduct the experiments to illustrate the performance of LCAA evaluation on NAS methods. We use FGSM, PGD and the searched attack scheme as the robustness evaluation. Then we use DARTS, PC-DARTS and NASP to search for the robust architecture. In our experiments, we observe the scene that too many max_pool operations emerge when considering the adversarial attack, which causes the poor performance of the searched model. Thus we adopt the early stopping strategy to limit the number of max_pool not more than 3. The statistical result of the robust accuracy of the searched model by NAS is presented in Table 3, Fig 7, Fig 8 and Fig 9.

Table 3: The robust accuracy of the searched architectures considering different adversarial attacks.

| Models          | Clean | PGD | CW | FGSM | DDNL2 | Attack |
|-----------------|-------|-----|----|------|-------|--------|
| DARTS-FGSM      | 94.17 | 39.8| 38.8| 77.8 | 60.0  |        |
| DARTS-PGD       | 93.11 | 39.8| 38.8| 80.0 | 63.8  |        |
| DARTS-LCAA      | 88.65 | 42.2| 41.4| 81.0 | 62.2  |        |
| PC-DARTS-FGSM   | 94.34 | 41.0| 39.2| 79.4 | 60.8  |        |
| PC-DARTS-PGD    | 92.59 | 43.8| 42.8| 81.2 | 58.8  |        |
| PC-DARTS-LCAA   | 91.91 | 47.8| 47.2| 81.4 | 57.2  |        |
| NASP-FGSM       | 92.42 | 43.6| 43.2| 81.0 | 60.8  |        |
| NASP-PGD        | 92.93 | 43.4| 43.2| 80.4 | 60.6  |        |
| NASP-LCAA       | 92.76 | 44.8| 44.6| 79.6 | 59.4  |        |

From Table 3, Fig 7, Fig 8 and Fig 9, we could see that the adversarial attack noise could influence the search performance greatly. In three NAS methods, the searched models considering FGSM or PGD are not robust enough. While the composite adversarial attack algorithms are included, we can find more robust architectures in multiple types of noises. Particularly in PC-DARTS, the robust accuracy of PC-DARTS-LCAA is significantly higher than PC-DARTS-PGD and PC-DARTS-FGSM, which has the improvement of 6.8% and 4% under PGD attack respectively. In other types of evaluation, the searched models by LCAA outperform that by FGSM and Jacobian matrix in almost all of NAS methods, the accuracy of which on the system and natural noise can be improved about 7%.

5 Conclusion

In this paper, we propose a framework called auto adversarial attack and defense. In this framework, to alleviate the uncomprehensive robustness evaluation in NAS, four types of robustness evaluation, including adversarial noise,
system noise, natural noise and quantified metric are incorporated. In addition, to address the unreliable evaluation in adversarial noise, we use the searched efficient composite adversarial attack with large search space as the metric. Experiments on CIFAR10 dataset show that comprehensive and reliable evaluation could help search more robust architecture in multiple NAS methods.

References

Boopathy, A., Weng, T., Chen, P., Liu, S., Daniel, L., 2019. Cnn-cert: An efficient framework for certifying robustness of convolutional neural networks, in: The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019, AAAI Press. pp. 3240–3247. URL: https://doi.org/10.1609/aaai.v33i01.33013240.

Cai, Q., Qian, Y., Zhou, S., Li, J., Yang, Y., Wu, F., Zhang, D., 2022. AVLSM: adaptive variational level set model for image segmentation in the presence of severe intensity inhomogeneity and high noise. IEEE Trans. Image Process. 31, 43–57. URL: https://doi.org/10.1109/TIP.2021.3127848, doi:10.1109/TIP.2021.3127848.

Carlini, N., Wagner, D., 2017. Towards evaluating the robustness of neural networks, in: 2017 ieee symposium on security and privacy (sp), IEEE. pp. 39–57.

Chen, H., Zhang, B., Xue, S., Gong, X., Liu, H., Ji, R., Doermann, D., 2020. Anti-bandit neural architecture search for model defense, in: European Conference on Computer Vision. Springer. pp. 70–85.

Chen, R., Li, J., Yan, J., Li, P., Sheng, B., 2021. Input-specific robustness certification for randomized smoothing. CoRR abs/2112.12084. URL: https://arxiv.org/abs/2112.12084, arXiv:2112.12084.

Figure 8: The robustness performance of PC-DARTS with different adversarial attacks.

Figure 9: The robustness performance of NASP with different adversarial attacks.
Chen, S., Wei, Q., Wang, W., Tang, J., Luo, B., Wang, Z., 2022. Remote sensing scene classification via multi-branch local attention network. IEEE Trans. Image Process. 31, 99–109. URL: [https://doi.org/10.1109/TIP.2021.3127851](https://doi.org/10.1109/TIP.2021.3127851)

Dong, Y., Liao, F., Pang, T., Su, H., Zhu, J., Hu, X., Li, J., 2018. Boosting adversarial attacks with momentum, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 9185–9193.

Engstrom, L., Tran, B., Tsipras, D., Schmidt, L., Madry, A., 2019. Exploring the landscape of spatial robustness, in: International Conference on Machine Learning, PMLR. pp. 1802–1811.

Goodfellow, I.J., Shlens, J., Szegedy, C., 2015. Explaining and harnessing adversarial examples, in: 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings.

Gowal, S., Uesato, J., Qin, C., Huang, P.S., Mann, T., Kohli, P., 2019. An alternative surrogate loss for pgd-based adversarial testing. arXiv preprint arXiv:1910.09338.

Guo, M., Yang, Y., Xu, R., Liu, Z., Lin, D., 2020. When NAS meets robustness: In search of robust architectures against adversarial attacks, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020, Computer Vision Foundation / IEEE. pp. 628–637. doi: [10.1109/CVPR42600.2020.00071](https://doi.org/10.1109/CVPR42600.2020.00071).

Hendrycks, D., Zhao, K., Basart, S., Steinhardt, J., Song, D., 2021a. Natural adversarial examples, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021, Computer Vision Foundation / IEEE. pp. 15262–15271.

Hendrycks, D., Zhao, K., Basart, S., Steinhardt, J., Song, D., 2021b. Natural adversarial examples, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 15262–15271.

Hosseini, R., Yang, X., Xie, P., 2021. DSRNA: differentiable search of robust neural architectures, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021, Computer Vision Foundation / IEEE. pp. 6196–6205.

Johnson, J., Alahi, A., Fei-Fei, L., 2016. Perceptual losses for real-time style transfer and super-resolution, in: European conference on computer vision, Springer. pp. 694–711.

Lin, J., Gan, C., Han, S., 2019. Defensive quantization: When efficiency meets robustness, in: 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019, OpenReview.net.

Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L., 2014. Microsoft coco: Common objects in context, in: European conference on computer vision, Springer. pp. 740–755.

Liu, H., Simonyan, K., Yang, Y., 2018. Darts: Differentiable architecture search. arXiv preprint arXiv:1806.09055.

Liu, J., Jin, Y., 2021a. Multi-objective search of robust neural architectures against multiple types of adversarial attacks. Neurocomputing 453, 73–84. URL: [https://doi.org/10.1016/j.neucom.2021.04.111](https://doi.org/10.1016/j.neucom.2021.04.111)

Liu, J., Jin, Y., 2021b. Multi-objective search of robust neural architectures against multiple types of adversarial attacks. Neurocomputing 453, 73–84.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A., 2018. Towards deep learning models resistant to adversarial attacks, in: 6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings, OpenReview.net.

Mao, X., Chen, Y., Wang, S., Su, H., He, Y., Xue, H., 2021. Composite adversarial attacks, in: Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, AAAI Press. pp. 8884–8892.

Mok, J., Na, B., Choe, H., Yoon, S., 2021. Advrush: Searching for adversarially robust neural architectures. CoRR abs/2108.01289. URL: [https://arxiv.org/abs/2108.01289](https://arxiv.org/abs/2108.01289)

Pang, T., Xu, K., Du, C., Chen, N., Zhu, J., 2019. Improving adversarial robustness via promoting ensemble diversity, in: Chaudhuri, K., Salakhutdinov, R. (Eds.), Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA, PMLR. pp. 4970–4979.

Rony, J., Hafemann, L.G., Oliveira, L.S., Ayed, I.B., Sabourin, R., Granger, E., 2019. Decoupling direction and norm for efficient gradient-based l2 adversarial attacks and defenses, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4322–4330.

Song, Y., Kim, T., Nowozin, S., Ermon, S., Kushman, N., 2018. Pixeldefend: Leveraging generative models to understand and defend against adversarial examples, in: 6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings, OpenReview.net.
Tang, S., Gong, R., Wang, Y., Liu, A., Wang, J., Chen, X., Yu, F., Liu, X., Song, D., Yuille, A.L., Torr, P.H.S., Tao, D., 2021. Robustart: Benchmarking robustness on architecture design and training techniques. CoRR abs/2109.05211. URL: https://arxiv.org/abs/2109.05211 [arXiv:2109.05211].

Vargas, D.V., Kotyan, S., IIIT-NR, S., 2019. Evolving robust neural architectures to defend from adversarial attacks. arXiv preprint arXiv:1906.11667.

Wang, Y., Li, Y., Gong, R., Xiao, T., Yu, F., 2021a. Real world robustness from systematic noise, in: Song, D., Tao, D., Yuille, A.L., Anandkumar, A., Liu, A., Chen, X., Li, Y., Xiao, C., Yang, X., Liu, X. (Eds.), ADVM ’21: Proceedings of the 1st International Workshop on Adversarial Learning for Multimedia, Virtual Event, China, 20 October 2021, ACM. pp. 42–48. URL: https://doi.org/10.1145/3475724.3483607, doi: 10.1145/3475724.3483607.

Wang, Y., Li, Y., Gong, R., Xiao, T., Yu, F., 2021b. Real world robustness from systematic noise, in: Proceedings of the 1st International Workshop on Adversarial Learning for Multimedia, pp. 42–48.

Xie, C., Wu, Y., van der Maaten, L., Yuille, A.L., He, K., 2019. Feature denoising for improving adversarial robustness, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019, Computer Vision Foundation / IEEE. pp. 501–509.

Xu, Y., Xie, L., Zhang, X., Chen, X., Qi, G.J., Tian, Q., Xiong, H., 2019. Pc-darts: Partial channel connections for memory-efficient architecture search. arXiv preprint arXiv:1907.05737.

Yao, Q., Xu, J., Tu, W.W., Zhu, Z., 2020. Efficient neural architecture search via proximal iterations, in: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 6664–6671.

Zhou, Q., Yu, C., Wang, Z., Qian, Q., Li, H., 2021. Instant-teaching: An end-to-end semi-supervised object detection framework, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021, Computer Vision Foundation / IEEE. pp. 4081–4090.