TRNAS: A Training-Free Robust Neural Architecture Search

Supplementary Material

1. Architecture Analysis of Searched DNNs

We further summarize and analyze the architecture in the paper.

- R-Score tends to select convolutional operations over pooling layers, which increases the number of learnable parameters.
- R-Score prefers depthwise separable convolutions over dilated convolutions, which enhances computational efficiency and generalization ability.
- R-Score tends to use smaller convolutional kernels, which helps reduce the model's sensitivity to small-scale perturbations.

2. Experiment Details of NAS-Rob-Bench-201

2.1. NAS-Rob-Bench-201

The design purpose of NAS-Rob-Bench-201 [15] is to quickly provide a test set from an architectural perspective to combat malicious attacks for the robust architecture design community. This benchmark contains 15,625 fully adversarially trained DNN architectures. We can transfer NAS algorithms to this benchmark to verify the algorithms' ability to search for robust architectures.

2.2. Analysis of Experimental Results

In NAS-Rob-Bench-201, we carefully verified the Benchmark and found slight differences in the performance of the optimal architecture (best) due to the minor errors introduced by the random seed [15].

As shown in Table 1, TRNAS is compared with existing SOTA NAS algorithms. * represents the original data extracted from NAS-Rob-Bench-201 without any modifications. We strictly follow the search process described for the comparison algorithms [2–4, 10, 13] and transfer them to NAS-Rob-Bench-201 for comparison. The top part describes the performance of the optimal architecture in the benchmark [15]. The middle part is the performance comparison analysis of standard NAS [9, 10, 13]. The lower part is the comparison analysis of robust NAS [2–4, 12].

We set the evaluation count of all NAS methods in NAS-Rob-Bench-201 to 1,000, whether for evolutionary or random search. In the best-case scenario, we only need 130 evaluations to find the best architecture. Our TRNAS first performs about 30 pruning evaluations using R-Score, eliminating obviously poor search space operations. For each edge, we delete a candidate operation. Subsequently, 100 architecture evaluations are conducted, allowing the evolutionary algorithm to search for the promising architecture in NAS-Rob-Bench-201. Therefore, regarding design efficiency for robust architectures, TRNAS significantly outperforms SOTA robust NAS algorithms, such as ZCPRob [2], CRoZe [3], and LRNAS [4]. Moreover, TRNAS also outperforms standard NAS algorithms such as SWAP [13], PADA [10], and DARTS [9].

Table 1. NAS-Rob-Bench-201

Method	Clean ACC	FGSM (3/255)	PGD (3/255)	FGSM (8/255)	PGD (8/255)
Best* [15]	79.4	69.8	69.2	53.7	48.2
Best [15]	79.6	69.7	69.2	53.5	48.1
Eigen* [16]	76.6	67.4	66.8	52.0	47.1
DARTS* [9]	33.2	28.6	28.5	21.5	21.3
NASI* [14]	66.6	57.1	56.7	41.0	37.9
PADA [10]	76.0	66.8	66.2	51.3	46.7
SWAP [13]	78.2	68.6	68.1	52.3	47.2
AdvRush* [12]	58.7	49.2	48.9	35.2	33.0
LRNAS [4]	72.9	63.7	63.2	48.5	44.3
CRoZe [3]	77.2	67.5	67.0	51.4	42.7
ZCPRob [2]	77.9	68.2	67.9	51.9	47.0
TRNAS	**79.6**	**69.7**	**69.2**	**53.5**	**48.1**

3. Experiment Details of DARTS

3.1. RobustBench

We use RobustBench (proposed by ZCPRob [2]) to evaluate the R-Score. RobustBench includes 223 adversarially trained network architectures and their adversarial accuracies. These architectures are randomly sampled from the DARTS [9] search space, which is widely recognized in robust NAS. Unlike standard NAS, collecting benchmark datasets for robust NAS is expensive. In addition, adversarial training takes over 10 × longer than standard training [2, 3]. Therefore, the limited number of architectures in RobustBench is normal. TRNAS adopts the state-of-the-art (SOTA) RobustBench as a benchmark to ensure a fair comparison.

3.2. Parameter Settings

In the search stage, we use a training-free evolutionary NAS method to search for robust DNNs. We measure the search cost based on the runtime of a single GPU. First, the search space consists of 8 stacked cells within the DARTS framework. Secondly, TRNAS performs 20 evolutionary updates. The population size for both the parent and offspring is set to 50. Subsequently, we used the R-score model to evaluate 100 architectures in the population. There are 20 iterations in total, so we need 2,000 (20 × 100) evaluations. However, there are 1,000 architectures that have already been

Table 2. Efficiency Analysis of Training-Free Robust Methods

Methods	All Architectures	Single	Effective Rate
CRoZe [3]	4075 s	2.04 s	79.30 %
ZCPRob [2]	9322 s	4.66 s	98.84 %
TRNAS	4845 s	2.42 s	100.0 %

evaluated, so only 1,000 evaluations are needed, about 0.02 GPU-days. The clustering size e of the multi-object selection (MOS) strategy during the search is 20. After completing the architecture search, the selected optimal architecture undergoes further adversarial training.

In the training stage, we follow the settings of previous robust NAS methods and use 7-step PGD [11] for adversarial training. At this stage, the network consists of 20 stacked cells. For CIFAR-10 and CIFAR-100 [7], we perform adversarial training for 120 epochs. The initial learning rate is set to 0.1 and decays to 0.01 at the 100th training epoch. The momentum is 0.9, and the weight decay is 1e-4. The batch size for the training data is set to 64. The total perturbation scale is 8/255, and SGD is used for network parameter optimization. For Tiny-ImageNet-200 [8], the training epoch is set to 90, the batch size is 26, and the initial number of channels is 64. Subsequently, the initial learning rate is 0.1, which decays to 0.01 and 0.001 at the 30th and 60th epochs, respectively. It is worth noting that the network robustness is usually best around 65th epoch from AdvRush framework [12].

In the evaluation stage, we use FGSM [5], PGD [11], and AutoAttack [1] for adversarial attacks to evaluate the trained architecture's adversarial robustness. The total perturbation range for the above attacks is set to 8/255. The single-step perturbation for FGSM [5] attacks is 8/255. In addition, the single-step perturbation for PGD attacks [11] is 2/255, and 20 or 100 steps are executed as needed. All experiments are run on a single NVIDIA RTX 4090 GPU and implemented using the PyTorch 2.0 framework.

3.3. The Search Efficiency of TRNAS

Although our R-Score proxy requires additional consumption, it achieves a 20× speedup over weight-sharing methods. In Table 2, under the same environment, the cost of evaluating 1,000 network architectures using R-Score is only 52% of that of ZCPRob [2]. In addition, CRoZe [3] can only predict 79.30% of the performance of architectures from RobustBench, while R-Score can predict each architecture. After multi-process optimization, our TRNAS method can be completed within 0.01 GPU days on a single 4090, achieving SOTA efficiency.

3.4. Performance on Tiny-ImageNet-200

The experimental results on Tiny-ImageNet-200 are show in Table. 3. TRNAS still outperforms other robust NAS methods in clean and robust scenarios.

Table 3. Comparison of SOTA NAS Methods on Tiny-ImageNet

Method	Params	Clean Acc	FGSM	PGD^{20}
RoBoT [6]	**6.47**	54.30	23.93	18.17
SWAP [13]	9.64	51.28	40.85	11.88
CRoZe [3]	13.03	54.28	22.09	17.28
ZCPRob [2]	7.37	52.83	32.81	15.72
TRNAS	9.37	**54.69**	**42.54**	**18.48**

3.5. Statistical analysis

We repeat the search experiment 20 times to ensure the reliability and stability of the experimental results. The experimental results show that TRNAS is the most robust and lightweight across all metrics. Table 4 details the comparative results of TRNAS with baseline methods CRoZe [3] and ZCPRob [2] across various metrics. In addition, the experimental data for CRoZe and ZCPRob are selected from their respective best performance in their papers, while the data of TRNAS represents the average and standard deviation of 20 experiments.

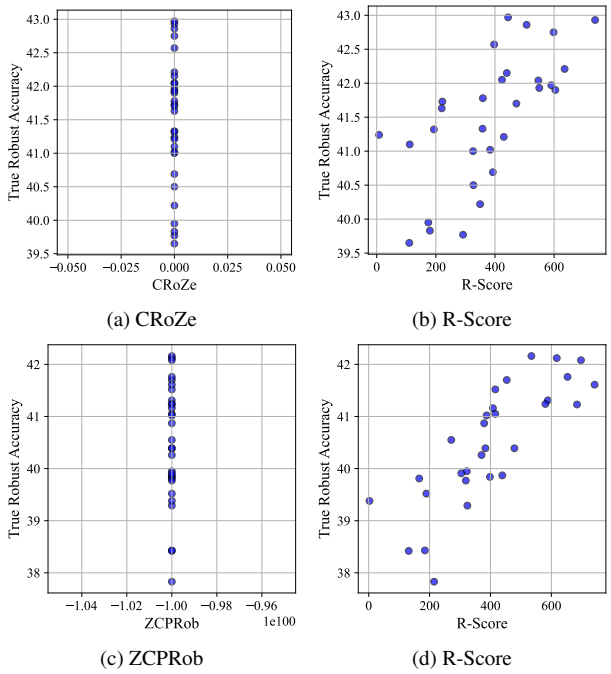
Table 4. The Statistical Result Analysis of SOTA NAS Methods

Methods	Params	Flops	Clean	FGSM	PGD^{20}
CRoZe [3]	5.5	841.00	83.30	58.47	52.63
ZCPRob [2]	3.4	555.54	85.60	60.20	52.75
TRNAS avg.	**3.39**	**549.84**	**85.94**	**60.66**	**52.96**
(± std)	±0.28	±38.91	±0.53	±0.53	±0.38

The average parameters of TRNAS are 3.39MB, lower than those of CRoZe and ZCPRob, demonstrating a set of lighter architectures. In terms of computational complexity, the average Flops of TRNAS is 549.84M, significantly lower than those of CRoZe and ZCPRob, indicating a clear advantage in computational efficiency. In performance metrics, the average performance of TRNAS in Clean, FGSM, and PGD test scenarios is 85.94% (±0.53), 60.66% (±0.53), and 52.96% (±0.38), respectively. These results are superior to those of CRoZe and ZCPRob, showing higher robustness and accuracy. Additionally, the smaller standard deviation of TRNAS across multiple metrics demonstrates stronger stability of performance.

3.6. Concise Theoretical Explanation of R-Score

The linear activation capability captures local perturbation sensitivity, while feature consistency reflects global stability inspired by Lipschitz continuity. Our R-Score combines both aspects, offering a more comprehensive robust-



(a) CRoZe (b) R-Score

(c) ZCPRob (d) R-Score

Figure 1. Prediction Errors of Different Proxies.

ness evaluation than ZCPRob [2] and CRoZe [3]. Figs. 1(a) and (c) show the prediction results of CRoZe and ZCPRob on some architectures from RobustBench [2]. Both fail to distinguish some architectures with significant performance differences, resulting in duplicated scores. In contrast, R-Score in Figs. 1(b) and (d) can more accurately distinguish architectural performance.

References

[1] Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International Conference on Machine Learning*, pages 2206–2216. PMLR, 2020. 2

[2] Feng et al. Zero-cost proxy for adversarial robustness evaluation. In *ICLR*, 2025. 1, 2, 3

[3] Ha et al. Generalizable lightweight proxy for robust NAS against diverse perturbations. *NIPS*, 36, 2024. 1, 2, 3

[4] Yuqi Feng, Zeqiong Lv, Hongyang Chen, Shangce Gao, Fengping An, and Yanan Sun. LRNAS: Differentiable searching for adversarially robust lightweight neural architecture. *IEEE Transactions on Neural Networks and Learning Systems*, 2024. 1

[5] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *stat*, 1050: 20, 2015. 2

[6] Zhenfeng He, Yao Shu, Zhongxiang Dai, and Bryan Kian Hsiang Low. Robustifying and boosting training-free neural architecture search. In *The Twelfth International Conference on Learning Representations*, 2024. 2

[7] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 2

[8] Yann Le and Xuan Yang. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7):3, 2015. 2

[9] Hanxiao Liu, Karen Simonyan, and Yiming Yang. DARTS: Differentiable architecture search. *International Conference on Learning Representations*, 2019. 1

[10] Shun Lu, Yu Hu, Longxing Yang, Zihao Sun, Jilin Mei, Jianchao Tan, and Chengru Song. PA&DA: Jointly sampling path and data for consistent nas. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11940–11949, 2023. 1

[11] Aleksander Mądry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *International Conference on Learning Representations*, 2017. 2

[12] Jisoo Mok, Byunggook Na, Hyeokjun Choe, and Sungroh Yoon. AdvRush: Searching for adversarially robust neural architectures. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12322–12332, 2021. 1, 2

[13] Yameng Peng, Andy Song, Haytham M Fayek, Vic Ciesielski, and Xiaojun Chang. SWAP-NAS: Sample-wise activation patterns for ultra-fast nas. In *The Twelfth International Conference on Learning Representations*, 2024. 1, 2

[14] Yao Shu, Shaofeng Cai, Zhongxiang Dai, Beng Chin Ooi, and Bryan Kian Hsiang Low. NASI]: Label-and data-agnostic neural architecture search at initialization. In *International Conference on Learning Representations*, 2021. 1

[15] Yongtao Wu, Fanghui Liu, Carl-Johann Simon-Gabriel, Grigorios G Chrysos, and Volkan Cevher. Robust nas under adversarial training: benchmark, theory, and beyond. *The Twelfth International Conference on Learning Representations*, 2024. 1

[16] Zhenyu Zhu, Fanghui Liu, Grigorios Chrysos, and Volkan Cevher. Generalization properties of nas under activation and skip connection search. *Advances in Neural Information Processing Systems*, 35:23551–23565, 2022. 1