Generalizable Lightweight Proxy for Robust NAS against Diverse Perturbations

Hyeonjeong Ha [*1] **Minseon Kim** [*1] **Sung Ju Hwang** [1 2]

Abstract

Recent neural architecture search (NAS) frameworks have been successful in finding optimal architectures for given conditions (e.g., performance or latency). However, they search for optimal architectures in terms of their performance on clean images only, while robustness against various types of perturbations or corruptions is crucial in practice. Although there exist several robust NAS frameworks that tackle this issue by integrating adversarial training into one-shot NAS, however, they are limited in that they only consider robustness against adversarial attacks and require significant computational resources to discover optimal architectures for a single task, which makes them impractical in real-world scenarios. To address these challenges, we propose a novel lightweight robust zero-cost proxy that considers the consistency across features, parameters, and gradients of both clean and perturbed images at the initialization state. Our approach facilitates an efficient and rapid search for neural architectures capable of learning generalizable features that exhibit robustness across diverse perturbations. The experimental results demonstrate that our proxy can rapidly and efficiently search for neural architectures that are consistently robust against various perturbations on multiple benchmark datasets and diverse search spaces, largely outperforming existing clean zero-shot NAS and robust NAS with reduced search cost.

[*]Equal contribution. Author ordering determined by coin flip. [1]Korea Advanced Institute of Science and Technology (KAIST) [2]DeepAuto, Republic of Korea. Correspondence to: Hyeonjeong Ha <hyeonjeongha@kaist.ac.kr>, Minseon Kim <minseonkim@kaist.ac.kr>, Sung Ju Hwang <sjhwang82@kaist.ac.kr>.

2nd AdvML Frontiers workshop at 40th International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

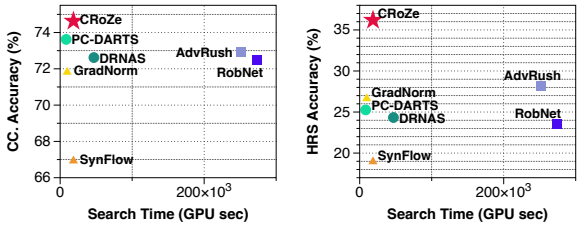


Figure 1: Final performance of the searched network in DARTS search space on CIFAR-10 through **clean one-shot NAS**, **robust NAS**, **clean zero-shot NAS** and **our CRoZe**.

1. Introduction

Neural architecture search (NAS) techniques have achieved remarkable success in optimizing neural networks for given tasks and constraints, yielding networks that outperform handcrafted neural architectures (Baker et al., 2017; Liu et al., 2018a; Luo et al., 2018; Pham et al., 2018; Xu et al., 2020). However, previous NAS approaches have primarily aimed to search for architectures with optimal performance and efficiency on clean inputs, while paying less attention to robustness against adversarial perturbations (Goodfellow et al., 2015; Madry et al., 2018) or common types of corruptions (Hendrycks & Dietterich, 2019). This can result in finding unsafe and vulnerable architectures with erroneous and high-confidence predictions on input examples even with small perturbations (Mok et al., 2021; Jung et al., 2023), limiting the practical deployment of NAS in real-world safety-critical applications.

To address the gap between robustness and NAS, previous robust NAS works (Mok et al., 2021; Guo et al., 2020) have proposed to search for adversarially robust architectures by integrating adversarial training into NAS. Yet, they are computationally inefficient as they utilize costly adversarial training on top of the one-shot NAS methods (Liu et al., 2019; Cai et al., 2019), requiring up to 33× larger computational cost than clean one-shot NAS (Xu et al., 2020). Especially, Guo et al. (2020) takes almost 4 GPU days on NVIDIA 3090 RTX GPU to train the supernet, as it requires performing adversarial training on subnets with perturbed examples (Figure 1, RobNet). Furthermore, they only target a single type of perturbation, i.e., adversarial perturbation (Goodfellow et al., 2015; Madry et al., 2018), thus, failing to generalize to diverse perturbations. In order to deploy NAS to real-world applications that require han-

dling diverse types of tasks and perturbations, we need a lightweight NAS method that can yield robust architectures without going over such costly processes.

To tackle this challenge, we propose a novel and lightweight **C**onsistency-based **Ro**bust **Ze**ro-cost proxy (**CRoZe**) that can rapidly evaluate the robustness of the neural architectures against *diverse semantic-preserving* perturbations without requiring any iterative training. While prior clean zero-shot NAS methods (Abdelfattah et al., 2021; Mellor et al., 2021) introduced proxies that score the networks with randomly initialized parameters (Lee et al., 2019; Wang et al., 2020; Liu et al., 2021; Tanaka et al., 2020) without any training, they only consider which parameters are highly sensitive to clean inputs for a given task, as determined by measuring the scale of the gradients based on the objectives and thus yield networks that are vulnerable against perturbed inputs (Figure 2a).

Specifically, our proxy captures the consistency across the features, parameters, and gradients of a randomly initialized model for both clean and perturbed inputs, which is updated with a single gradient step (Figure 2b). This metric design measures the model's robustness in multiple aspects, which is indicative of its generalized robustness to diverse types of perturbations. This prevents the metric from being biased toward a specific type of perturbation and ensures its robustness across diverse semantic-preserving perturbations. Empirically, we find that a neural architecture with the highest performance for a single type of perturbation tends to exhibit larger feature variance for other types of perturbations (Supplementary Figure 4), while our proxy that considers the robustness in multiple aspects obtains features with smaller variance even on diverse types of perturbations. This suggests that our proxy is able to effectively discover architectures with enhanced generalized robustness.

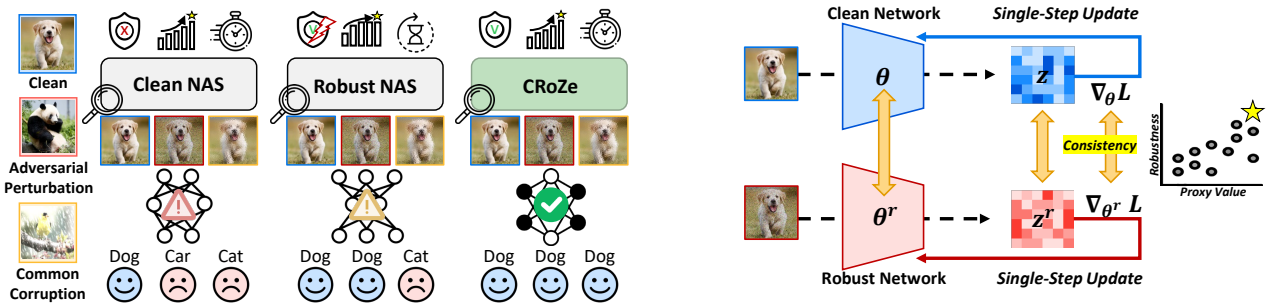
We validate our approach through extensive experiments on diverse search spaces (NAS-Bench 201, DARTS) and multiple datasets (CIFAR-10, CIFAR-100, ImageNet16-120), with not only the adversarial perturbations but also with various types of common corruptions (Hendrycks & Dietterich, 2019), against both clean zero-shot NAS (Abdelfattah et al., 2021; Mellor et al., 2021) and robust NAS baselines (Mok et al., 2021; Guo et al., 2020). The experimental results clearly demonstrate our effectiveness in finding generalizable robust neural architectures. Our contributions can be summarized as follows:

- We propose a simple yet effective consistency-based zero-cost proxy for robust NAS against diverse perturbations via measuring the consistency of features, parameters, and gradients between perturbed and clean samples.
- Our approach can rapidly search for generalizable neural architectures that do not perform well only on clean samples but also are highly robust against diverse types of perturbations on various datasets.
- Our proxy obtains superior Spearman's ρ across benchmarks compared to clean zero-shot NAS methods and identifies robust architectures that exceed robust NAS frameworks by **6.21%** with **14.7 times less search cost** within the DARTS search space on CIFAR-10.

2. Related Work

Robustness of DNNs against Perturbations. Despite the recent advances, deep neural networks (DNNs) are still vulnerable to small perturbations on the input, e.g., common corruptions (Hendrycks & Dietterich, 2019), random noises (Dodge & Karam, 2017), and adversarial perturbations (Biggio et al., 2013; Szegedy et al., 2014), which can result in incorrect predictions with high confidence. To overcome such vulnerability against diverse perturbations, many approaches have been proposed to train the neural network to be robust against each type of perturbation individually. To learn a rich visual representation from limited crawled data, previous works (Hendrycks et al., 2020; Cubuk et al., 2020) utilized a combination of strong data augmentation functions to improve robustness to common corruptions and random Gaussian noises. Furthermore, to overcome adversarial vulnerability, widely used defense mechanisms (Goodfellow et al., 2015; Madry et al., 2018; Moosavi-Dezfooli et al., 2016) generate adversarially perturbed images by taking multiple gradient steps to maximize the training loss and use them in training to improve the model's robustness.

Neural Architecture Search. Neural architecture search (NAS) leverages reinforcement learning (Zoph & Le, 2017; Baker et al., 2017; Zhong et al., 2018) or evolutionary algorithms (Real et al., 2017; Liu et al., 2018b; Elsken et al., 2019; Real et al., 2019) to automate the design of optimal architectures for specific tasks or devices. However, those are computationally intensive, making them impractical to be applied in real-world applications. To address this, zero-shot NAS methods (Abdelfattah et al., 2021; Mellor et al., 2021) have emerged that significantly reduce search costs by predicting the performance of architecture at the initialization state only with a single batch of a given dataset. Despite the improvement in NAS, previous zero-shot NAS methods, and conventional NAS methods aim only to find architectures with high accuracy on clean examples, without considering their robustness against various perturbations. In particular, SynFlow (Abdelfattah et al., 2021) lacks data incorporation in its scoring mechanism, potentially failing to find the network that can handle diverse perturbations. As a result, models found with previous NAS methods often lead to incorrect predictions with high confidence (Mok et al., 2021; Jung et al., 2023) even with small imperceptible perturbations applied to the inputs. A new class of NAS methods (Guo et al., 2020; Mok et al., 2021) that



(a) Clean NAS and Robust NAS vs CRoZe.

(b) Consistency-based zero-cost proxy (CRoZe).

Figure 2: **Generalizable lightweight proxy for robust NAS against diverse perturbations.** While previous NAS methods search neural architectures primarily on clean samples (Clean NAS) or adversarial perturbations (Robust NAS) with excessive search costs and fail to generalize across diverse perturbations, our proposed proxy, namely CRoZe, can rapidly search high-performing neural architectures against diverse perturbations. CRoZe evaluates the network's robustness in a single step based on the consistency across the features (z and z^r), parameters (θ and θ^r), and gradients ($\nabla_\theta \mathcal{L}$ and $\nabla_{\theta^r} \mathcal{L}$) between clean and robust network against clean and perturbed inputs.

considers robustness against adversarial perturbations has emerged. Yet, they require adversarial training of the supernet, which demands more computational resources than conventional NAS (Real et al., 2017; Elsken et al., 2019) due to repeated adversarial gradient steps. Thus, there is a need for a lightweight NAS approach that can achieve generalized robustness for safe real-world applications.

3. Methods

Our goal is to efficiently search for robust architectures that have high performance on various tasks, regardless of the type of perturbations applied to the input samples. To achieve this goal, we propose a **C**onsistency-based **Ro**bust **Ze**ro-cost proxy (**CRoZe**) that considers the consistency of the features, parameters, and gradients between a single batch of clean and perturbed inputs obtained by taking a single gradient step. CRoZe enables the rapid evaluation of the robustness of the neural architectures in the random states, without any adversarial training (Figure 2).

3.1. Robust Architectures

Formally, our goal is to accurately estimate the final robustness of a given neural architecture $\mathcal{A}$ with given a single batch of inputs $B = \{(x, y)\}$, without training. Here, $x \in X$ is the input sample, and $y \in Y$ is its corresponding label for given dataset $\mathcal{D} = \{X, Y\}$. In the following section, the network $f_\theta(\cdot)$ consists of an encoder $e_\psi(\cdot)$ and a linear layer $h_\pi(\cdot)$, which is $\mathcal{A}$ that is parameterized with ψ and π, respectively. The most straightforward approach to evaluate the robustness of the network is measuring the accuracy against the perturbed input x' with unseen semantic-preserving perturbations, as follows:

$$\text{Acc.} = \frac{1}{N}\sum_{n=1}^{N} \zeta(\arg\max_{c \in Y} \mathbb{P}(h_\pi \circ e_\psi(x') = c) = y), \quad (1)$$

where ζ is the Kronecker delta function, which returns 1 if the predicted class c is the same as y, and 0 otherwise, x' is a perturbed input, such as one with random Gaussian noise, common types of corruptions (Hendrycks et al., 2020), or adversarial perturbations (Madry et al., 2018; Goodfellow et al., 2015) applied to it. Specifically, to have a correct prediction on the unseen perturbed input x', the model needs to extract similar features between x' and x, assuming that the model can correctly predict the label for input x as follows:

$$\| e_\psi(x) - e_\psi(x') \| \leq \epsilon, \quad (2)$$

where ϵ is sufficiently small bound. Thus, a robust model is one that can extract consistent features across a wide range of perturbations. However, precisely assessing the accuracy of the model against perturbed inputs requires training from scratch with a full dataset, which incurs a linear increase in the computational cost with respect to the number of neural architectures to be evaluated.

3.2. Estimating Robust Network through Perturbation

In this section, we explain details on preliminary protocols before computing our proxy. Due to the impractical computation cost to obtain a combinatorial number of fully-trained models in a given neural architecture search space (i.e., 10^{19} for DARTS), we propose to utilize two surrogate networks which together can estimate the robustness of fully-trained networks within a single gradient step. The two surrogate networks are a clean network f_θ with the randomly initialized parameter θ and a robust network f_{θ^r} with the robust parameter θ^r, which is determined with a parameter perturbations from f_θ. Then, the obtained θ^r is used to generate the single batch of perturbed inputs for our proxy.

Robust Parameter Update via Layer-wise Parameter Perturbation. We employ a surrogate robust network to estimate the output of fully-trained networks against perturbed inputs. To make the perturbation stronger, we use a double-perturbation scheme that combines layer-wise parameter perturbations (Wu et al., 2020) and input perturbations, both of which maximize the training objectives $\mathcal{L}$. This layer-wise perturbation allows us to learn smoother up-

dated parameters by min-max optimization, through which we can obtain the model with the maximal possible generalization capability (Wu et al., 2020; Foret et al., 2021) within a single step. Specifically, given a network f is composed of M layers, $f_\theta = f_{\theta_M} \circ \cdots \circ f_{\theta_1}$, with parameters $\theta = \{\theta_1, \ldots, \theta_M\}$, the m^{th} layer-wise parameter perturbation is done as follows:

$$\theta_m^r \leftarrow \theta_m + \beta * \frac{\nabla_{\theta_m} \mathcal{L}\big(f_\theta(x), y\big)}{\|\nabla_{\theta_m} \mathcal{L}\big(f_\theta(x), y\big)\|} * \|\theta_m\|, \quad (3)$$

where β is the step size for parameter perturbations, $\|\cdot\|$ is the norm, and $\mathcal{L}$ is the cross-entropy objective. This bounds the size of the perturbation by the norm of the original parameter $\|\theta_m\|$.

Perturbed Input. On top of the perturbed parameters (Eq. 3), we generate perturbed input images by employing fast gradient sign method (FGSM) (Goodfellow et al., 2015), which is the worst case adversarial perturbation to the input x as follows:

$$\delta = \epsilon \texttt{sign}\Big(\nabla_x \mathcal{L}\big(f_{\theta^r}(x), y\big)\Big), \quad (4)$$

where δ is a generated adversarial perturbation that maximizes the cross-entropy objective $\mathcal{L}$ of given input x and given label y. Then, we utilize the perturbed inputs ($x' = x + \delta$) to estimate the robustness of the fully-trained model. Although CRoZe is an input perturbation-agnostic proxy (Table 5), we employ adversarially perturbed inputs for all the following sections.

3.3. Consistency-based Proxy for Robust NAS

We now elaborate the details on our proxy that evaluate the robustness of the architecture with the two surrogate networks: the clean network that is randomly initialized and uses clean images x as inputs, and the robust network parameterized with θ^r which uses perturbed images $x + \delta$ as inputs.

Features, Parameters, and Gradients. As we described in Section 3.1, we first evaluate the representational consistency between clean input (x) and perturbed input (x') by forwarding them through the encoder of clean surrogate network $f_\theta(\cdot)$ and robust surrogate network $f_{\theta^r}(\cdot)$, as follows:

$$\mathcal{Z}_m(f_\theta(x), f_{\theta^r}(x')) = 1 + \frac{z_m \cdot z_m^r}{\|z_m\| \|z_m^r\|}, \quad (5)$$

where z_m and z_m^r are output feature of each network $f_\theta(\cdot)$ and $f_{\theta^r}(\cdot)$, respectively, from each m^{th} layer. Especially, we measure layer-wise consistency with cosine similarity function between clean and robust features. The higher feature consistency infers the higher robustness of the network.

However, the proxy solely considering the feature consistency within a single batch can be heavily reliant on the selection of the batch. Therefore, to complement the feature consistency, we propose incorporating the consistency of updated parameters and gradient conflicts from each surrogate network as additional measures to evaluate the robustness of the network. To introduce these concepts, let us first denote the gradient and updated parameter of each surrogate network. The gradient g from the clean surrogate network f_θ and robust surrogate network f_{θ^r} against clean images x and perturbed images x', are obtained as follows:

$$g = \nabla_\theta \mathcal{L}\big(f_\theta(x), y\big), \quad g^r = \nabla_{\theta^r} \mathcal{L}\big(f_{\theta^r}(x'), y\big), \quad (6)$$

where g and g^r are the gradients with respect to cross-entropy objectives $\mathcal{L}$ of the clean images x and perturbed images $x' = x + \delta$, respectively. Then, we can acquire the single-step updated clean parameters θ and robust parameters θ^r calculated with gradients g and g^r and learning rate γ, respectively as follows:

$$\theta_1 \leftarrow \theta - \gamma g, \quad \theta_1^r \leftarrow \theta^r - \gamma g^r. \quad (7)$$

Since each surrogate network represents the model for each task, i.e., clean classification and perturbed classification, the parameters and gradients of each surrogate network correspond to the updated weights and convergence directions for each task. Thus, the network that has high robustness will exhibit identical or similar parameter spaces for both classification tasks. However, as acquiring parameters of a fully-trained network is impractical, we estimate the converged parameters with the single-step updated parameters θ_1 and θ_1^r. Accordingly, since the higher similarity of single-step updated parameters may promote the model to converge to an identical or similar parameter space for both tasks, we evaluate the parameter similarity as one of our proxy terms as follows:

$$\mathcal{P}_m(\theta_1, \theta_1^r) = 1 + \frac{\theta_{1,m} \cdot \theta_{1,m}^r}{\|\theta_{1,m}\| \|\theta_{1,m}^r\|}. \quad (8)$$

Furthermore, each gradient of the surrogate networks represents the converged direction of given objectives for each task, which is cross-entropy loss of clean input and perturbed input (Eq. 6). Thus, we employ the gradients similarity as an evaluation of the difficulties of optimizing architecture for both tasks. Therefore, when the gradient directions are highly aligned between the two tasks, the learning trajectory for both tasks becomes more predictable, facilitating the optimization of both tasks easily. In contrast, orthogonal gradient directions lead to greater unpredictability, hindering optimization and potentially resulting in suboptimality for both clean or perturbed classification tasks. Therefore, to evaluate the stability of optimizing both tasks, we measure the absolute value of gradient similarity as follows:

$$\mathcal{G}_m(g, g^r) = \left| \frac{g_m \cdot g_m^r}{\|g_m\| \|g_m^r\|} \right|. \quad (9)$$

Consistency-based Robust Zero-Cost Proxy: CRoZe. In sum, to evaluate the robustness of the given architecture, we propose a scoring mechanism that evaluates the

similarities of features, parameters, and gradients between the clean network f_θ and the robust network f_{θ^r} that are obtained with a single gradient update. Therefore, the robustness score for a given neural architecture is computed as follows:

$$\text{CRoZe}(x, x'; f_\theta, f_{\theta^r}) = \sum_{m=1}^{M} \mathcal{Z}_m \times \mathcal{P}_m \times \mathcal{G}_m. \quad (10)$$

That is, we score the network f_θ with a higher CRoZe score as more robust to perturbations. In the next section, we show that this measure is highly correlated with the actual robustness of a fully-trained model (Table 1a).

4. Experiments

We now experimentally validate our proxy designed to identify robust architectures that perform well on both clean and perturbed inputs, on multiple benchmarks. We especially demonstrate the effectiveness of CRoZe in terms of Spearman's ρ (Section 4.2) and computational efficiency with the final performance of the chosen architecture using our proxy (Section 4.3). Further additional analysis regarding our proxy is described in Supplementary B.2.

4.1. Experimental Setting

Datasets. For the NAS-Bench-201 (Dong & Yang, 2019; Jung et al., 2023) search space, we validate our proxy across different tasks (CIFAR-10, CIFAR-100, and ImageNet16-120) and perturbations (FGSM (Goodfellow et al., 2015), PGD (Madry et al., 2018), and 15 types of common corruptions (Hendrycks & Dietterich, 2019)). To measure Spearman's ρ between final accuracies and our proxy values, we use both clean NAS-Bench-201 (Dong & Yang, 2019) and robust NAS-Bench-201 (Jung et al., 2023), which include clean accuracies and robust accuracies. Finally, we search for the optimal architectures with our proxy in DARTS (Liu et al., 2019) search space and compare the final accuracies. More experimental details are in Supplementary A.

4.2. Results on NAS-Bench-201

Standard-Trained Neural Architectures. In order to verify the effectiveness of our proxy in searching for high-performing neural architectures across various tasks and perturbations, we conduct experiments using Spearman's ρ as a metric to evaluate the preservation of the rank between the proxy values and final accuracies (Abdelfattah et al., 2021; Mellor et al., 2021; Dong et al., 2023). For Spearman's ρ between clean accuracies and proxy values, existing clean zero-shot NAS works (Abdelfattah et al., 2021; Mellor et al., 2021) performed worse than using the number of parameters as a proxy (#Params.). In contrast, our proxy shows significantly higher correlations with clean accuracies across all tasks, demonstrating improvements of 10.2% and 9.31% on CIFAR-10 and CIFAR-100, respectively, compared to best-performing baselines (Table 1a, 2).

Furthermore, CRoZe shows remarkable Spearman's ρ for robust accuracies obtained against adversarial perturbations and corrupted noises across tasks. Notably, our proxy outperforms the SynFlow (Abdelfattah et al., 2021) by 6.73% and 9.05% in an average of Spearman's ρ for adversarial perturbations and common corruptions on CIFAR-10, respectively (Table 1a). Our results on multiple benchmark tasks with diverse perturbations highlight the ability of our proxy to effectively search for robust architectures that can make consistently outperform predictions against various perturbations. Importantly, our proxy is designed to prioritize generalizability, and as a result, it exhibits consistently enhanced correlation with final accuracies for both clean and perturbed samples. This result indicates that considering generalization ability is effective in identifying robust neural architectures against diverse perturbations but also leads to improved performance for clean neural architectures.

Adversarially-Trained Neural Architectures. We also validate the ability of our proxy to precisely predict the robustness of adversarially-trained networks, specifically for adversarial perturbations. Adversarial training (Madry et al., 2018) is a straightforward approach to achieve robustness in the presence of adversarial perturbations. To assess the Spearman's ρ of robustness in adversarially-trained networks, we construct a dataset consisting of final robust accuracies of 500 randomly sampled neural architectures from NAS-Bench-201 search space that are adversarially-trained (Madry et al., 2018) from scratch.

Considering the trade-off between the clean and robust accuracy in adversarial training (Zhang et al., 2019), we employ the harmonic robustness score (HRS) (Devaguptapu et al., 2021) to evaluate the overall performance of the adversarially-trained models. When comparing the correlations between clean performances and proxy values, existing clean zero-shot NAS approaches, i.e., SynFlow and Grasp, demonstrate higher correlations in the FGSM or PGD, but their correlations with clean performances are poorer than GradNorm and Fisher, respectively. This result shows that clean zero-shot NAS methods tend to search for architectures that are more prone to overfitting to either clean or robust tasks (Table 1b). In contrast, CRoZe consistently achieves higher Spearman's ρ for both clean and robust tasks, ultimately enabling the search for architecture with high HRS due to consideration of alignment in gradients.

4.3. End-to-End Generalization Performance on DARTS

In this section, we evaluate the effectiveness of CRoZe in rapidly searching for generalized neural architectures in the DARTS search space and compare it with previous clean one-shot NAS (Xu et al., 2020; Chen et al., 2020), clean zero-shot NAS (Tanaka et al., 2020), and robust NAS (Mok et al., 2021; Guo et al., 2020) in terms of performance and computational cost.

Table 1: Comparison of Spearman's ρ between the actual accuracies and the proxy values on CIFAR-10 in the NAS-Bench-201 search space. Plain, Grasp, Fisher, GradNorm, SynFlow are zero-cost methods from Abdelfattah et al. (2021). NASWOT (Mellor et al., 2021) is using activation as a proxy. Clean stands for clean accuracy and robust accuracies are evaluated against adversarial perturbations (Goodfellow et al., 2015) with various attack sizes (ϵ) and common corruptions (Hendrycks & Dietterich, 2019). Avg. stands for average Spearman's ρ values with all accuracies. **Bold** and underline stands for the best and second.

Proxy Type	Clean	Adversarial Perturbation			Common Corruption				
		$\epsilon = 8$	$\epsilon = 4$	$\epsilon = 2$	Weather	Noise	Blur	Digital	Avg.
FLOPs	0.726	0.753	0.740	0.716	0.665	0.138	0.219	0.473	0.554
#Params.	0.747	0.756	0.739	0.713	0.674	0.131	0.215	0.489	0.558
Plain	-0.073	-0.059	-0.055	-0.029	-0.041	0.048	0.035	-0.032	-0.026
Grasp	0.440	0.547	0.563	0.541	0.459	0.217	0.164	0.327	0.407
Fisher	0.356	0.457	0.491	0.498	0.407	0.217	0.221	0.240	0.361
GradNorm	0.598	0.750	0.766	0.743	0.641	**0.246**	**0.227**	0.423	0.549
SynFlow	0.737	0.778	0.750	0.727	0.673	0.188	0.165	0.554	0.572
NASWOT	0.660	0.511	0.507	0.435	0.466	-0.066	-0.005	0.413	0.365
CRoZe	**0.823**	**0.826**	**0.801**	**0.780**	**0.743**	0.190	0.224	**0.566**	**0.619**

(a) Standard-Trained

Proxy Type	Clean	FGSM	PGD	HRS(FGSM)	HRS(PGD)
FLOPs	0.670	0.330	0.418	0.531	0.515
#Params.	0.678	0.341	0.429	0.541	0.526
Plain	-0.042	-0.007	-0.012	-0.016	-0.016
Grasp	0.470	0.324	0.341	0.392	0.375
Fisher	0.482	0.226	0.276	0.335	0.334
GradNorm	0.659	0.336	0.400	0.490	0.478
SynFlow	0.635	0.355	0.420	0.519	0.498
NASWOT	0.600	0.332	0.381	0.437	0.438
CRoZe	**0.723**	**0.417**	**0.501**	**0.602**	**0.588**

(b) Adversarially-Trained

Table 2: Comparison of Spearman's ρ between the actual accuracies and the proxy values on CIFAR-100 and ImageNet16-120 in NAS-Bench-201 search space. Plain, Grasp, Fisher, GradNorm, SynFlow are zero-cost methods from Abdelfattah et al. (2021). NASWOT (Mellor et al., 2021) is zero-cost proxy approach using activation as a proxy. Avg. stands for average Spearman's ρ values with all accuracies within each task.

Proxy Type	CIFAR-100							ImageNet16-120		
	Clean	FGSM	Weather	Noise	Blur	Digital	Avg.	Clean	FGSM	Avg.
FLOPs	0.705	0.663	0.674	0.25	0.444	0.607	0.557	0.657	0.611	0.634
#Params.	0.720	0.654	0.685	0.240	0.438	0.618	0.559	0.683	0.627	0.655
Plain	-0.126	-0.100	-0.095	-0.002	-0.064	-0.080	-0.078	-0.15	-0.145	-0.148
Grasp	0.475	0.548	0.486	0.262	0.374	0.441	0.431	0.400	0.437	0.419
Fisher	0.378	0.573	0.419	0.325	0.435	0.391	0.420	0.315	0.388	0.352
GradNorm	0.635	**0.791**	0.549	**0.358**	**0.534**	0.609	0.579	0.562	0.643	0.603
SynFlow	0.769	0.685	0.703	0.217	0.389	0.642	0.568	0.751	0.695	0.723
NASWOT	0.683	0.513	0.353	0.273	0.517	0.000	0.390	0.653	0.686	0.670
CRoZe	**0.787**	0.693	**0.747**	0.251	0.450	**0.682**	**0.602**	**0.769**	**0.696**	**0.733**

We validate the final performance of the neural architectures discovered by CRoZe (Table 3) and compare the search time and performance with existing NAS frameworks including robust NAS (RobNet (Guo et al., 2020) and AdvRush (Mok et al., 2021)), clean zero-shot NAS (SynFlow, GradNorm (Abdelfattah et al., 2021)), and clean one-shot NAS (PC-DARTS (Xu et al., 2020) and DrNAS (Chen et al., 2020)). For a fair comparison between clean zero-shot NAS (SynFlow, GradNorm) and CRoZe, we sample the same number (5,000) of candidate architectures using the warmup+move strategy in the DARTS search space. All experiments are conducted on a single NVIDIA 3090 RTX GPU to measure search costs.

Our proxy surpasses the robust NAS methods, RobNet and AdvRush, in terms of robust accuracy against FGSM on CIFAR-10, achieving improvements of 8.95% and 6.21%, respectively. Notably, CRoZe also shows the highest HRS accuracy with an 8.56% increase compared to AdvRush on CIFAR-10, indicating that the neural architectures discovered by CRoZe effectively mitigate the trade-off between clean and robust accuracy, even with 14.7 times reduced search cost. When compared to the previously best-performing clean zero-cost proxy, SynFlow, CRoZe finds architectures with significantly superior performance across clean, common corruptions, and FGSM scenarios, showcasing the effectiveness of our proxy in identifying generalized

Table 3: Comparisons of the final performance of the searched network and search time in DARTS search space on CIFAR-10 and CIFAR-100.

NAS Method	Zero cost	# Params (M)	Time (GPU sec)	Standard-Trained			
				Clean	CC.	FGSM	HRS
CIFAR-10							
PC-DARTS (Xu et al., 2020)		3.60	8355	**95.35**	73.62	14.56	25.26
DrNAS (Chen et al., 2020)		4.10	46857	94.64	72.62	13.96	24.33
RobNet (Guo et al., 2020)		5.44	274062	95.30	72.51	13.43	23.54
AdvRush (Mok et al., 2021)		4.20	251245	94.80	72.00	16.17	27.63
GradNorm (Abdelfattah et al., 2021)	✓	4.69	9740	92.84	71.82	15.55	26.64
SynFlow (Abdelfattah et al., 2021)	✓	5.08	10138	90.41	66.93	10.59	18.96
CRoZe	✓	5.52	17066	94.45	**74.63**	**22.38**	**36.19**
CIFAR-100							
PC-DARTS (Xu et al., 2020)		3.60	8355	76.96	49.95	7.93	14.38
DrNAS (Chen et al., 2020)		4.10	46857	**77.46**	**50.76**	7.87	14.29
RobNet (Guo et al., 2020)		5.44	274062	76.15	49.43	6.47	11.93
AdvRush (Mok et al., 2021)		4.20	251245	76.33	49.53	8.21	14.83
GradNorm (Abdelfattah et al., 2021)	✓	3.83	9554	67.95	42.81	5.11	9.51
SynFlow (Abdelfattah et al., 2021)	✓	4.42	9776	75.93	48.53	8.22	14.83
CRoZe	✓	4.72	17457	75.18	49.35	**10.84**	**18.95**

architectures. Additionally, the neural architecture chosen by CRoZe outperforms clean one-shot NAS approaches in HRS accuracy (Figure 1).

5. Conclusion

While neural architecture search (NAS) is a powerful technique for automatically discovering high-performing deep learning models, previous works suffer from two major drawbacks: computational inefficiency and compromised robustness against diverse perturbations, which hinder their applications in real-world scenarios with safety-critical applications. In this paper, we proposed a simple yet effective lightweight robust NAS method that can rapidly search for well-generalized neural architectures against diverse perturbations. To this end, we proposed a novel consistency-based zero-cost proxy that evaluates the robustness of randomly initialized neural networks by measuring the consistency in their features, parameters, and gradients for both clean and perturbed inputs. Experimental results demonstrate the effectiveness of our approach in discovering well-generalized architectures across diverse search spaces, multiple datasets, and various types of perturbations, outperforming the baselines with significantly reduced search costs. Such simplicity and effectiveness of our approach open up new possibilities for automatically discovering high-performing models that are well-suited for safety-critical applications.

Acknowledgement

This work was supported by the Institute of Information communications Technology Planning Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2020-0-00153) and Institute of Information communications Technology Planning Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2019-0-00075, Artificial Intelligence Graduate School Program (KAIST)).

References

Abdelfattah, M. S., Mehrotra, A., Dudziak, Ł., and Lane, N. D. Zero-cost proxies for lightweight nas. *International Conference on Learning Representations*, 2021.

Baker, B., Gupta, O., Naik, N., and Raskar, R. Designing neural network architectures using reinforcement learning. In *International Conference on Learning Representations*, 2017.

Biggio, B., Corona, I., Maiorca, D., Nelson, B., Šrndić, N., Laskov, P., Giacinto, G., and Roli, F. Evasion attacks against machine learning at test time. In *Joint European conference on machine learning and knowledge discovery in databases*, pp. 387–402. Springer, 2013.

Cai, H., Gan, C., Wang, T., Zhang, Z., and Han, S. Once-for-all: Train one network and specialize it for efficient deployment. *International Conference on Learning Representations*, 2019.

Chen, X., Wang, R., Cheng, M., Tang, X., and Hsieh, C.-J. Drnas: Dirichlet neural architecture search. *arXiv preprint arXiv:2006.10355*, 2020.

Cubuk, E. D., Zoph, B., Shlens, J., and Le, Q. V. Randaugment: Practical automated data augmentation with a reduced search space. In *Advances in Neural Information Processing Systems*, 2020.

Devaguptapu, C., Agarwal, D., Mittal, G., Gopalani, P., and Balasubramanian, V. N. On adversarial robustness: A neural architecture search perspective. In *IEEE International Conference on Computer Vision*, pp. 152–161, 2021.

Dodge, S. and Karam, L. A study and comparison of human and deep learning recognition performance under visual distortions. In *2017 26th international conference on computer communication and networks (ICCCN)*, pp. 1–7. IEEE, 2017.

Dong, P., Li, L., and Wei, Z. Diswot: Student architecture search for distillation without training. *IEEE Conference on Computer Vision and Pattern Recognition*, 2023.

Dong, X. and Yang, Y. Searching for a robust neural architecture in four gpu hours. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1761–1770, 2019.

Elsken, T., Metzen, J. H., and Hutter, F. Efficient multi-objective neural architecture search via lamarckian evolution. *International Conference on Learning Representations*, 2019.

Foret, P., Kleiner, A., Mobahi, H., and Neyshabur, B. Sharpness-aware minimization for efficiently improving generalization. In *International Conference on Learning Representations*, 2021.

Goodfellow, I. J., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*, 2015.

Guo, M., Yang, Y., Xu, R., Liu, Z., and Lin, D. When nas meets robustness: In search of robust architectures against adversarial attacks. *IEEE Conference on Computer Vision and Pattern Recognition*, 2020.

Hendrycks, D. and Dietterich, T. Benchmarking neural network robustness to common corruptions and perturbations. *International Conference on Learning Representations*, 2019.

Hendrycks, D., Mu, N., Cubuk, E. D., Zoph, B., Gilmer, J., and Lakshminarayanan, B. Augmix: A simple data processing method to improve robustness and uncertainty. *International Conference on Learning Representations*, 2020.

Jung, S., Lukasik, J., and Keuper, M. Neural architecture design and robustness: A dataset. In *International Conference on Learning Representations*, 2023.

Lee, N., Ajanthan, T., and Torr, P. H. Snip: Single-shot network pruning based on connection sensitivity. *International Conference on Learning Representations*, 2019.

Liu, C., Zoph, B., Neumann, M., Shlens, J., Hua, W., Li, L.-J., Fei-Fei, L., Yuille, A., Huang, J., and Murphy, K. Progressive neural architecture search. In *European Conference on Computer Vision*, pp. 19–34, 2018a.

Liu, H., Simonyan, K., Vinyals, O., Fernando, C., and Kavukcuoglu, K. Hierarchical representations for efficient architecture search. In *International Conference on Learning Representations*, 2018b.

Liu, H., Simonyan, K., and Yang, Y. Darts: Differentiable architecture search. *International Conference on Learning Representations*, 2019.

Liu, L., Zhang, S., Kuang, Z., Zhou, A., Xue, J.-H., Wang, X., Chen, Y., Yang, W., Liao, Q., and Zhang, W. Group fisher pruning for practical network compression. In *International Conference on Machine Learning*, pp. 7021–7032. PMLR, 2021.

Luo, R., Tian, F., Qin, T., Chen, E., and Liu, T.-Y. Neural architecture optimization. *Advances in Neural Information Processing Systems*, 31, 2018.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018.

Mellor, J., Turner, J., Storkey, A., and Crowley, E. J. Neural architecture search without training. In *International Conference on Machine Learning*, pp. 7588–7598. PMLR, 2021.

Mok, J., Na, B., Choe, H., and Yoon, S. Advrush: Searching for adversarially robust neural architectures. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 12322–12332, 2021.

Moosavi-Dezfooli, S.-M., Fawzi, A., and Frossard, P. Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2574–2582, 2016.

Pham, H., Guan, M., Zoph, B., Le, Q., and Dean, J. Efficient neural architecture search via parameters sharing. In *International Conference on Machine Learning*, pp. 4095–4104. PMLR, 2018.

Real, E., Moore, S., Selle, A., Saxena, S., Suematsu, Y. L., Tan, J., Le, Q. V., and Kurakin, A. Large-scale evolution of image classifiers. In *International Conference on Machine Learning (ICML)*, 2017.

Real, E., Aggarwal, A., Huang, Y., and Le, Q. V. Regularized evolution for image classifier architecture search. In *AAAI Conference on Artificial Intelligence*, 2019.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. Intriguing properties of neural networks. *International Conference on Learning Representations*, 2014.

Tanaka, H., Kunin, D., Yamins, D. L., and Ganguli, S. Pruning neural networks without any data by iteratively conserving synaptic flow. *Advances in Neural Information Processing Systems*, 33:6377–6389, 2020.

Wang, C., Zhang, G., and Grosse, R. Picking winning tickets before training by preserving gradient flow. *International Conference on Learning Representations*, 2020.

Wu, D., Xia, S.-T., and Wang, Y. Adversarial weight perturbation helps robust generalization. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, 2020.

Xu, Y., Xie, L., Zhang, X., Chen, X., Qi, G.-J., Tian, Q., and Xiong, H. Pc-darts: Partial channel connections for memory-efficient architecture search. *International Conference on Learning Representations*, 2020.

Zhang, H., Yu, Y., Jiao, J., Xing, E. P., Ghaoui, L. E., and Jordan, M. I. Theoretically principled trade-off between robustness and accuracy. In *International Conference on Machine Learning*, 2019.

Zhong, Z., Yan, J., Wu, W., Shao, J., and Liu, C.-L. Practical block-wise neural network architecture generation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2423–2432, 2018.

Zoph, B. and Le, Q. V. Neural architecture search with reinforcement learning. *International Conference on Learning Representations*, 2017.

Zoph, B., Vasudevan, V., Shlens, J., and Le, Q. V. Learning transferable architectures for scalable image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.