Beyond the Limits: Overcoming Negative Correlation of Activation-Based Training-Free NAS

Haidong Kang[1] Lianbo Ma[1*] Pengjun Chen[1] Guo Yu[2] Xingwei Wang[1] Min Huang[1]
[1]Northeastern University [2]Nanjing Tech University
{hdkang, pjchen}@stumail.neu.edu.cn, {wangxw, malb}@swc.neu.edu.cn
guo.yu@njtech.edu.cn, mhuang@mail.neu.edu.cn

Abstract

*Training-free Neural Architecture Search (NAS) has emerged an efficient way to discover high-performing lightweight models with zero-cost proxies (e.g., the activation-based proxies (AZP)). In this paper, we observe a new negative correlation phenomenon that the correlations of the AZP dramatically decrease to be negative with the increasing number of convolutions, significantly degrading the prediction performance of AZP over target architectures. No existing works focus on such negative correlation and its underlying mechanism. To address this, through deep analysis of the architectural characteristics scored by AZP, we propose a series of AZP design principles and reveal the potential reason of the above phenomenon that high non-linearity dramatically degrades the magnitude of AZP score. Those findings show that existing AZP designs do not obey the proposed principles. Finally, grounded in these insights, we propose a simple yet efficient Negative Correlations-Defied (**NCD**) method, which utilize stochastic activation masking (SAM) and non-linear rescaling (NIR) to effectively eliminate negative correlation of AZP and significantly improve performance. Extensive experimental results validate the effectiveness and efficiency of our method, outperforming state-of-the-art methods on mainstream 12 search spaces with 4 real-world tasks.*

1. Introduction

Neural Architecture Search (NAS) [23, 43, 49] have attracted lots of interests for its potential to automate the process of searching lightweight CNNs [16, 33] with promising performance from a given search space, while remedying the trial-and-error (manual) neural architecture design efforts and costs. Despite these advantages, NAS still suffers from high computational costs and high resource consumption [25, 31, 48], which has become a key bottleneck that limits its applications in real-world scenarios.

*Corresponding author

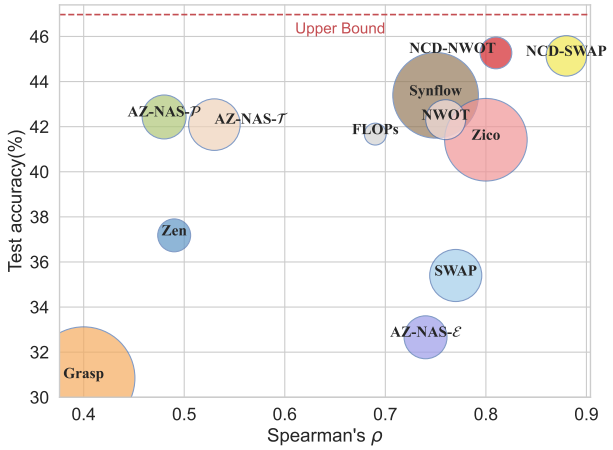
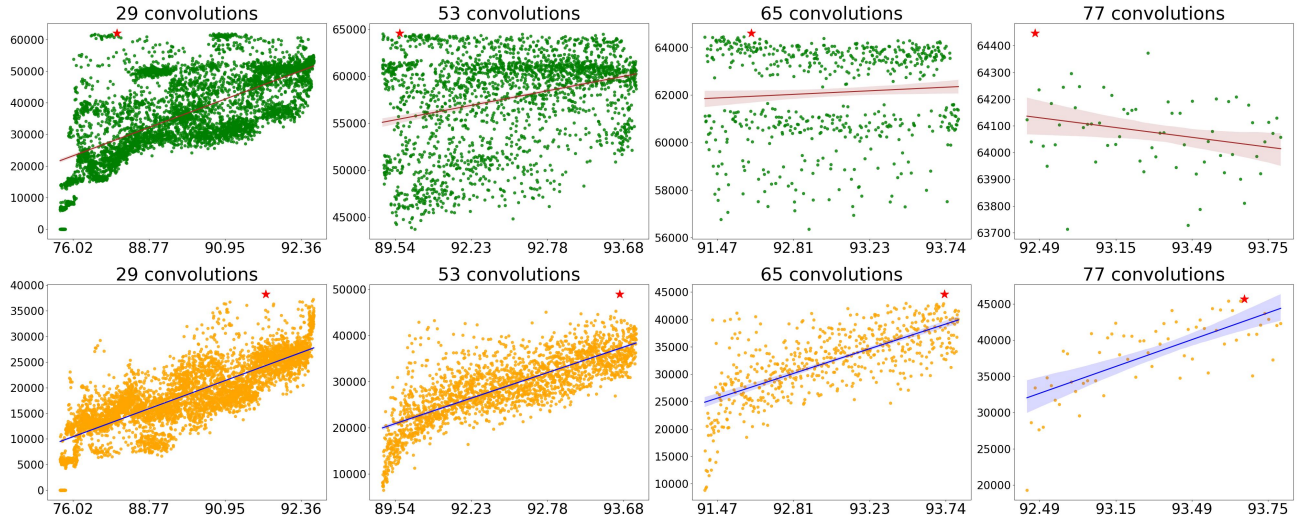


Figure 1. Correlation coefficients (Spearman's ρ) and performance comparison between our method (NCD-NWOT, NCD-SWAP) and peer competitors on NAS-Bench-201 search space in ImageNet16-120. To be specific, the circle size denotes the runtime costs, the circle color denotes different methods. The Upper Bound represents the optimal test accuracy on NAS-Bench-201.

Various training-free NAS methods have been proposed to liberate NAS from this bottleneck through the lens of without training [26]. Most of them utilize zero-cost proxies to predict the performance ranking of candidate neural architectures at initialization (with no need of gradient descent). Typically, the design principle of such zero-cost proxies relies on some statistical or theoretical properties of target architectures [37]. Several lines of research in training-free NAS can be categorized into four groups based on their design principles: statistic-based (i.e., the number of parameters or floating-point operations), activation-based [26], gradient-based [18], and theory-based [27]. Particularly, the activation-based proxies (AZP) are regarded as a representative method, which works well without additional gradient. Our work focuses on the activation-based training-free NAS, i.e., AZP-based NAS.



(b)

Figure 2. **AZP Score** *vs.* **Test Accuracy** on NAS-Bench-201 in CIFAR10 dataset. (a) SWAP. (b) our proposed NCD. In addition, we use a straight line to show the correlation. We show four subspaces (i.e., 29, 53, 65, 77) based on the number of convolutions on the NAS-Bench-201 search space. Notably, the x-axis represents the accuracy of architectures, while the y-axis denotes the AZP score. ⋆ denotes the final architecture searched by AZP with the highest score.

1.1. Challenges

Negative Correlation. Current activation-based training-free NAS methods usually suffer from a significant drawback: *the negative correlation issue* (as shown in Fig. 2a), i.e., the performance ranking in terms of architectures predicted by zero-cost proxies is not always consistent with, even sometimes opposite to the real ones. Due to this issue, training-free NAS methods are often less practical than training-based NAS methods that utilize the final performance of architectures to build architectural representations. This limitation highlights the need for more effective and efficient training-free NAS methods.

Our New Observations. Different from existing studies, we observe a new negative correlation phenomenon, that is, the correlations of AZP dramatically decrease to be negative with the increasing number of convolutions. More specifically, some negative correlation confronted by AZP-based NAS is related to the information about FLOPs and non-linearity within target architectures, which has not been revealed in previous works. Obviously, such particular negative correlation would lead to significant deterioration of AZP's prediction performance. Regarding to this point, we have made deep analysis about the underlying mechanism of negative correlation in training-free NAS.

1.2. Contributions

In this paper, we attempt to understand and overcome the above drawback caused by negative correlation. To realize this, we first propose four principles to guide the design of AZP. Especially, we conduct an in-depth analysis of architectural characteristics scored by AZP, and observe that high non-linearity dramatically degrades the magnitude of AZP score, which could be potential reasons of the negative correlation issue due to not obeying above principles, where the relationship between the influence of non-linearity and architectural representations pattern is ignored.

Then, to tackle this issue, motivated by the above empirical observations, we propose a simple yet efficient **NCD** method in the views of stochastic activation masking and non-linear rescaling. Instead of only relying on simple activation patterns (0 or 1), our NCD considers the additional influence of FLOPs and non-linearity on activation-based architectural representation patterns. As shown in Fig.2b, our NCD perfectly addresses the negative correlation issue, which empirically validates that our proposed method strictly obeys the above principles. To sum up, the main contributions of our work include:

- **Distinctive Negative Correlation Phenomena.** We discover that a distinctive negative correlation phenomenon of AZP that has not been mentioned in previous studies. Through deep analysis about architectural characteristics scored by AZP, we propose four principles to design AZP, and then reveal the key reason of the above phenomena: *ignoring the impact of non-linearity*.
- **Negative Correlation-Defied AZP-based NAS.** We propose a simple yet effective NCD method, which utilizes stochastic activation masking (SAM) and non-linear rescaling (NIR) to overcome above negative correlation. In fact, our method captures the actual impact of architecture's non-linear capability to activation-based architectural representations pattern, and thus strictly obeys the above design principles.

- **Numerical Verification.** Rigorous theoretical analysis and extensive experiments demonstrate that NCD consistently achieves better performance gains in terms of correlation and accuracy than SOTA methods (Fig. 1).

2. Mechanism Analysis of AZP

2.1. Rethinking AZP

The main goal of AZP is to enable NAS to be free from high computational costs and high resource consumption through the training-free way. To achieve this goal, we should consider a key question: **How to predict the ranking of candidate architectures at initialization?** Naturally, we can utilize some statistical or theoretical properties of architecture as proxies to obtain the ranking of architectures at initialization. For example, we can use the number of parameters as a proxy to rank neural architecture. However, these simple proxies can not accurately capture the non-linearity of architecture, and thus limit the scope of architecture exploration. Consequently, this raises a concern about how to design the proxies that are ideal for training-free NAS. For sake of clarity, we present the illustration of AZP in the search spaces DARTS and NAS-Bench-201, as shown in Fig. 3. We can clearly observe that: (1) **The activation (ReLU) function** plays a key role in the design of architecture; (2) **The non-linearity** of candidate architecture is attributed to the ReLU function.

The Activation Patterns. The above observations motivate us to tailor activation-based proxies (AZP) to make them adapt to training-free NAS. To realize this, considering the fact that ReLU function can transform its input space into active and inactive regions, we need to define the activation patterns to accurately obtain the representation of non-linearity in neural architecture. To be specific, given a neural architecture $F(\cdot)$ from a search space $\mathcal{A}$ and a batch $\mathcal{B}$ of input images $X = \{s_i\}_{i=1}^{\mathcal{B}}$ in dataset $\mathcal{D}$ for task $\mathcal{T}$, we can get feature maps for each layer, and then flatten and concatenate them into one-dimensional vectors $F(s_i)$. Subsequently, each activation pattern c_i can be obtained by applying the Heaviside step function $H(x)$ to the corresponding flattened feature maps in $F(\cdot)$ at samples s_i. The $H(x)$ is defined as:

$$H(x) = \begin{cases} 0 & \text{if } x < 0, \\ 1 & \text{if } x \geq 0. \end{cases} \tag{1}$$

As shown in Eq. 1 and Fig. 3, we observe that the expressibility of architecture $F(\cdot)$ relies on simple activation patterns (0 or 1), which replace the role of gradient descent to optimize the weights of $F(\cdot)$ in dataset $\mathcal{D}$ for task $\mathcal{T}$.

Analysis of representative AZP. We take NWOT[26] (a representative AZP-based NAS) as example to analyze the characteristics of AZP in NAS. The intuition of NWOT is that the more similar the activation patterns c_i are then the more challenging it is for architecture to separate these input samples to achieve better expressibility. In fact, NWOT is to measure the following dissimilarity between c_i:

$$\boldsymbol{K}_H = \begin{pmatrix} N_A - d_H(c_1, c_2) & \cdots & N_A - d_H(c_1, c_N) \\ \vdots & \ddots & \vdots \\ N_A - d_H(c_N, c_1) & \cdots & N_A - d_H(c_N, c_N) \end{pmatrix}$$

$$S_{NWOT} = log|\boldsymbol{K}_H|, \tag{2}$$

where N_A is the length of c_i, $d_H(c_i, c_j)$ denotes the Hamming distance between c_i and c_j, and S_{NWOT} represents an indicator of architecture accuracy. As shown in Eq. 2, we can obtain that: (1) the greater dissimilarity among each c_i, the higher the S_{NWOT} value, and (2) the higher S_{NWOT} value indicates better expressibility and higher accuracy.

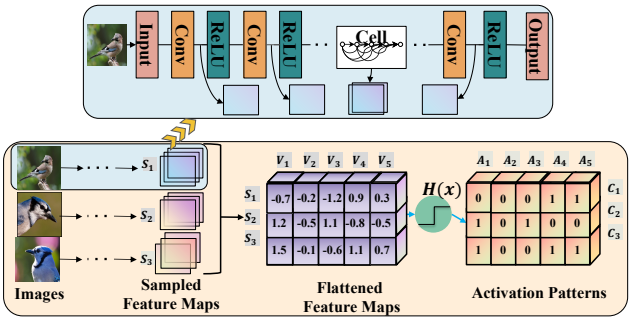


Figure 3. The illustration of the activation pattern.

Although the NWOT effectively indicates the accuracy of the architecture, it possesses a constrained numerical range. This limited range can make architectures indistinguishable when their scores approach the numerical upper bound. To alleviate this limitation, SWAP[29] suggested stacking each activation pattern c_i along the batch dimension and subsequently counting the number of unique vectors among all active region vectors a_i as the proxy score. For instance, the activation patterns depicted in Fig.3 result in a score of 4. To this end, the higher SWAP value indicates better expressibility and higher accuracy.

2.2. Design Principles

According to the aforementioned analysis of AZP, it is challenging to design AZP due to the extreme complexity of neural architectures. Considering the intrinsic feature of training-free NAS in terms of the role of AZP, we propose a set of basic design principles that need to be obeyed: (1) AZP should accurately predict the ranking of candidate neural architectures at initialization, (2) AZP only relies on some statistical or theoretical properties of neural architectures, (3) the score of AZP should be consistent with architectural characteristics (i.e., parameters, flops, depths, etc.), and (4) AZP should reflect the truly nonlinear capability of neural architectures.

To this end, we firmly believe that the proposed design principles will benefit the design of effective AZP, facilitating the development of the NAS community. Besides

of AZP, those design principles can be generalized to other Zero-cost proxies (i.e., gradient-based, theory-based ones).

3. The Negative Correlation Issue

3.1. Experimentation Negative Correlation Issue

To verify and understand the negative correlation issue of AZP, first, we introduce several subspaces (i.e., 5, 17, 29, 41, 53, 65, 77) based on the number of convolutions on the NAS-Bench-201 search space to explore the impact of AZP on subspaces. Then, we perform detailed experimentation on the aforementioned negative correlation issue of AZP (i.e., NWOT, SWAP) on those subspaces in the CIFAR10 dataset. To be specific, we provide comparison figures of correlation distribution between AZP's score and real accuracy on those subspaces (as presented in Fig. 2). Notably, we use a straight line to show Spearman's ρ correlation. From Fig. 2a, we observe that the ρ value of SWAP significantly decreases with the increasing number of convolutions. Especially, in subspace 77 there is a serious *negative correlation* between SWAP and test accuracy.

Furthermore, we provide a more detailed comparison of the correlation between representative AZP (i.e., NWOT, SWAP) score and real accuracy on different subspaces (as shown in Table 1). As shown, as the number of convolutions in the subspace increases, the correlation of NWOT and SWAP decreases accordingly. Especially, the correlation of NWOT degrades to -0.734 on subspace 77. Such correlation degradation would misguide the search of AZP-based NAS in the search space (§2.1).

3.2. Further Analysis

To reveal the reasons behind negative correlation, we compare the correlation between FLOPs and NWOT on different subspaces, as shown in Fig. 4a. As shown, there exists a slight positive correlation in subspace 17, while AZP has a significant negative correlation with FLOPs on subspace 77. Similar observation is obtained on SWAP. This finding shows that existing AZP methods (e.g., SWAP, NWOT) violate the design principle of AZP, i.e., principle (3).

Why are the FLOPs negatively correlated with the AZP on the subspace 77? What factor of the AZP causes the negative correlation? To answer the above questions, we construct neural architectures by stacking 100 layers consisting of operations (i.e., conv 3x3, conv 5x5, ReLU, BN layer), and visualize AZP scores of the output feature map obtained from each layer, as shown in Fig. 4b, where left/right plots the relationship between existing AZP (NWOT, SWAP) and layers for linear/nonlinear architecture (without ReLU/with ReLU). From Fig. 4b (left), we find that these AZP can well reflect the expressibility of linear neural architecture. In brief, the activation patterns (0 or 1) of existing AZP can deal with simple linear neural architecture. By contrast, from Fig. 4b (right), we observe that the score of NWOT increases towards a peak point and then dramatically decreases, which is very confusing.

Table 1. The impact of different subspaces of NAS-Bench-201 on Spearman's ρ correlation of AZP methods. The red, and orange indicate the best, and second-best, respectively.

Search space	SWAP(ρ)	NCD-SWAP(ρ)	NWOT(ρ)	NCD-NWOT(ρ)	Average Acc(%)
5 convolutions	0.769	0.765	0.773	0.747	66.28
17 convolutions	0.863	0.804	0.809	0.751	81.05
29 convolutions	0.659	0.829↑(0.170)	0.522	0.590↑(0.068)	87.38
41 convolutions	0.428	0.823↑(0.395)	0.280	0.518↑(0.238)	90.73
53 convolutions	0.233	0.782↑(0.549)	-0.008	0.481↑(0.489)	92.27
65 convolutions	0.010	0.694↑(0.684)	-0.379	0.591↑(0.970)	92.86
77 convolutions	-0.246	0.715↑(0.961)	-0.734	0.679↑(1.413)	**93.21**
Average	0.387	0.773↑(0.386)	0.180	0.622↑(0.442)	-

For above results, we are curious about whether the negative correlation of AZP is associated to the complex non-linearity of neural architecture (which is caused by the accumulation of multiple ReLU transformations [32]). As pointed out in [26], the high non-linearity of the output results comes from the summation of multiple ReLU activation values. This motivates us to explore the impact of the accumulation of ReLU transformation on the non-linearity of neural architecture. For instance, given one-dimensional input tensor $\mathbf{x}^{(l)} \in \mathbb{R}^n$ from the preceding layer l, the current layer $l+1$ can produce a scalar value $y^{(l+1)}$ with increased non-linearity as follows:

$$y^{(l+1)} = \sum_{i=1}^{n} \sigma(x_i^{(l)}), \tag{3}$$

where $\sigma(.)$ is the ReLU function. To this end, when the non-linearity of the feature map reaches a stable point, the further increase of non-linearity will significantly decrease the magnitude of the AZP score. Consequently, the current layer must produce a feature map with a lower degree of non-linearity enhancement to mitigate this decline. In brief, the convolution operation can be viewed as the weighted summation of the input activation transformations. For this point, convolution with smaller FLOPs (i.e., reducing kernel size) can reduce the non-linearity and attain a higher score. This conclusion can be supported by Fig. 4b, where conv 5x5 leads to more score degradation than conv 3x3.

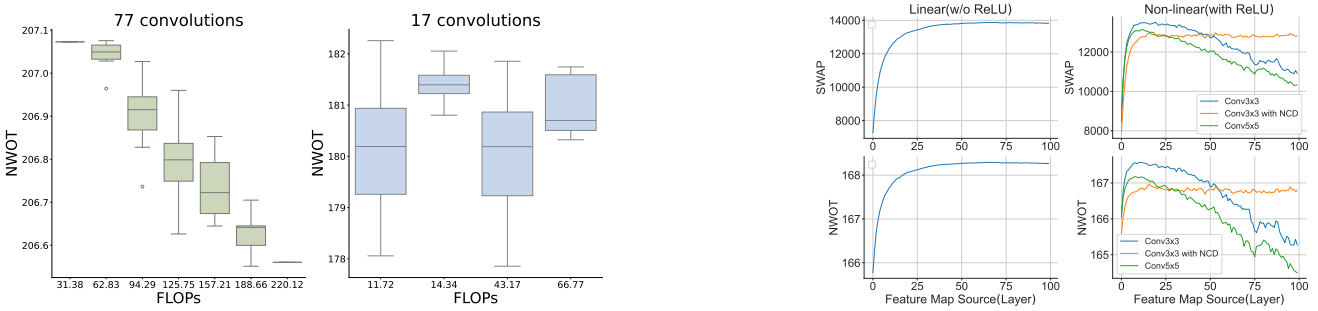
Conclusion. In summary, the root cause of negative correlation is attributed to high non-linearity of architectures.

4. Method

To address negative correlation, we present a framework of NCD (as shown in **App. A.1**), which strictly obeys principles, and utilizes *Stochastic Activation Masking (SAM)* and a *Non-linear Rescaling (NIR)* to avoid high non-linearity.

4.1. Stochastic Activation Masking

Given that the convolution can be viewed as the weighted summation of input activation values, a direct way to avoid high non-linearity is to reduce the number of activation values involved in the convolution calculation of the network. However, merely eliminating the fixed activation values from the convolution process will restrict the integrity



(a) (b)

Figure 4. (a) The boxplots for NAS-Bench-201 subspaces featuring 17 and 77 convolutions, where the x-axis represents FLOPs and the y-axis denotes the NWOT score. (b) We construct 2 architectures that are stacked with 100 layers, each layer is a cell with Conv-BN(Linear)/ReLU-Conv-BN(Non-linear). Subsequently, we collect the AZP scores (SWAP, NWOT) calculated on the feature map originating from different layers. Conv3x3/5x5 denotes using the convolution with kernel size 3,5, respectively.

of input features, which may lead to information loss of the output feature maps and negatively affect the AZP assessment on the output feature map. Therefore, we introduce the stochastic activation masking (SAM) to decrease the non-linearity of architectures. Each time the filter is applied to the input activation feature map, the activation values within the corresponding receptive field are masked with a probability $\alpha \in [0, 1]$, defined as:

$$y = \sum (\mathbf{W} \odot \mathbf{M} \odot \mathbf{X}), \qquad \mathbf{M}_{(i,j,k)} \sim \text{Bernoulli}(1-\alpha), \tag{4}$$

where $\mathbf{W}$, $\mathbf{X}$, $\mathbf{M}$ represent the filter weights, mask tensor, and input activation tensor, respectively. The symbol $\sum$ denotes the summation of all elements, and $\odot$ denotes the Hadamard product. Since the input activation values contribute to computations with filters to generate output values across various channels, each activation value is likely retained in the output computation by stochastic masking, thereby alleviating the impact of information loss without changing the sorting of the original architectures. Furthermore, SAM can significantly accelerate the AZP, we provide detailed efficiency analysis in **App. A.2**.

4.2. Non-linear Rescaling

In the context of NAS, Batch Normalization (BN) is commonly employed after convolutions to address the internal covariate shift [14] while improving the convergence rate and generalization capability [2]. However, we observe that BN still contains numerous summations of activation values as formulated in Eq. 3. This outcome results in a substantial rise in non-linearity within the output values of BN. Given a convolution output tensor $\mathbf{X} \in \mathbb{R}^{B\times C\times H\times W}$, where B, C, H, and W denote the batch, channel, height, and width, respectively. The BN with value $x_{(i,j,k,p)}$ can be defined as:

$$x^{BN}_{(i,j,k,p)} = \frac{x_{(i,j,k,p)} - \mu_{(:,j,k,p)}}{\sqrt{Var[\mathbf{x}_{(:,j,k,p)}] + \epsilon}}, \tag{5}$$

$$Var[\mathbf{x}_{(:,j,k,p)}] = \frac{\sum_{u=1}^{B} \left(x_{(u,j,k,p)} - \mu_{(:,j,k,p)}\right)^2}{B}, \tag{6}$$

where ϵ is used for numerical stability. Therefore, in the calculation of the BN process, the sole source for the summation of the ReLU activation values is derived from the computation of the $\mu_{(j,k,p)}$.

Theorem 4.1. *Given a convolution output tensor $\mathbf{X} \in \mathbb{R}^{B\times C\times H\times W}$, the calculation of $\mu_{(j,k,p)}$ within the BN of value $x_{(i,j,k,p)}$ can be formulated as:*

$$\mu_{(:,j,k,p)} = \frac{\sum_{a=1}^{B}\sum_{b=1}^{C}\sum_{c=\lfloor -\frac{K}{2}\rfloor}^{K}\sum_{d=\lfloor -\frac{K}{2}\rfloor}^{K} \hat{x}_{(a,b,c,d)})}{B}, \tag{7}$$
$$\hat{x}_{(a,b,c,d)} = \omega_{(b,c-\lfloor -\frac{K}{2}\rfloor, d-\lfloor -\frac{K}{2}\rfloor)} \cdot \sigma(x^{input}_{(a,b,j+c,p+d)}),$$

where $\omega_{(a,b,c)}$ is the weight of convolution kernel tensor $\mathbf{W} \in \mathbb{R}^{C\times K\times K}$, and $\sigma(.)$ is the ReLU function.

Proof. Given that the $\mu_{(:,j,k,p)}$ in BN is calculated within the batch dimension B, which can be formulated as:

$$\mu_{(:,j,k,p)} = \frac{\sum_{a=1}^{B} x^{output}_{(a,j,k,p)}}{B}, \tag{8}$$

where $x^{output}_{(a,j,k,p)}$ is the value of the output feature map resulting from the convolution operation. Due to the fact that the value of the output feature map is computed by the values within the local receptive field of the feature map applied with ReLU, we have:

$$x^{output}_{(a,j,k,p)} = \sum_{b=1}^{C}\sum_{c=\lfloor -\frac{K}{2}\rfloor}^{K}\sum_{d=\lfloor -\frac{K}{2}\rfloor}^{K} \hat{x}_{(a,b,c,d)}, \tag{9}$$
$$\hat{x}_{(a,b,c,d)} = \omega_{(b,c-\lfloor -\frac{K}{2}\rfloor, d-\lfloor -\frac{K}{2}\rfloor)} \cdot \sigma(x^{input}_{(a,b,j+c,p+d)}),$$

where $\sigma(.)$ is the ReLU function, and ω is the weight of the filter kernel. Then, Eq. 7 can be obtained by substituting Eq. 9 into Eq. 8. Theorem 4.1 is proved. □

Remark 1. Theorem 4.1 shows that $\mu_{(:,j,k,p)}$ in BN can be viewed as a weighted summation of ReLU activation values, quantified by $B \cdot C \cdot K^2$. The acquired

normalized values inevitably increase non-linearity. However, it is challenging to alleviate the non-linearity by diminishing the number of activation values in the summation, as this alters the statistical properties of the data distribution and results in a shifted computed μ.

Theorem 4.2. *Given a convolution output tensor* $\mathbf{X} \in \mathbb{R}^{B\times C\times H\times W}$*, the calculation of* $\mu_{(i,:,:,:)}$ *within the Layer Normalization(LN) of value* $x_{(i,j,k,p)}$ *can be formulated as:*

$$\mu_{(i,:,:,:)} = \frac{\sum_{a=1}^{H}\sum_{b=1}^{W}\sum_{c=1}^{C}\sum_{d=\lfloor -\frac{K}{2}\rfloor}^{K}\sum_{e=\lfloor -\frac{K}{2}\rfloor}^{K}\hat{x}_{(i,c,a+d,b+e)}}{H\cdot W},$$
$$\hat{x}_{(i,c,a+d,b+e)} = \mathbf{E}[w]\cdot\sigma(x^{input}_{(i,c,j+c,p+d)}), \tag{10}$$

where $\mathbf{E}[w]$ *represents the expectation of the kernel weight* w*. The detailed proof is provided in* ***App. A.3****.*

Remark 2. Theorem 4.2 shows that in LN, although more ReLU activation values are involved in the summation, they are all weighted by the expectation of the convolution weight w. Since AZP works at initialization with w initialized with 0 mean, **every input activation value in LN can be approximated as weighted by zero, which considerably decreases non-linearity of the output value.** It is clear that substituting BN with LN during AZP evaluation mitigates negative correlation.

5. Experiments

We conduct experiments on 12 search spaces (i.e., NAS-Bench-201, NAS-Bench-101, DARTS, TransNAS-Bench-101–Micro, TransNAS-Bench-101–Macro, S1, S2, S3, S4, MobileNet-like, AutoFormer, OoD-ViT-NAS) and 4 tasks (i.e., image recognition, autoencoding, scene classification, and self-supervised jigsaw puzzle) to validate the effectiveness of NCD. The detailed experimental settings are presented in **App. B**.

5.1. Effectiveness on Negative Correlation of AZP

From results in Fig. 2, Fig. 1, and Table 1, we can obtain following observations: (1) Current AZP methods indeed suffer from negative correlation, which significantly degrades their prediction accuracy, as shown in Fig. 2a and Table 1. In particular, when architectures are sampled from subspace 77 on the NAS-Bench-201 search space, the correlation of NWOT and SWAP degrades to -0.734 and -0.246, respectively. The accuracy of NWOT and SWAP degrades to 35.40 % and 42.31 % in ImageNet16-120, as shown in Fig. 1. (2) By contrast, our NCD achieves SOTA test accuracy and Spearman's ρ correlation. As shown in Fig. 2b and Table 1, our method changes the negative correlation into the positive one in subspace 77, e.g., from -0.246ρ to 0.715ρ and from -0.734ρ to 0.679ρ, respectively.

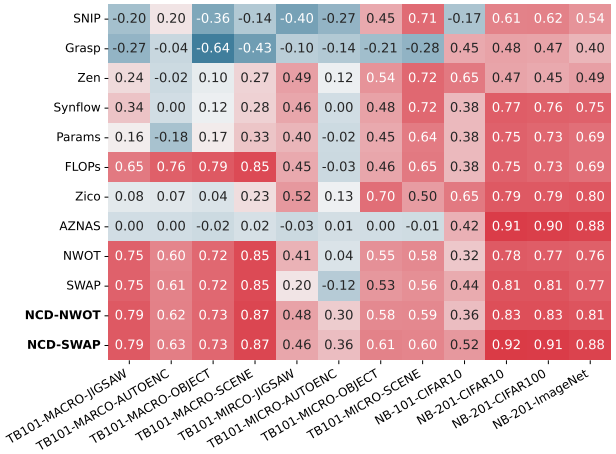


Figure 5. Comparison of **NCD** *vs.* **10 existing zero-cost proxies** in terms of Spearman's rank correlation coefficients across 12 search spaces over five independent runs.

In addition, as shown in Fig. 5, we compare our method with a wide scope of 10 peer competitors across 12 popular search spaces. Especially, NCD consistently achieves remarkable Spearman's ρ correlation. Notably, our method shows stronger generalization ability than other methods, i.e., "AZNAS", "Zico", and "Zen". To sum up, these results again validate the effectiveness of our method in solving negative correlation of AZP, which also provides new insights for advancing training-free NAS.

5.2. Results on NAS-Bench-201&101

Analysis of NAS-Bench-201. Table 2 reports our method and its competitors on the NAS-Bench-201 in CIFAR-10/100 and ImageNet16-120 datasets. In this case, we can find that our method consistently surpasses several training-free methods across all datasets. To be specific, compared to SOTA method, our method achieves 0.31%, 0.29%, and 0.93% test accuracy improvements in CIFAR-10, CIFAR-100, and ImageNet16-120 datasets with 0.9% fewer costs. Specifically, compared with training-free methods in the ImageNet16-120 dataset, the test accuracy of our method is 29.96% higher than Snip, 14.42% higher than Synflow, 9.87% higher than NWOT, and 2.96% higher than SWAP.

Analysis of NAS-Bench-101. As shown in Table 2, we observe that our method obtains optimal test accuracy (93.49%) and fewest runtime (29.01 ms). To sum up, those empirical results demonstrate the superiority of our method, which can serve as a valuable solution tailored for NAS.

Table 2. Performance comparison on NAS-Bench-201 (NB-201) search space with CIFAR-10/100, and ImageNet16-120 datasets, and NAS-Bench-101 (NB-101) search space with CIFAR-10 dataset. We report the average results obtained from our method, which are conducted over 5 independent runs. The red, blue, and orange indicate the best, second-best, and third-best results, respectively.

Method	CIFAR-10(NB-201) Test acc.(%)	CIFAR-100(NB-201) Test acc.(%)	ImageNet16-120(NB-201) Test acc.(%)	CIFAR10(NB-101) Test acc.(%)	Runtime (ms/arch)
Params	93.61 ± 0.08	70.95 ± 0.37	41.67 ± 0.64	92.18 ± 1.53	-
FLOPs	93.61 ± 0.08	70.95 ± 0.37	41.67 ± 0.64	92.18 ± 1.53	-
Snip [1]	86.63 ± 3.89	56.53 ± 7.56	15.13 ± 10.86	85.41 ± 1.38	45.78
Grasp [1]	88.01 ± 3.14	61.43 ± 7.04	30.85 ± 5.66	87.08 ± 1.40	93.70
Synflow [1]	93.67 ± 0.39	71.70 ± 0.94	43.39 ± 3.21	89.93 ± 2.97	78.26
NWOT [26]	91.95 ± 1.29	68.88 ± 1.60	42.31 ± 3.43	93.16 ± 0.36	36.54
ZenNAS [21]	89.55 ± 1.12	64.69 ± 3.86	37.18 ± 3.17	93.06 ± 0.73	30.07
ZiCo [19]	93.69 ± 0.07	70.63 ± 1.08	41.42 ± 0.97	92.64 ± 0.99	75.50
AZ-NAS [18]	93.49 ± 0.30	70.33 ± 1.16	44.34 ± 1.26	92.01 ± 0.85	69.43
SWAP [29]	90.48 ± 0.94	67.13 ± 1.83	35.40 ± 3.96	90.51 ± 2.08	47.61
NCD-SWAP($\alpha = 0.95$)	93.68 ± 0.47	70.94 ± 1.31	45.14 ± 0.53	93.23 ± 0.22	37.03
NCD-NWOT($\alpha = 0.95$)	94.00 ± 0.44 ↑(0.31)	71.99 ± 0.88 ↑(0.29)	45.27 ± 1.05 ↑(0.93)	93.49 ± 0.34 ↑(0.33)	29.01 ↓(7.53)
Optimal	94.33 ± 0.08	73.30 ± 0.20	46.97 ± 0.19	93.87 ± 0.08	-

Table 3. Performance comparison on the DARTS search space in CIFAR-10/100 (left) and ImageNet1k(right) datasets, where "Img" denotes network models directly searched in ImageNet1k, and "C100" denotes network models searched in CIFAR-100. For fair comparison, we retrained searched architectures using the publicly available codes [23], and compare them with results reported in original paper. "∗" denotes the corresponding results from the released training codes [46]. "-" indicates that the value is not found in the original paper or the training codes are not provided. "Paper" denotes the results that come from the original paper.

Method	CIFAR-10 Test Err. (%) Retrain	CIFAR-10 Test Err. (%) Paper	CIFAR-100 Test Err. (%) Paper	Params (M)	GPU-Days	Search Method
ResNet18 [12]	-	-	24.39	11.2	-	Manual
AmoebaNet-A [49]	-	2.55 ± 0.05	16.82	-	3150	Evolution
ENAS [30]	-	-	18.91	4.6	0.5	Evolution
PNAS [22]	-	3.41	17.63	3.2	225	SMBO
NASNet-A [49]	-	2.63	17.81	3.3	1,800	RL
DARTS(2nd) [23]	-	2.76 ± 0.09	17.54	3.3	1	Gradient
DARTS(1nd) [23]	-	3.00 ± 0.14	16.82	3.3	0.4	Gradient
P-DARTS [6]	2.70 ± 0.15∗	2.50	16.63	3.4	0.3	Gradient
PC-DARTS [42]	2.71 ± 0.11∗	2.57 ± 0.07	17.11	3.6	0.1	Gradient
DARTS+ [20]	-	2.50 ± 0.11	16.28	3.7	0.4	Gradient
DARTS- [7]	2.62∗	2.59 ± 0.08	17.51	3.5±0.13	0.4	Gradient
FairDARTS-D [8]	2.71∗	2.54 ± 0.05	-	3.8	0.4	Gradient
DARTS+PT [38]	-	2.61 ± 0.08	-	3.0	0.8	Gradient
β-DARTS [43]	-	2.53 ± 0.08	16.52	3.8±0.15	0.4	Gradient
Λ-DARTS [28]	-	2.52 ± 0.11	16.53	3.5±0.13	-	Gradient
FP-DARTS [39]	-	2.50 ± 0.05	16.5±0.05	3.9	0.08	Gradient
DARTS-AER^b [17]	-	2.53 ± 0.02	-	3.64 ± 0.18	0.3	Gradient
IS-DARTS [11]	-	2.56 ± 0.04	-	4.25 ± 0.22	0.42	Gradient
NWOT [26]	4.27	-	-	5.0	-	Training-free
TENAS [5]	2.63±0.064	-	-	3.8	0.05	Training-free
NASI-ADA [34]	-	2.90±0.13	-	3.7	0.24	Training-free
SWAP [29]	-	2.48±0.09	-	4.3	0.004	Training-free
NCD-SWAP($\alpha = 0.5$, C10)	2.50±0.05	-	16.50±0.00	4.3	0.004	Training-free
NCD-NWOT($\alpha = 0.5$, C10)	2.44±0.03 ↓(0.04)	-	16.48±0.00 ↓(0.02)	4.4	0.002 ↓(2×)	Training-free

Method	Top-1(%)	Top-5(%)	Params (M)	FLOPs (M)	GPU-Days	Search Method
ResNet50 [12]	75.3	92.2	25.6	4100	-	Manual
AmoebaNet-A [49]	74.5	92.4	6.4	555	3150	Evolution
ProxylessNAS-RL [3]	74.6	92.3	5.8	465	8.3	RL
EfficientNet-B0 [36]	76.3	93.2	5.3	390	≈3000	RL
NASNet-A [49]	74.0	91.6	5.3	564	2000	RL
DARTS [23]	73.3	91.3	4.7	574	4	Gradient
FBNet [41]	74.9	-	5.5	375	216	Gradient
P-DARTS(C100) [6]	75.3	92.5	5.1	577	0.3	Gradient
PC-DARTS(Img) [42]	75.8	92.7	5.3	597	3.7	Gradient
DARTS+ [20]	76.3	92.8	5.1	591	0.2	Gradient
DARTS-(Img) [7]	76.2	93.0	4.9	467	4.5	Gradient
FairDARTS-B(Img) [8]	75.1	92.5	4.8	541	-	Gradient
DARTS+PT(C10) [38]	74.5	92.0	4.6	-	0.8	Gradient
β-DARTS(C100) [43]	75.8	92.9	5.4	597	0.4	Gradient
Λ-DARTS [28]	75.7	-	5.2	-	-	Gradient
FP-DARTS(C10) [39]	75.7	92.7	5.4	-	0.08	Gradient
PDARTS-AER^{ad} [17]	76.0	92.8	5.1	578	2.0	Gradient
IS-DARTS [11]	75.9	92.9	6.4	-	0.42	Gradient
NAO [24]	74.3	91.8	11.4	584	200	Proxy
TENAS [5]	75.5	92.5	5.4	-	0.17	Training-free
NASI-ADA(C10) [34]	75.0	92.2	4.9	559	0.01	Training-free
SWAP [34] (Img)	76.0	92.4	5.8	-	0.006	Training-free
NCD-SWAP($\alpha = 0.5$, C10)	76.7 ↑(0.2)	93.5 ↑(0.3)	4.8	563	0.004 ↓(1.5×)	Training-free
NCD-NWOT($\alpha = 0.5$, C10)	76.9 ↑(0.5)	93.8 ↑(0.6)	4.3	527	0.002 ↓(3×)	Training-free

Table 4. Results on TransNAS-Bench-101-Micro search space.

Method	Autoencoding SSIM	Scene Classification Accuracy (%)	Jigsaw Accuracy (%)
Ground Truth	0.58	54.9	95.4
Grad_norm [1]	0.36± 0.03	48.7±0.7	80.3±0.3
SNIP [1]	0.33±0.04	48.7±1.1	80.3±0.1
Grasp [1]	0.33±0.06	50.2±1.6	91.1±0.3
Fisher [1]	0.49±0.01	48.7±0.6	83.5±1.2
Synflow [1]	0.46±0.07	53.7±1.2	90.9±0.4
NWOT [26]	0.43±0.02	53.2±0.6	92.3±0.3
Zen-score [21]	0.46±0.01	53.7±0.2	87.5±0.4
GradSign [45]	0.35±0.03	53.6±0.4	93.1±0.4
Params	0.46	53.7	85.9
FLOPs	0.46	53.7	85.9
ZiCo [19]	0.48±0.02	53.7±0.4	93.2±0.4
SWAP [29]	0.42±0.02	45.0±10.9	89.8±5.6
NCD-SWAP($\alpha = 0.85$)	0.55±0.01 ↑(0.06)	53.8± 0.8 ↑(0.1)	93.3±1.1 ↑(0.1)
NCD-NWOT($\alpha = 0.85$)	0.55±0.01 ↑(0.06)	53.8±0.7 ↑(0.1)	93.6±0.1 ↑(0.4)

5.3. Results on DARTS Search Space

Results in CIFAR-10 and CIFAR-100 Datasets. As depicted in the left Table 3, we can find that our method shows superior performance with gradient-based methods and training-free methods, indicating strong generalizability among various search spaces. In particular, our method boosts AZP (i.e., NWOT) to obtain SOTA performance with the least search costs. For example, "NCD-NWOT" obtains maximum 0.02% higher test accuracy in CIFAR-10 with 2× search costs than peer competitors.

Results in ImageNet1k Dataset. The experimental results in ImageNet1k are reported in the right Table 3. From the table, we can see that our method combined with SWAP consistently outperforms compared to SOTA methods: it achieves best 76.7 % Top-1 accuracy, with 0.004 (GPU-days) fewest search costs. Compared with training-free methods, our method achieves the Top-1 accuracy of 1.7% higher than SWAP, 1.7% higher than NASI-ADA, and 1.2% higher than TENAS. Notably, our method only searches in the small CIFAR-10 dataset, and then can well generalize to large ImageNet1k, which largely reduces search costs.

5.4. Results on TransNAS-Bench-101–Micro

As depicted in Table 4, we can find that our method achieves best performance on transNAS-Bench-101–Micro, indicating strong generalizability among three tasks. Notably, compared with other methods for Autoencoding task, our method achieves maximum 0.11% higher SSIM and 0.12% higher SSIM than NWOT, and SWAP, respectively.

5.5. Results on MobileNet-like Search Space

To further scrutinize the generalizability of our method, we consider a wide scope of the baselines on MobileNet-like search space in ImageNet1k dataset, and Fig. 5 shows results between our method and its peer competitors. As shown, we can observe that our method surpasses its competitors in terms of Top-1/5 accuracy, showing superior generalizability on MobileNet-like search space. Notably, our method obtains the highest Top-1 accuracy of 78.0%, which is higher than the SOTA method (i.e., RLNAS) 2.4%.

Table 5. Results on MobileNet-like search space in ImageNet1k.

Method	Year	Top-1 (%)	Top-5 (%)	Params (M)
PloxylessNAS [3]	ICLR2019	75.1	92.8	7.1
FBNet-C [41]	CVPR2019	74.9	92.1	4.4
FairNAS-A [8]	ICCV2021	75.3	7.6	4.6
FairDARTS-D [8]	ICCV2021	75.6	92.4	5.3
RLNAS [44]	CVPR2021	75.6	92.6	5.3
GM+ProxylessNAS [13]	ICCV2022	75.6	93.0	4.9
OLES [15]	NeurIPS2023	75.3	92.4	4.7
NCD-NWOT($\alpha = 0.5$)	-	78.0↑(2.4)	94.0 ↑(1.0)	6.9

5.6. Strong Generalizability on Transformer

To further validate the generalizability of our method for transformer, we conduct extensive experiments on the AutoFormer search space in the ImageNet1k dataset. As depicted in Fig. 6, we can clearly observe that our method obtains state-of-the-art performance in terms of accuracy and search costs. Notably, our method achieves maximum 0.2% higher Top-1 accuracy on AutoFormer-T search space with 1.5× search speed than SOTA method "NWOT". Those empirical results demonstrate the strong generalizability of our method for transformer designing.

Table 6. Comparison with Vision Transformers in ImageNet1k.

Method	Year	Search Cost	Top-1 (%)	Search Method	Model Type
ViT-Ti[10]	ICLR2020	-	74.5	Manual	Transformer
NWOT [26]	ICML2021	0.03	78.4	Training-free	CNNs
AutoFormer-T [4]	CVPR2021	24	74.7	Evolution	Transformer
ViTAS-C [35]	ECCV2022	32	74.7	Evolution	Transformer
TF-TAS-T [47]	CVPR2022	0.5	75.3	Training-free	Transformer
Auto-Prox [40]	AAAI2024	0.1	75.6	Training-free	Transformer
AZ-NAS [18]	AAAI2024	0.04	76.4	Training-free	CNNs
ParZC [9]	AAAI2025	0.05	75.5	Hybrid	CNNs
NWOT+SAM($\alpha = 0.9$)	-	0.02↓(1.5×)	78.6↑(0.2)	Training-free	CNNs

5.7. Ablation Studies

The impact of NIR and SAM: We conduct ablation experiments on NB-201 and DARTS to thoroughly scrutinize the impact of component (i.e., NIR, SAM) on accuracy. Table 7 yields two conclusions: (1) we can observe from ① and ② that NIR and SAM successfully boost accuracy of baselines. (2) We find that ③ obtains a joint contribution of NIR and SAM to accuracy, verifying its significance in overcoming negative correlation of AZP.

The impact of α: To scrutinize the stability of α, we conduct ablation study for α on NB201 in C10 dataset. As shown in Table 8, we find that the accuracy of our method is insensitive to α. Those results indicate that our method has strong stability for α. Moreover, we conduct an in-depth analysis of the reason for different optimal α utilized among various search spaces. In brief, the major reason for different optimal α utilized among various search spaces is attributed to the bias of different search spaces. This is because that the topology of those search spaces is significantly diverse. For example, the optimal α utilized on S1-S4 is equal ($\alpha = 0.5$) to the value of the DARTS due to the similarity in topology is very high. Same conclusion can be drawn α utilized on NB-201 and NB-101. However, if the discrepancy of two search spaces (i.e., NB-201 *v.s.* DARTS, NB-201 *v.s.* AutoFormer) is huge, α varies. This is attributed to the bias of different search spaces, not our method arises. In addition, Table 8 provide strong evidence where varying α is stable to accuracy. Consequently, we can conclude that different optimal α utilized among various search spaces does not dramatically affect the performance, validating strong stability of NCD to α.

Table 7. Ablation study of SAM and NIR on NB-201. We compare the average test acc.(%) over 5 independent runs.

	SAM	NIR	CIFAR-10	CIFAR-100	ImageNet16-120
SWAP	✗	✗	90.48	67.13	35.40
①	✓	✗	93.15	69.55	42.48
②	✗	✓	93.01	70.44	44.49
③	✓	✓	93.68	70.94	45.14
NWOT	✗	✗	91.95	68.88	42.31
①	✓	✗	93.62	71.25	42.93
②	✗	✓	93.48	70.94	43.32
③	✓	✓	94.00	71.99	45.27

Table 8. Ablation study for hyper-parameter α on NB201 in C10.

	α=0	α=0.5	α=0.7	α=0.8	α=0.9	α=0.95	α=0.99
SWAP+SAM	90.48	90.58	91.96	92.21	92.51	93.15	93.05
NWOT+SAM	91.95	92.16	92.32	93.46	93.60	93.62	93.46
SWAP+SAM+NIR	93.01	93.37	93.43	93.35	93.61	93.68	93.09
NWOT+SAM+NIR	93.48	93.40	93.58	93.61	93.59	94.00	93.30

5.8. Additional Evaluation

Due to the page limit, we provide more experimental results in **App. A-F**. ❶ **Effectiveness on S1-S4 search spaces** are presented in **App. C**. ❷ **Generalizability on on OoD-ViT-NAS search spaces** are presented in **App. D**. ❸ **Visualizations of architectures** are presented in **App. F** ❹ **More experimental results** are presented in **App. E**.

Related Work. The related work is provided in **App. G**.

6. Conclusion and Future Work

This paper identifies a key issue in activation-based training-free NAS methods: the negative correlation phenomenon, which will be intensified as the number of convolutional layers increases and thus results in a marked decline in performance correlation. To mitigate the issue, we introduce an NCD method that leverages non-linear rescaling and masked activation patterns, to resolve the negative correlation issue as well as improve the overall performance. Experimental results show that our method consistently outperforms state-of-the-art approaches across multiple benchmarks, significantly enhancing correlation and evaluation efficiency, and offering valuable insights for advancing the NAS field. In the future, we will study how to reduce the search costs of our proposed method.

Acknowledgments

This work is supported by National Natural Science Foundation of China under Grant 62472079, the State Key Program of National Natural Science of China (62432003), Fundamental Research Funds for the Central Universities (N2417003), the Key Technologies R&D Program of Liaoning Province (2023JH1/10400082).

References

[1] Mohamed S. Abdelfattah, Abhinav Mehrotra, Lukasz Dudziak, and Nicholas Donald Lane. Zero-cost proxies for lightweight NAS. In 9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021. OpenReview.net, 2021. 7

[2] Nils Bjorck, Carla P Gomes, Bart Selman, and Kilian Q Weinberger. Understanding batch normalization. Advances in Neural Information Processing Systems, 31, 2018. 5

[3] Han Cai, Ligeng Zhu, and Song Han. Proxylessnas: Direct neural architecture search on target task and hardware. ArXiv, abs/1812.00332, 2018. 7, 8

[4] Minghao Chen, Houwen Peng, Jianlong Fu, and Haibin Ling. Autoformer: Searching transformers for visual recognition. In International Conference on Computer Vision, 2021. 8

[5] Wuyang Chen, Xinyu Gong, and Zhangyang Wang. Neural architecture search on imagenet in four gpu hours: A theoretically inspired perspective. ArXiv, abs/2102.11535, 2021. 7

[6] Xin Chen, Lingxi Xie, Jun Wu, and Qi Tian. Progressive differentiable architecture search: Bridging the depth gap between search and evaluation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 1294–1303, 2019. 7

[7] Xiangxiang Chu, Xiaoxing Wang, Bo Zhang, Shun Lu, Xiaolin Wei, and Junchi Yan. Darts-: robustly stepping out of performance collapse without indicators. ArXiv, abs/2009.01027, 2020. 7

[8] Xiangxiang Chu, Tianbao Zhou, Bo Zhang, and Jixiang Li. Fair darts: Eliminating unfair advantages in differentiable architecture search. In Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XV, pages 465–480. Springer, 2020. 7, 8

[9] Peijie Dong, Lujun Li, Xinglin Pan, Zimian Wei, Xiang Liu, Qiang Wang, and Xiaowen Chu. Parzc: Parametric zero-cost proxies for efficient nas. In Thirty-Ninth AAAI Conference on Artificial Intelligence (AAAI-2025), 2025. 8

[10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In International Conference on Learning Representations, 2020. 8

[11] Hongyi He, Longjun Liu, Haonan Zhang, and Nanning Zheng. Is-darts: Stabilizing darts through precise measurement on candidate importance. In Proceedings of the AAAI Conference on Artificial Intelligence, pages 12367–12375, 2024. 7

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 770–778, 2016. 7

[13] Shoukang Hu, Ruochen Wang, Lanqing Hong, Zhenguo Li, Cho-Jui Hsieh, and Jiashi Feng. Generalizing few-shot nas with gradient matching. In Proceedings of the IEEE/CVF International Conference on Computer Vision, 2022. 8

[14] Sergey Ioffe. Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167, 2015. 5

[15] Shen Jiang, Zipeng Ji, Guanghui Zhu, Chunfeng Yuan, and Yihua Huang. Operation-level early stopping for robustifying differentiable nas. Advances in Neural Information Processing Systems, 36, 2023. 8

[16] Wenxiang Jiang, Hanwei Zhang, Xi Wang, Zhongwen Guo, and Hao Wang. Nerfail: Neural radiance fields-based multiview adversarial attack. In Proceedings of the AAAI Conference on Artificial Intelligence, pages 21197–21205, 2024. 1

[17] Kun Jing, Luoyu Chen, and Jungang Xu. An architecture entropy regularizer for differentiable neural architecture search. Neural Networks, 158:111–120, 2023. 7

[18] Junghyup Lee and Bumsub Ham. Az-nas: Assembling zero-cost proxies for network architecture search. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 5893–5903, 2024. 1, 7, 8

[19] Guihong Li, Yuedong Yang, Kartikeya Bhardwaj, and Radu Marculescu. Zico: Zero-shot nas via inverse coefficient of variation on gradients. arXiv preprint arXiv:2301.11300, 2023. 7

[20] Hanwen Liang, Shifeng Zhang, Jiacheng Sun, Xingqiu He, Weiran Huang, Kechen Zhuang, and Zhenguo Li. Darts+: Improved differentiable architecture search with early stopping. ArXiv, abs/1909.06035, 2019. 7

[21] Ming Lin, Pichao Wang, Zhenhong Sun, Hesen Chen, Xiuyu Sun, Qi Qian, Hao Li, and Rong Jin. Zen-nas: A zero-shot NAS for high-performance image recognition. In 2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021, pages 337–346. IEEE, 2021. 7

[22] Chenxi Liu, Barret Zoph, Maxim Neumann, Jonathon Shlens, Wei Hua, Li-Jia Li, Li Fei-Fei, Alan Yuille, Jonathan Huang, and Kevin Murphy. Progressive neural architecture search. In Proceedings of the European Conference on Computer Vision (ECCV), pages 19–34, 2018. 7

[23] Hanxiao Liu, Karen Simonyan, and Yiming Yang. Darts: Differentiable architecture search. ArXiv, abs/1806.09055, 2018. 1, 7

[24] Renqian Luo, Fei Tian, Tao Qin, Enhong Chen, and Tie-Yan Liu. Neural architecture optimization. Advances in Neural Information Processing Systems, 31, 2018. 7

[25] L. Ma, H. Kang, G. Yu, Q. Li, and Q. He. Single-domain generalized predictor for neural architecture search system. IEEE Transactions on Computers, 73(05):1400–1413, 2024. 1

[26] Joe Mellor, Jack Turner, Amos Storkey, and Elliot J Crowley. Neural architecture search without training. In International Conference on Machine Learning, pages 7588–7598. PMLR, 2021. 1, 3, 4, 7, 8

[27] Jisoo Mok, Byunggook Na, Ji-Hoon Kim, Dongyoon Han, and Sungroh Yoon. Demystifying the neural tangent kernel from a practical perspective: Can it be trusted for neural architecture search without training? In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 11861–11870, 2022. 1

[28] Sajad Movahedi, Melika Adabinejad, Ayyoob Imani, Arezou Keshavarz, Mostafa Dehghani, Azadeh Shakery, and Babak Nadjar Araabi. λ-darts: Mitigating performance collapse by harmonizing operation selection among cells. In The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023, 2023. 7

[29] 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. 3, 7

[30] Hieu Pham, Melody Guan, Barret Zoph, Quoc Le, and Jeff Dean. Efficient neural architecture search via parameters sharing. In International Conference on Machine Learning, pages 4095–4104. PMLR, 2018. 7

[31] Esteban Real, Alok Aggarwal, Yanping Huang, and Quoc V. Le. Regularized evolution for image classifier architecture search. In AAAI Conference on Artificial Intelligence, page 4780–4789, 2019. 1

[32] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 4510–4520, 2018. 4

[33] Fei Shen and Jinhui Tang. Imagpose: A unified conditional framework for pose-guided person generation. Advances in Neural Information Processing Systems, 37:6246–6266, 2024. 1

[34] 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, 2022. 7

[35] Xiu Su, Shan You, Jiyang Xie, Mingkai Zheng, Fei Wang, Chen Qian, Changshui Zhang, Xiaogang Wang, and Chang Xu. Vision transformer architecture search. In Computer Vision – ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXI, page 139–157, 2022. 8

[36] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In International Conference on Machine Learning, pages 6105–6114. PMLR, 2019. 7

[37] Hidenori Tanaka, Daniel Kunin, Daniel L Yamins, and Surya Ganguli. Pruning neural networks without any data by iteratively conserving synaptic flow. Advances in Neural Information Processing Systems, 33:6377–6389, 2020. 1

[38] Ruochen Wang, Minhao Cheng, Xiangning Chen, Xiaocheng Tang, and Cho-Jui Hsieh. Rethinking architecture selection in differentiable nas. ArXiv, abs/2108.04392, 2021. 7

[39] Wenna Wang, Xiuwei Zhang, Hengfei Cui, Hanlin Yin, and Yannnig Zhang. Fp-darts: Fast parallel differentiable neural architecture search for image classification. Pattern Recognition, 136:109193, 2023. 7

[40] Zimian Wei, Peijie Dong, Zheng Hui, Anggeng Li, Lujun Li, Menglong Lu, Hengyue Pan, and Dongsheng Li. Autoprox: Training-free vision transformer architecture search via automatic proxy discovery. In Proceedings of the AAAI Conference on Artificial Intelligence, pages 15814–15822, 2024. 8

[41] Bichen Wu, Xiaoliang Dai, Peizhao Zhang, Yanghan Wang, Fei Sun, Yiming Wu, Yuandong Tian, Peter Vajda, Yangqing Jia, and Kurt Keutzer. Fbnet: Hardware-aware efficient convnet design via differentiable neural architecture search. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 10734–10742, 2019. 7, 8

[42] Yuhui Xu, Lingxi Xie, Xiaopeng Zhang, Xin Chen, Guo-Jun Qi, Qi Tian, and Hongkai Xiong. Pc-darts: Partial channel connections for memory-efficient architecture search. ArXiv, abs/1907.05737, 2019. 7

[43] Peng Ye, Baopu Li, Yikang Li, Tao Chen, Jiayuan Fan, and Wanli Ouyang. $\backslash beta$-darts: Beta-decay regularization for differentiable architecture search. In 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 10864–10873. IEEE, 2022. 1, 7

[44] Xuanyang Zhang, Pengfei Hou, Xiangyu Zhang, and Jian Sun. Neural architecture search with random labels. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 10907–10916, 2021. 8

[45] Zhihao Zhang and Zhihao Jia. Gradsign: Model performance inference with theoretical insights. arXiv preprint arXiv:2110.08616, 2021. 7

[46] Xiawu Zheng, Rongrong Ji, Yuhang Chen, Qiang Wang, Baochang Zhang, Jie Chen, Qixiang Ye, Feiyue Huang, and Yonghong Tian. Migo-nas: Towards fast and generalizable neural architecture search. IEEE Transactions on Pattern Analysis and Machine Intelligence, 43(9):2936–2952, 2021. 7

[47] Qinqin Zhou, Kekai Sheng, Xiawu Zheng, Ke Li, Xing Sun, Yonghong Tian, Jie Chen, and Rongrong Ji. Training-free transformer architecture search. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 10894–10903, 2022. 8

[48] Xiaokang Zhou, Wei Liang, I Kevin, Kai Wang, Hao Wang, Laurence T Yang, and Qun Jin. Deep-learning-enhanced human activity recognition for internet of healthcare things. IEEE Internet of Things Journal, 7(7):6429–6438, 2020. 1

[49] Barret Zoph, Vijay Vasudevan, Jonathon Shlens, and Quoc V Le. Learning transferable architectures for scalable image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 8697–8710, 2018. 1, 7