Supplementary Material

Yi Fan
State Key Laboratory of Novel Software Technology, Nanjing University
Nanjing 210023, China
fanyiplus@smail.nju.edu.cn

Yu-Bin Yang
State Key Laboratory of Novel Software Technology, Nanjing University
Nanjing 210023, China
yangyubin@nju.edu.cn

A. Figure illustrations of evaluation process

In Sec. 3 of the main text, we introduce six probing tasks. Here, we provide illustrations for these six tasks, as shown in Figs. 1 to 6. Additionally, we also present the complete search process in Fig. 7, which includes the evaluation process for a candidate network.

B. Hyperparameter values

In this section, we provide all hyperparameter values we use in our experiments, as listed in Table Tab. 1. Although we have already described their meanings in Sec. 3 of the main text, we include brief explanations in the table for readability.

C. Search space

In this section, we introduce the three search spaces designed in Sec. 4.2 of the main text, which are used for searching Convolutional Neural Networks (CNNs), Transformers, and Mambas, respectively. Essentially, the process of designing a search space involves selecting a set of hyperparameters related to network structure and specifying their respective value sets. Since each combination of hyperparameter values corresponds to a specific network architecture, sampling a candidate network from the search space only requires sampling one value from each hyperparameter's value set. We refer to these hyperparameters as Adjustable Hyperparameters (AHs).

C.1. Search space for CNN

For the overall architecture, we adopt a MobileNetV3-like inverted residual architecture suitable for edge device deployment. The macrostructure AHs are given in Tab. 2. The entire network consists of four stages, each containing several inverted residual blocks, with detailed AHs provided in Tab. 3. Additionally, we design adjustable variables for micro-operations, including normalization layers (batch normalization, group normalization), activation functions (ReLU, Swish), and skip connections (present/absent).

C.2. Search space for Transformer

We integrate the advantages of ViT and Swin to design a hybrid local-global architecture. Similar to the CNN search space, it also includes several macrostructure AHs (shown in Tab. 4) and stage-level AHs (shown in Tab. 5), with the entire network divided into four stages. Furthermore, for multi-head attention, either standard Multi-Head Attention (MHA) or local-window MHA can be selected. To stabilize training, we enforce the use of pre-layer normalization.

C.3. Search space for Mamba

We design the search space primarily based on the VMamba architecture, with AHs shown in Tab. 6. Unlike CNN and Transformer, the network here is no longer divided into stages.

D. Search results compared with training-free NAS methods

In Sec. 4.2 of the main text, we present the search results of our method and a comparison with various training-based NAS methods. Here, we compare our method with various training-free NAS methods. For a fair comparison, we control the search time of all methods to one hour. Specifically, we no longer fix the number of candidate networks to 500; instead, we continuously sample and evaluate networks from the search space until the search time reaches one hour, and then select the highest-scoring network from the evaluated candidates as the search result. As before, we conduct the search separately in three search spaces. The results are presented in Tab. 7. As can be seen, the

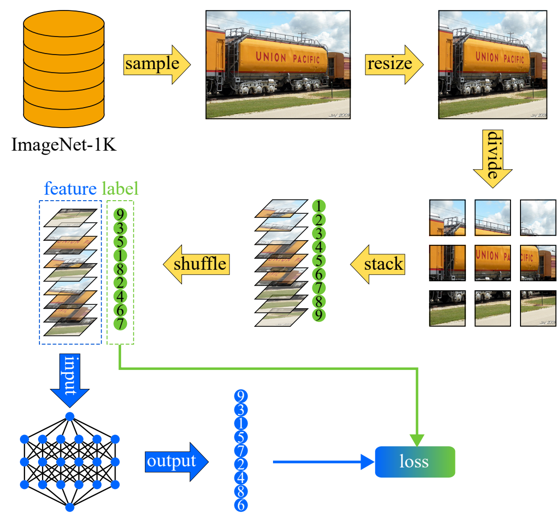

Figure 1. Illustration of probing task Local-Global Jigsaw (LGJ).

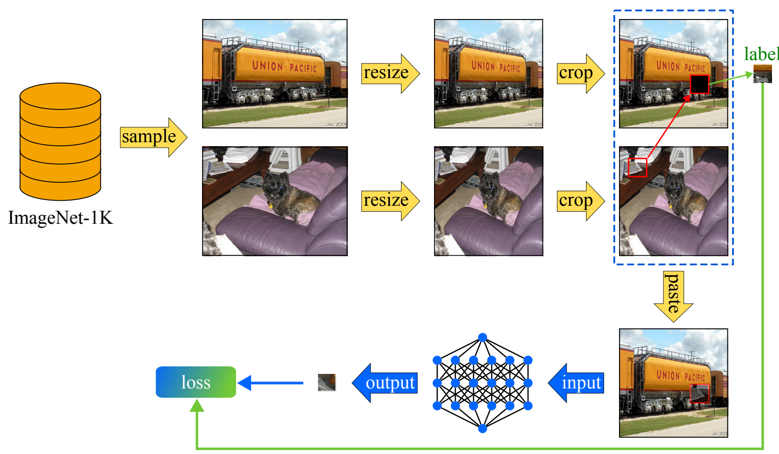

Figure 2. Illustration of probing task Occlusion In-Painting (OIP).

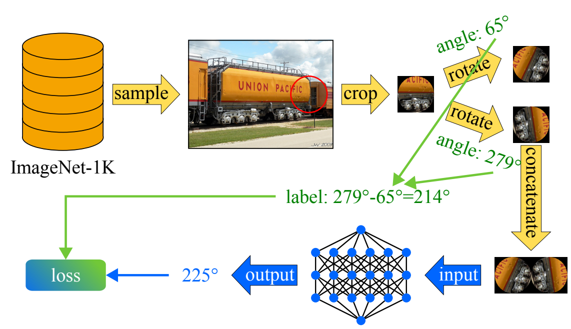

Figure 3. Illustration of probing task Rotation Match (RM).

search results obtained by our method remain state-of-the-art.

References

[1] Mohamed S Abdelfattah, Abhinav Mehrotra, Łukasz Dudziak, and Nicholas D Lane. Zero-cost proxies for lightweight nas. *arXiv preprint arXiv:2101.08134*, 2021. 7

[2] Yi Fan and Yu-Bin Yang. Training-free neural architectural

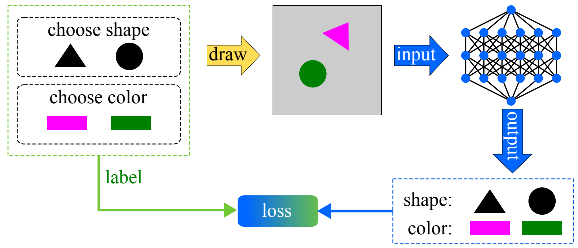

Figure 4. Illustration of probing task Color-Shape Binding (CSB).

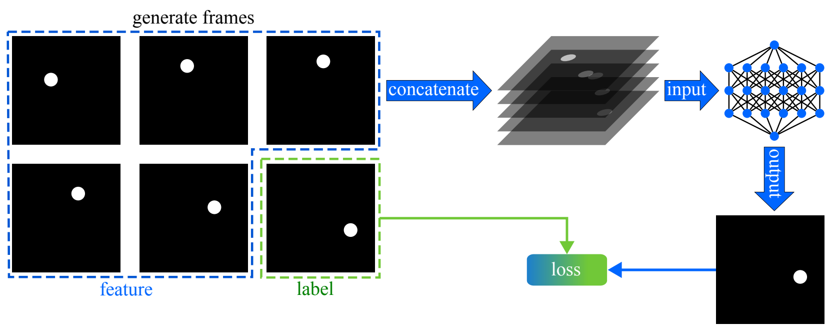

Figure 5. Illustration of probing task Motion Forecast (MF).

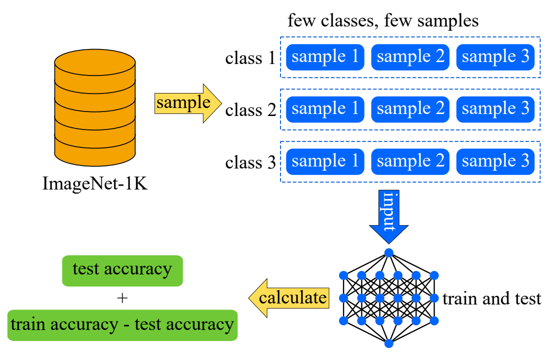

Figure 6. Illustration of probing task Visual Memorization (VM).

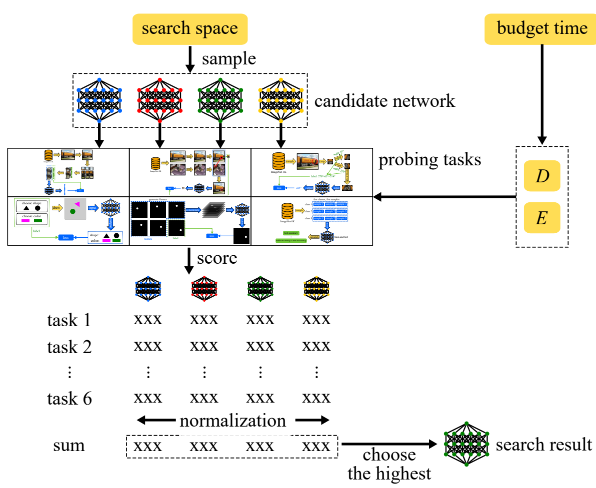

Figure 7. The complete search process of our method.

Table 1. Hyperparameter values used in our experiments.

Hyperparameter	Meaning (parentheses indicate the associated probing tasks)	Value
h	Height after resizing raw image (LGJ, OIP); diameter of sample region (RM); canvas height (CSB, MF)	48
w	Width after resizing raw image (LGJ, OIP); canvas width (CSB, MF)	48
h_p	Image patches height (LGJ)	8
w_p	Image patches width (LGJ)	8
γ	Similarity threshold of two patches (LGJ)	0.9
$o_{\text{h\_min}}$	Minimum height of masks in (OIP)	6
$o_{\text{h\_max}}$	Maximum height of masks in (OIP)	12
$o_{\text{w\_min}}$	Minimum width of masks in (OIP)	6
$o_{\text{w\_max}}$	Maximum width of masks in (OIP)	12
α	Noisy opacity of masks (OIP)	0.5
μ	Weight of variance-penalty term when calculating score (OIP)	0.2
c_1	Number of shape categories (CSB)	6
c_2	Number of colors (CSB)	4
T	Number of frames (MF)	10
k	Number of classes (VM)	10
m_{train}	Number of instances from each class's training split (VM)	10
m_{test}	Number of instances from each class's test split (VM)	10

Table 2. Macrostructure AHs in CNN search space.

AHs name	Value set	Description
Input resolution	{224, 192, 160, 128}	Dynamically adjusted for different tasks
Width multiplier	{0.5, 0.75, 1.0, 1.1}	Global channel scaling factor
Stem channels	{16, 24, 32}	Number of channels in the initial convolution layer

search on transformer via evaluating expressivity and trainability. In *2024 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2024. 7

[3] Yi-Cheng Huang, Wei-Hua Li, Chih-Han Tsou, Jun-Cheng Chen, and Chu-Song Chen. Up-nas: Unified proxy for neural architecture search. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1675–1684, 2024. 7

[4] Tangyu Jiang, Haodi Wang, and Rongfang Bie. Meco: zero-shot nas with one data and single forward pass via minimum eigenvalue of correlation. *Advances in Neural Information Processing Systems*, 36:61020–61047, 2023. 7

[5] 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. 7

[6] Namhoon Lee, Thalaiyasingam Ajanthan, and Philip HS Torr. Snip: Single-shot network pruning based on connection sensitivity. *arXiv preprint arXiv:1810.02340*, 2018. 7

[7] 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

[8] 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 347–356, 2021. 7

[9] Liyang Liu, Shilong Zhang, Zhanghui Kuang, Aojun Zhou, Jing-Hao Xue, Xinjiang Wang, Yimin Chen, Wenming Yang, Qingmin Liao, and Wayne Zhang. Group fisher pruning for practical network compression. In *International Conference on Machine Learning*, pages 7021–7032. PMLR, 2021. 7

[10] Yameng Peng, Andy Song, Haytham M Fayek, Vic Ciesielski, and Xiaojun Chang. Swap-nas: Sample-wise activation patterns for ultra-fast nas. In *ICLR*, 2024. 7

[11] Aaron Serianni and Jugal Kalita. Training-free neural architecture search for rnns and transformers. In *The 61st Annual Meeting Of The Association For Computational Linguistics*, 2023. 7

[12] 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. 7

[13] Chaoqi Wang, Guodong Zhang, and Roger Grosse. Picking winning tickets before training by preserving gradient flow. *arXiv preprint arXiv:2002.07376*, 2020. 7

[14] Zimian Wei, Peijie Dong, Zheng Hui, Anggeng Li, Lujun Li, Menglong Lu, Hengyue Pan, and Dongsheng Li. Auto-prox: Training-free vision transformer architecture search via automatic proxy discovery. In *Proceedings of the AAAI Con-

Table 3. Stage-level AHs in CNN search space.

AHs name	Value set	Description
Blocks per stage	Stages 1-4: $\{1,2,3\} \times 4$	Total depth $\leq$ 12 layers
Expansion ratio	$\{3, 4, 6\}$	1×1 expansion factor
Kernel size	$\{3, 5, 7\}$	Depthwise separable convolution
Output channels	Stages 1-4: $\{16, 24, 32, 48, 64, 96, 128\}$	Stage-wise increasing

Table 4. Macrostructure AHs in Transformer search space.

AHs name	Value set	Description
Patch size	$\{4, 8, 16\}$	Affects sequence length and computational cost
Base embedding dimension	$\{48, 64, 80, 96\}$	Must be divisible by head dimension
Number of layers	$\{6, 8, 10, 12\}$	Total depth

ference on Artificial Intelligence, pages 15814–15822, 2024. 7

[15] Qinqin Zhou, Kekai Sheng, Xiawu Zheng, Ke Li, Yonghong Tian, Jie Chen, and Rongrong Ji. Training-free transformer architecture search with zero-cost proxy guided evolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 7

Table 5. Stage-level AHs in Transformer search space.

AHs name	Value set	Description
Blocks per stage	$\{1,2,3\} \times 4$	Total blocks = macro-level depth
Number of attention heads	$\{2, 3, 4, 6\}$	Per-head dimension ≥ 32
MultiLayer Perceptron (MLP) ratio	$\{3.0, 3.5, 4.0\}$	FFN hidden layer expansion ratio
Query-Key-Value (Q-K-V) dimension	$\{32, 48, 64\}$	Can be smaller than embedding dimension

Table 6. Macrostructure AHs in Mamba search space.

AHs name	Value set	Description
Patch embedding size	$\{4, 8, 16\}$	Analogous to ViT's patchify
Hidden dimension	$\{48, 64, 80, 96\}$	SSM core dimension
Number of layers	$\{6, 8, 10, 12\}$	Total depth
State dimension	$\{8, 12, 16\}$	SSM state space size
Expansion factor	$\{1.5, 2.0, 2.5\}$	Linear layer expansion ratio
State Space Model (SSM) block type	{Mamba, MLP-mixer, hybrid}	Supports architectural diversity
Scan direction	{bidirectional, 4-directional}	Spatial scanning strategy for vision tasks

Table 7. Search results of various training-free NAS methods. P, F, L, T-1, T-5, SS, MS are abbreviations for parameter number (M), FLOPs (G), latency (s), top-1 accuracy (%), top-5 accuracy (%), mIoU (SS), and mIoU (MS), respectively. For object detection, we adopt the Mask R-CNN 3× MS schedule. Best results in bold.

Network	Method	P	F	L	CL		DT						SG	
					T-1	T-5	AP^b	AP^b_{50}	AP^b_{75}	AP^m	AP^m_{50}	AP^m_{75}	SS	MS
CNN	#Param	16	3.7	0.67	77.2	94.0	46.7	64.6	49.8	29.0	53.3	49.9	41.8	44.9
	Gradnorm [1]	16	4.3	1.03	78.5	95.6	47.2	65.7	50.1	28.7	54.8	50.7	41.9	45.9
	Synflow [12]	17	3.9	1.40	78.5	96.9	46.8	66.6	51.2	29.9	55.6	51.2	42.5	45.9
	GraSP [13]	15	3.5	1.42	74.6	91.0	45.3	61.9	48.0	28.3	51.6	48.3	39.8	43.1
	Fisher [9]	15	3.6	1.90	75.2	93.0	45.5	63.1	48.9	28.4	52.3	48.6	41.1	43.9
	Snip [6]	16	3.9	1.03	75.2	91.3	44.9	62.2	48.4	27.1	52.1	48.4	41.2	43.3
	Zen-NAS [8]	16	4.0	1.05	75.9	92.2	44.9	62.8	47.7	28.3	53.0	49.8	41.2	44.4
	ZiCo [7]	16	4.0	1.26	75.7	91.5	45.1	62.7	48.7	28.6	51.3	48.8	41.0	44.1
	MeCo [4]	16	3.4	1.12	75.1	92.1	45.2	62.4	48.7	28.6	53.1	49.6	40.4	43.9
	AZ-NAS [5]	17	3.0	0.90	79.6	95.8	47.4	65.6	51.0	29.4	55.1	51.2	43.2	46.3
	SWAP-NAS [10]	17	3.9	1.08	80.3	96.6	48.4	66.4	51.5	30.1	**55.7**	51.7	43.6	46.9
	Auto-Prox [14]	16	4.0	1.56	79.9	96.7	47.9	65.8	51.2	29.5	55.2	51.3	44.0	46.2
	UP-NAS [3]	17	3.8	0.58	79.9	96.8	47.4	66.7	51.5	29.6	54.5	52.0	43.3	46.5
	DSS++ [15]	16	4.0	0.99	73.9	90.0	44.1	61.6	47.6	27.2	52.1	47.8	40.1	42.8
	AttnNAS [2]	15	3.2	1.15	74.6	90.8	44.8	62.0	47.7	27.8	51.6	48.1	40.2	43.3
	ours	17	4.1	1.10	**80.9**	**97.9**	**49.8**	**67.1**	**52.1**	**30.7**	**55.7**	**52.9**	**44.3**	**47.6**
Transformer	#Param	16	4.3	0.91	77.1	94.0	46.3	64.0	50.3	29.6	53.3	49.7	41.0	44.3
	Gradnorm [1]	15	3.0	1.08	74.5	90.4	43.7	62.0	48.0	27.6	51.5	47.9	40.8	42.7
	Synflow [12]	15	4.2	1.17	74.8	91.5	45.5	61.8	48.1	27.4	51.7	48.9	40.9	43.4
	GraSP [13]	15	3.1	1.22	75.0	91.2	45.0	62.1	47.3	28.0	51.7	48.2	40.9	43.7
	Fisher [9]	15	3.6	1.30	74.3	90.3	44.6	61.7	47.4	28.3	51.4	47.8	40.2	42.8
	Snip [6]	16	3.4	1.31	75.0	92.1	45.1	62.2	48.6	28.5	52.6	49.0	41.4	44.0
	Zen-NAS [8]	16	4.0	1.45	76.2	91.5	45.8	62.8	48.4	28.3	51.5	49.5	40.8	44.5
	ZiCo [7]	15	3.8	1.20	74.1	90.1	44.2	61.5	47.6	27.4	51.2	48.0	39.7	43.3
	MeCo [4]	16	3.8	0.93	75.7	92.4	45.0	62.8	48.9	28.2	52.2	49.1	40.7	44.3
	AZ-NAS [5]	17	3.9	1.65	75.3	92.5	44.6	62.4	49.0	28.5	52.3	48.8	41.3	44.4
	SWAP-NAS [10]	16	3.6	0.67	75.6	92.5	44.5	63.0	48.7	28.6	52.5	48.9	40.6	44.3
	Auto-Prox [14]	15	3.3	1.00	74.3	90.6	43.7	61.5	48.1	27.3	52.0	47.9	39.6	43.1
	UP-NAS [3]	16	3.8	1.26	73.7	90.5	44.9	61.8	47.4	27.5	51.4	47.8	40.7	43.0
	AC [11]	16	3.8	0.20	78.4	96.4	47.7	65.1	51.0	28.7	54.0	50.1	42.7	46.1
	HI [11]	16	3.7	1.20	79.3	96.1	47.2	65.0	50.6	29.2	54.8	51.1	42.5	46.9
	HC [11]	16	4.3	1.65	79.5	96.0	46.4	65.8	50.5	29.4	54.1	51.1	43.3	46.1
	DSS++ [15]	16	3.8	1.07	79.9	96.7	47.8	66.3	50.4	29.3	54.8	51.4	43.8	46.6
	AttnNAS [2]	16	3.6	1.99	79.6	96.7	48.2	66.4	51.2	29.8	56.0	51.7	42.8	47.0
	ours	17	4.1	1.50	**80.5**	**98.2**	**49.3**	**67.8**	**52.2**	**31.3**	**56.6**	**52.5**	**44.1**	**47.6**
Mamba	#Param	16	4.6	1.61	76.9	93.7	46.7	64.1	50.3	29.0	52.6	50.0	41.5	45.3
	Gradnorm [1]	15	4.6	1.14	73.9	90.4	44.7	61.7	47.7	28.2	52.1	47.5	39.5	43.3
	Synflow [12]	15	3.7	0.89	74.8	91.3	45.6	62.3	48.2	28.5	52.7	48.5	40.9	43.6
	GraSP [13]	15	3.5	1.42	74.4	91.3	44.3	62.8	48.1	27.8	52.3	48.1	40.8	43.6
	Fisher [9]	14	3.4	0.98	74.1	90.8	45.1	62.1	48.6	27.7	51.2	48.3	40.0	43.5
	Snip [6]	16	4.0	2.07	75.3	92.5	45.7	62.7	49.2	28.2	52.0	48.8	40.3	43.8
	Zen-NAS [8]	16	2.8	1.26	76.1	91.1	45.7	63.2	48.8	27.9	52.7	49.5	41.0	44.1
	ZiCo [7]	15	3.4	1.43	74.3	90.2	44.0	61.3	48.1	28.1	51.7	48.4	40.5	42.8
	MeCo [4]	16	3.4	1.17	75.8	92.2	45.8	63.1	49.3	29.1	52.2	49.1	40.4	44.5
	AZ-NAS [5]	16	3.2	1.51	75.7	92.3	45.5	62.2	48.9	28.3	52.0	49.0	41.2	43.7
	SWAP-NAS [10]	16	3.9	1.03	75.4	91.9	45.2	63.4	49.3	27.8	52.3	49.2	41.2	43.9
	Auto-Prox [14]	15	3.9	1.45	74.4	90.3	44.2	61.1	47.5	27.9	51.2	47.7	39.6	43.2
	UP-NAS [3]	15	3.5	0.13	73.8	90.7	44.9	61.2	48.0	27.0	50.8	47.1	40.2	43.4
	DSS++ [15]	15	3.6	1.17	77.4	93.7	46.1	63.4	49.3	28.5	53.4	49.8	41.5	45.8
	AttnNAS [2]	16	4.2	1.64	78.7	95.1	45.9	65.1	50.0	29.1	53.5	50.6	42.9	45.7
	ours	16	3.8	1.05	**80.9**	**97.8**	**48.1**	**67.0**	**51.7**	**29.7**	**56.0**	**52.3**	**44.4**	**47.1**