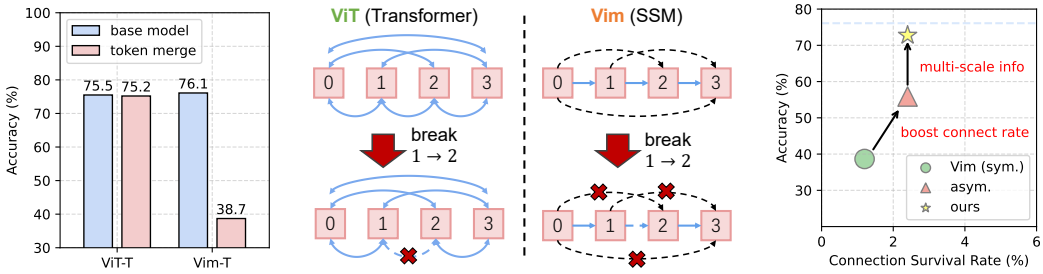
Pruning-Robust Mamba with Asymmetric Multi-Scale Scanning Paths

Jindi Lv[1] **Yuhao Zhou**[1§] **Mingjia Shi**[2] **Zhiyuan Liang**[3] **Panpan Zhang**[3]
Xiaojiang Peng[3] **Wangbo Zhao**[3] **Zheng Zhu**[4] **Jiancheng Lv**[1] **Qing Ye**[1†] **Kai Wang**[3]
[1]Sichuan University [2]University of Virginia [3]National University of Singapore [4]GigaAI



(a) Accuracy of token merging: ViT-T (Transformer) vs. Vim-T (SSM).

(b) An illustrative example showing the impact of token connection disruption in ViT (Transformer) versus Vim (SSM).

(c) Correlation between token connection survival rate and accuracy.

Figure 1: Comparative analysis of token reduction. In Figure (b), blue solid lines denote direct token connections, whereas black dotted lines signify potential connections.

Abstract

Mamba has proven efficient for long-sequence modeling in vision tasks. However, when token reduction techniques are applied to improve efficiency, Mamba-based models exhibit drastic performance degradation compared to Vision Transformers (ViTs). This decline is potentially attributed to Mamba's chain-like scanning mechanism, which we hypothesize not only induces cascading losses in token connectivity but also limits the diversity of spatial receptive fields. In this paper, we propose Asymmetric Multi-scale Vision Mamba (AMVim), a novel architecture designed to enhance pruning robustness. AMVim employs a dual-path structure, integrating a window-aware scanning mechanism into one path while retaining sequential scanning in the other. This asymmetry design promotes token connection diversity and enables multi-scale information flow, reinforcing spatial awareness. Empirical results demonstrate that AMVim achieves state-of-the-art pruning robustness. During token reduction, AMVim-T achieves a substantial 34% improvement in training-free accuracy with identical model sizes and FLOPs. Meanwhile, AMVim-S exhibits only a 1.5% accuracy drop, performing comparably to ViT. Notably, AMVim also delivers superior performance during pruning-free settings, further validating its architectural advantages.

1 Introduction

In recent years, the Mamba architecture, built upon state space models (SSMs), has emerged as a transformative paradigm for efficiently modeling long-range dependencies in vision tasks [1, 2, 3, 4, 5]. By introducing an innovative chain-like scanning mechanism, Mamba [6, 7, 8, 9, 10, 11] successfully reduces the computational complexity from the quadratic demands of Transformers [12, 13, 14] to a linear scale. Alongside these advancements, token reduction techniques (eg., pruning [15, 16, 17]

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

and merging [18, 19]) for Mamba have garnered increasing attention as promising avenues toward further optimization.

Nevertheless, Mamba exhibits significantly greater performance degradation during token reduction compared to Transformers (eg., ViT [13]), as shown in Figure 1a. This discrepancy arises from Transformers using self-attention to establish fully connected token relationships, while Mamba processes tokens sequentially along chain-like scanning paths. This chain-based structure makes Mamba susceptible to cascading information loss during token reduction. For clarity, an illustrative example is provided in Figure 1b.

Recent Mamba variants [20, 21, 22], such as Vim [6], have attempted to mitigate this limitation through **dual-path** scanning strategies that combine forward and reverse sequential paths. While these symmetric designs enhance sequence modeling capabilities and improve baseline performance, they remain ineffective for token reduction. The inherent symmetry of dual-path scanning confines token relationships to the same chain-like structure, failing to address the systemic vulnerability to large-scale connection disruption during pruning.

Inspired by these observations, we hypothesize that minimizing connection disruption during token reduction can mitigate performance drop. To validate this, we introduce asymmetric scanning into dual-path Mamba. As illustrated in Figure 1c, asymmetric scanning paths reduce accuracy degradation from 38% (with symmetric paths) to 21% at the same pruning ratio. This suggests that diversifying chain-like dependencies effectively mitigate pruning-induced performance decline.

To further elucidate this phenomenon, we quantify the token connection survival rate across different dual-path strategies during token reduction. Figure 1c reveals a strong positive correlation between connection survival rates and accuracy, with asymmetric paths exhibiting superior robustness. This confirms our hypothesis: enhancing token connection survival rates via asymmetric path diversification is pivotal for improving Mamba's pruning resilience.

In this work, we propose **AMVim**, a novel **A**symmetric **M**ulti-scale **Vi**sion **M**amba for pruning robustness. To enhance space information diversity, we integrate a window-aware scanning mechanism into one path. By adopting a different scanning direction within windows compared to the main path, we construct multi-level asymmetric paths. This multi-dimensional information flow enables each token to perceive neighborhood information from multiple perspectives. Furthermore, the integration of window-based scanning with the main path creates a multi-level complementary design, allowing for interactions between global context and local dependencies.

Empirically, as shown in Figure 1c, our method results in just a 3% accuracy drop during token reduction (with the blue dotted line representing the baseline accuracy of 76.1%) on ImageNet-1K, achieving a 34% improvement over Vim. This highlights that the multi-scale scanning mechanism enhances the spatial awareness of SSMs, significantly reducing token sensitivity to local variations.

We highlight the main contributions of this paper below:

- We hypothesize token connection survival rate is a critical factor in performance degradation and propose asymmetric scanning paths to effectively mitigate this issue.
- We design a multi-scale asymmetric scanning mechanism that balances global and local spatial information while preserving the benefits of asymmetric paths.
- Our method achieves state-of-the-art pruning resilience, outperforming Vim-T by 34% on ImageNet-1K under identical parameters and FLOPs.

2 Related Work

2.1 State Space Models

SSMs [23, 24, 25, 26, 24, 27] were initially proposed in the NLP community to model long-range dependencies in text. Recently, SSM variants have emerged as effective alternatives to ViTs [13, 14, 12, 28, 29, 30], reducing computational complexity in visual tasks from quadratic to linear time. S4ND [31] is the first work to apply SSMs to visual tasks, extended the S4 [32] model by normalizing the parameters to a diagonal structure. However, this approach struggled to capture image information in an input-dependent manner. In response, Vim [6] was proposed, introducing bidirectional scanning to enhance spatial awareness in vision tasks. Building upon this, PlainMamba [33] introduced

continuous 2D scanning, which improves spatial continuity by maintaining adjacency among tokens within the scanning sequence. Moreover, VMamba [7] proposed an SS2D scanning mechanism that enables comprehensive scanning across four distinct paths.

Despite significant advancements, recent studies [34, 35] have identified limitations in Mamba's chain-like scanning structure, particularly in capturing local spatial dependencies. To address this, Shi et al. [35] introduced a multi-scale 2D scanning technique based on VMamba, combining original and downsampled feature maps to alleviate the long-range forgetting issue. Similarly, LocalMamba[34] proposed a window-based scanning mechanism that dynamically selects search paths at each layer to capture local dependencies. In contrast, we introduce a multi-scale asymmetric scanning mechanism. By improving both direct token connection complementarity and multi-scale information synergy, our method enhances the spatial perception capability of SSM.

2.2 Token Reduction

Token reduction aims to enhance computational efficiency by dynamically removing or consolidating redundant tokens during inference. These methods are typically classified into token pruning [15, 19, 17, 16] and token merging [18, 36, 37, 38, 39]. Token pruning identifies and eliminates low-importance tokens. For example, DynamicViT [15] employs the Gumbel-Softmax strategy to prune less informative tokens, while EViT [19] relies on the attentiveness of the [CLS] token to determine key tokens. In contrast, token merging combines semantically similar tokens, as demonstrated by ToMe [18], a training-free approach that merges tokens via bipartite matching. However, the underlying architectural differences between ViTs and Mambas present unique challenges when applying these techniques to Mambas. First, most token pruning methods are designed for ViTs and rely on self-attention scores, which Mamba lacks, making direct transfer infeasible. Second, token merging in Mamba leads to significant performance degradation due to its chain-like scanning mechanism, which enforces rigid sequential dependencies between tokens.

Recent work has sought to address these challenges. For instance, Zhan et al. [40] proposed a pruning-aware hidden state alignment method to selectively skip tokens in Mamba. Their follow-up work [41] introduced hybrid metrics combining token importance and similarity for pruning. These approaches focus on designing specialized pruning strategies for Mamba. In contrast, our approach aims to strengthen the intrinsic robustness of Mamba's architecture for token reduction. By mitigating chain dependency vulnerabilities, our method enables seamless integration of existing token merging techniques while further improving performance, offering a significant advantage for practical deployment.

3 Method

3.1 Preliminaries

SSMs map a 1D input sequence $x(t) \in \mathbb{R}$ to an output sequence $y(t) \in \mathbb{R}$ through an implicit latent state $h(t) \in \mathbb{R}^N$, governed by linear ordinary differential equations (ODEs):

$$h'(t) = \mathbf{A}h(t) + \mathbf{B}x(t), \quad y(t) \quad = \mathbf{C}h(t), \tag{1}$$

where $\mathbf{A} \in \mathbb{R}^{N\times N}$ governs state transitions, $\mathbf{B} \in \mathbb{R}^{N\times 1}$ projects inputs to the state space, and $\mathbf{C} \in \mathbb{R}^{1\times N}$ maps the state to outputs. A defining feature of SSMs is their chain-like scanning path, which processes inputs sequentially to achieve linear computational complexity. This sequential dependency can be formalized in the discretized recurrence:

$$h_t = \overline{\mathbf{A}}h_{t-1} + \overline{\mathbf{B}}x_t, \quad y_t \quad = \mathbf{C}h_t. \tag{2}$$

where $\overline{\mathbf{A}} = e^{\Delta\mathbf{A}}$ and $\overline{\mathbf{B}} = (\Delta\mathbf{A})^{-1}(e^{\Delta\mathbf{A}} - \mathbf{I}) \cdot \Delta\mathbf{B}$ are derived from the continuous-time parameters via zero-order hold (ZOH) discretization. The chain-like structure ensures that each token $x(t)$ interacts directly with its immediate predecessor $x(t-1)$, propagating information sequentially through the state $h(t)$.

Mamba enhances SSMs with input-dependent selectivity (S6), dynamically adjusting parameters $\mathbf{B}, \mathbf{C}, \Delta$ based on $x(t)$:

$$\mathbf{B}_t = \text{Linear}_B(x_t), \quad \mathbf{C}_t = \text{Linear}_C(x_t), \quad \Delta_t = \text{Softplus}(\text{Linear}_\Delta(x_t)). \tag{3}$$

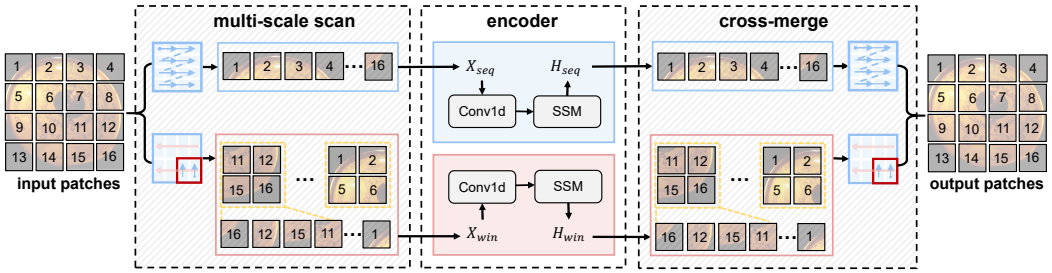
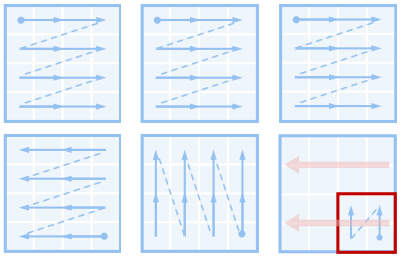


Figure 3: Illustration of the Window-based Multi-Scale State Space (WMS3) block. Input patches are processed along two scanning paths at different scales (multi-scale scan), with each sequence independently encoded (encoder). The outputs are then restored to sequential order and merged to form a 2D feature map as the final result (cross-merge).

While this improves flexibility, the underlying chain-like scanning path remains a core component. Importantly, this chain structure design introduces a critical limitation: the model relies on local neighbor dependencies for information propagation, which amplifies sensitivity to token removal.

3.2 Multi-Scale 2D Selective Scanning

The sequential scanning operation in S6 works well for time-series data in NLP tasks but struggles with non-causal visual data, which is inherently non-sequential and spatially complex. To address this, Vim [6] introduces bidirectional symmetric scanning paths (Figure 2a) to enhance spatial context. Although this design improves spatial context modeling, it fails to resolve the performance collapse during token reduction.



(a) Vim (b) Asym. (c) Ours

Figure 2: Comparison of dual-path scanning strategies. The solid dot represents the starting position.

We attribute this limitation to the chain-like scanning mechanism underlying Mamba, which enforces a rigid neighbor dependency as depicted in Equation 2. To mitigate this issue, we propose a simple yet effective solution: asymmetric scanning paths (Figure 2b). By diversifying the scanning directions, asymmetric paths enhance the complementarity of token connections, as shown in Figure 1c, leading to a significant boost in pruning robustness. However, achieving full token connection complementarity through scanning paths alone remains challenging under computational constraints.

Recent studies [34, 35] emphasize the critical role of local spatial awareness in enhancing Mamba's robustness. Inspired by this insight, we propose a window-based multi-scale selective scanning mechanism. As shown in Figure 2c, this design integrates a window-aware scanning strategy into one path while preserving the global scanning directions of Vim. Specifically, the tokens are partitioned into non-overlapping windows, where tokens are scanned vertically within windows and horizontally globally. This orthogonal design ensures asymmetric dependencies, minimizing redundant token connections while maximizing connection complementarity. Moreover, by harmonizing global structures and local textures, it enables more comprehensive spatial understanding.

3.3 Overall Model Architecture

In this work, we extend Vim [6] by introducing the Window-based Multi-Scale State Space (WMS3) block. As shown in Figure 3, WMS3 operates through three sequential stages: multi-scale scanning, encoding, and cross-merge. Given an input token sequence $\mathbf{X} \in \mathbb{R}^{L \times D}$, where L is the sequence length and D is the feature dimension, WMS3 first reorders $\mathbf{X}$ along two distinct traversal paths: the sequential path and the window-aware path.

Each reordered sequence is then independently processed by a dedicated encoder block, which integrates a 1D convolutional layer and a S6 module. Formally, for a sequence $\mathbf{X}_p$ along path $p \in$

{sequential, window}, the encoder block computes:

$$\mathbf{H}_p = \mathrm{E}(\mathbf{X}_p) = \mathrm{S6}(\mathrm{Conv1D}(\mathbf{X}_p)) \tag{4}$$

where $\mathrm{Conv1D}(\cdot)$ enhances local feature interactions, and $\mathrm{S6}(\cdot)$ models long-range dependencies via selective state transitions.

Finally, the outputs from both paths are restored to their original spatial order and merged via a cross-merge operation:

$$\mathbf{H} = \mathrm{Linear}(\mathbf{H}_{\text{sequential}} + \mathbf{H}_{\text{window}}),$$

where $\mathrm{Linear}(\cdot)$ denotes a linear projection layer. Beyond the WMS3 block, our architecture retains the core design of Vim [6], including the placement of the [CLS] token at the sequence center. This multi-scale design expands spatial perception granularity without additional computational overhead, significantly improving robustness to token reduction while maintaining efficiency.

4 Experiment

4.1 Datasets and Settings

We evaluate AMVim on the ImageNet-1K dataset, which includes 1,000 object classes, 1.28 million training images, and 50,000 validation images. Images are augmented and resized to 224×224 for evaluation. This study focuses on the ImageNet-1K classification task, and we report top-1 validation accuracy. All experiments are conducted with 4 × NVIDIA L40S GPUs.

The window size of AMVim is set to 3×3. AMVim is fine-tuned for 150 epochs with AdamW optimization, initialized using the publicly available weights of Vim [6]. A batch size of 128 is used with two-step gradient accumulation, resulting in an effective total batch size of 1,024. Additional training details are listed in Table 9 in Appendix.

During token reduction, we employ the ToMe technique [18] by default. To ensure a fair comparison, token merging is applied to the even-indexed blocks, covering a total of 12 layers. In each layer, [5, 8, 11, 14] tokens are pruned, corresponding to reduction ratios of [0.17, 0.27, 0.36, 0.46].

4.2 Pruning Robustness Analysis

Comparison with SOTA methods.

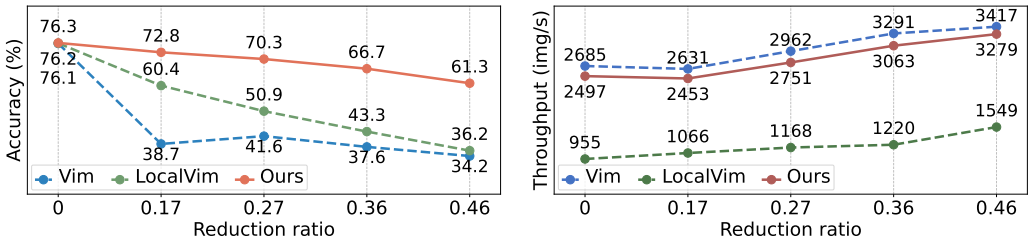
To validate the advancement of the proposed architecture, we compare AMVim with two state-of-the-art token pruning techniques specifically designed for Mamba: Token Recognition [19] and Hidden State Alignment [40]. For a fair comparison, these two methods are evaluated on Vim [6] with a dual-path structure. Additionally, we adhered to their fine-tuning protocols [40] during token reduction and reported the final fine-tuned accuracy.

Table 1: Performance comparison with token pruning methods designed for Mamba on ImageNet-1K classification. AMVim achieves the highest top-1 accuracy across different model scales while maintaining comparable FLOPs.

method	reduction ratio	top-1 acc.(%)		FLOPs (G)	
		Tiny	Small	Tiny	Small
Vim (baseline)	0.00	76.1	80.5	1.45	5.08
Token Recognition [19]	0.17	71.3	74.8	1.28	3.57
Hidden State Alignment [40]	0.17	75.1	78.8	1.29	3.60
AMVim-ToMe (finetune)	0.17	**75.3**	**79.5**	**1.27**	3.60

The results, presented in Table 1, show that AMVim consistently achieves the highest accuracy across various model scales while maintaining comparable FLOPs. Specifically, AMVim-T outperforms Token Recognition by 3.7% in accuracy, while AMVim-S achieves a 4.7% improvement. Furthermore, AMVim-S exhibits only a 1% accuracy drop while reducing FLOPs by one-quarter, significantly surpassing both Token Recognition and Hidden State Alignment. These results highlight that instead of designing specialized token pruning methods for Mamba, addressing its architectural fragility offers a more promising solution.

Robustness across various reduction ratios. Figure 4a compares the top-1 accuracy of our method with recent state-of-the-art vision Mamba models across various token reduction ratios. As the reduction ratio increases, all methods exhibit a predictable decline in performance due to increased information loss. Our method consistently demonstrates superior robustness, outperforming Vim [6] by approximately 30% across all reduction ratios. LocalVim [34] shows improved robustness over



(a) Accuracy.

(b) Throughput.

Figure 4: Comparison with state-of-the-art vision Mamba methods in terms of top-1 accuracy and throughput across various token reduction ratios. Our method demonstrates superior robustness against Vim and LocalVim, while achieving comparable throughput with Vim and twice that of LocalVim. This highlights that our method strikes a good balance between efficiency and robustness.

Vim by incorporating local modeling. However, it still lags significantly behind our method, especially as the reduction ratio increases. Notably, even at a nearly threefold higher reduction ratio (0.46 vs. 0.17), our method still achieves higher accuracy than both Vim and LocalVim.

Figure 4b presents the computational throughput across methods. Our approach achieves throughput comparable to Vim, with a marginal deficit of 200 img/s attributable to directional change operations. In contrast, it delivers approximately 2× higher throughput than LocalVim, which incurs heavier computational overhead from per-layer directional changes. These results collectively underscore our method's optimal trade-off between performance robustness and computational efficiency.

Robustness on various pruning methods.

To further validate the pruning robustness of AMVim, we introduce random token pruning, a method that randomly discards tokens without relying on any pruning criteria (e.g., similarity or importance). Table 2 compares the top-1 accuracy of Vim and AMVim under both token merging and random pruning conditions. The results demonstrate that AMVim consistently outperforms Vim by approximately 30% across all scenarios, irrespective of the pruning method employed. This significant performance gap highlights the exceptional robustness of AMVim's architecture, which consistently delivers strong performance under any pruning method, solidifying its design superiority over Vim.

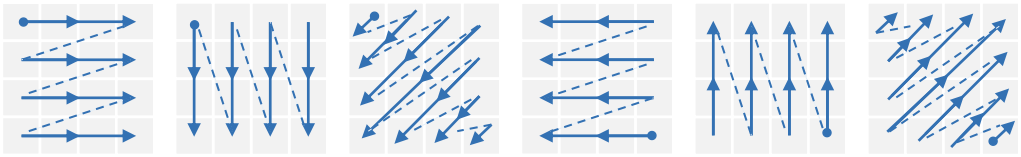
Notably, AMVim exhibits inferior performance under random token pruning compared to token merging. This observation is expected, as token merging leverages similarity as a guiding metric, whereas random pruning lacks such guidance. In contrast, at reduction rates of 0.17 and 0.27, random pruning yields better performance than token merging on Vim. This counterintuitive result suggests

Table 2: Random pruning vs. Token merging: top-1 accuracy (%) under various token reduction techniques. △ denotes the performance difference between Vim and AMVim. AMVim achieves roughly 30% higher accuracy than Vim across both token merging and token pruning methods.

operation	reduction ratio	Vim-T (%)	AMVim-T (%)	△ (%)
pruning	0.17	42.7	72.4	29.7↑
	0.27	45.9	69.5	23.6↑
	0.36	36.6	65.2	28.6↑
	0.46	22.5	58.6	32.1↑
merging	0.17	38.7	72.8	34.1↑
	0.27	41.6	70.3	28.7↑
	0.36	37.6	66.7	29.1↑
	0.46	34.2	61.3	27.1↑

Table 3: Performance comparison between Vim and AMVim on ImageNet-1K. "ToMe" indicates token reduction applied via ToMe [18] for each method, with training-free accuracy reported. △ represents the performance gap between Vim and AMVim. AMVim consistently outperforms Vim across both pruned and non-pruned settings.

method	image size	#param (M)	FLOPs (G)	top-1 acc. (%)	△ (%)
Vim-T	224^2	7	1.5	76.1	0
Vim-S	224^2	26	5.1	80.5	0
ToMe-Vim-T	224^2	7	1.3	38.7	0
ToMe-Vim-S	224^2	26	4.4	78.4	0
AMVim-T	224^2	7	1.5	**76.3**	0.2↑
AMVim-S	224^2	26	5.1	**80.7**	0.2↑
ToMe-AMVim-T	224^2	7	1.3	**72.8**	34.1↑
ToMe-AMVim-S	224^2	26	4.4	**79.2**	0.8↑



(a) D1 (b) D3 (c) D5 (d) D2 (e) D4 (f) D6

Figure 5: Illustration of the scanning directions commonly used in vision Mamba. The solid dot represents the starting position.

that Vim's performance degradation is not inherently tied to the pruning method but rather stems from its architectural vulnerability, rendering the guiding metric ineffective. These findings collectively highlight the resilience of AMVim's architecture while exposing fundamental limitations in Vim's.

4.3 Image Classification

Table 3 compares the performance of Vim and AMVim on the ImageNet-1K classification task with an image size of 224×224. The results show that AMVim consistently achieves higher top-1 accuracy than Vim across various model scales while maintaining the same parameter count and FLOPs. This improvement can be attributed to AMVim's enhanced spatial awareness capability, which effectively achieves the learning of both local dependency and global context.

With token reduction applied (reduction ratio = 0.16), AMVim-T exhibits a training-free accuracy drop of only 3.8%, while AMVim-S shows an even smaller drop of 1.5%, approaching the performance of ViT. Compared to Vim, AMVim-T delivers a substantial 34.1% improvement in pruned accuracy, and AMVim-S shows a 0.8% increase. These results collectively demonstrate that AMVim not only delivers superior robustness to token reduction but also enhances model expressive capability through its innovative design. AMVim strikes a balance between performance and stability, emerging as a powerful solution for vision tasks.

4.4 Semantic Segmentation

We evaluate AMVim on the downstream semantic segmentation task using the ADE20K dataset [42], with results summarized in Table 4. When integrated into the UperNet framework [43], AMVim consistently outperforms Vim across both full-precision and token-reduced settings. In the non-pruned setting, AMVim achieves improvements of +0.2% and +0.1% mIoU over Vim-Ti and Vim-S, respectively, demonstrating enhanced representational capacity due to its multi-scale scanning mechanism.

Table 4: Semantic segmentation performance on ADE20K [42] val set with UperNet. "ToMe" denotes token merging applied via ToMe [18] in a training-free manner. △ indicates the improvement of AMVim over the Vim baseline. AMVim consistently outperforms Vim, and when combined with ToMe, achieves significantly higher mIoU under heavy pruning.

method	backbone	image size	#param (M)	val mIoU (%)	△ (%)
UperNet	Vim-Ti	512^2	13	40.0	0
UperNet	Vim-S	512^2	46	43.3	0
UperNet	AMVim-Ti	512^2	13	**40.2**	0.2↑
UperNet	AMVim-S	512^2	46	**43.4**	0.1↑
UperNet	ToMe-Vim-Ti	512^2	13	21.1	0
UperNet	ToMe-Vim-S	512^2	46	22.0	0
UperNet	ToMe-AMVim-Ti	512^2	13	**27.0**	5.9↑
UperNet	ToMe-AMVim-S	512^2	46	**28.7**	6.7↑

Under ToMe-based token reduction [18], the performance advantage of AMVim becomes even more pronounced, yielding mIoU gains of +5.9% and +6.7% over the corresponding pruned Vim models. This significant improvement under aggressive token merging highlights the robustness and generalization capability of AMVim in dense prediction tasks. These results confirm that AMVim not only improves accuracy in standard evaluation but also exhibits stronger generalization under token merging, making it a more effective and reliable backbone for efficient downstream vision tasks.

4.5 Ablation Study

We conducted extensive ablation studies to validate the effectiveness of each component in AMVim. The scanning directions involved in these experiments are shown in Figure 5. This study specifically

Table 5: Comparison of Symmetric and Asymmetric Paths. △ denotes the performance gap between Vim (i.e., D1-D2) and other dual-path configurations. Asymmetric paths demonstrate superior pruning robustness than symmetric ones.

path1	path2	reduction ratio	FLOPs (G)	top-1 acc. (%)	△ (%)
D1	D2	0	1.45	76.1	0
		0.17	1.27	38.7	0
		0.27	1.13	41.6	0
		0.36	1.00	37.6	0
		0.46	0.86	34.2	0
D1	D4	0	1.45	76.0	0.1↓
		0.17	1.27	55.9	17.2↑
		0.27	1.13	54.5	12.9↑
		0.36	1.00	48.6	11.0↑
		0.46	0.86	44.9	10.7↑
D1	rand	0	1.45	74.9	1.2↓
		0.17	1.27	45.1	6.5↑
		0.27	1.13	43.5	1.9↑
		0.36	1.00	39.3	1.7↑
		0.46	0.86	34.1	0.1↓

Table 6: Ablation study on the impact of the window-aware scanning mechanism on top-1 accuracy (%) during token reduction. Integrating the window-aware scanning mechanism significantly enhances resilience to pruning.

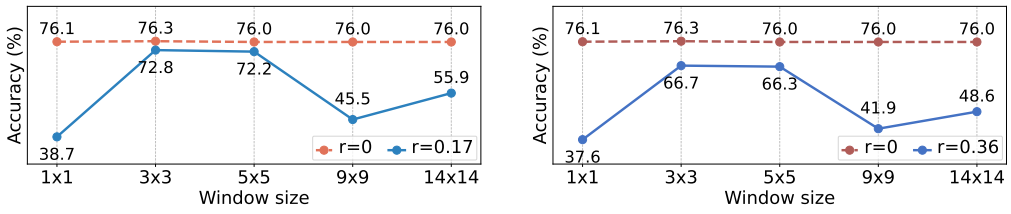
path1	path2	reduction ratio	FLOPs (G)	top-1 acc. (%)	△ (%)
D1	D2	0	1.45	76.1	0
		0.17	1.27	38.7	0
		0.27	1.13	41.6	0
		0.36	1.00	37.6	0
		0.46	0.86	34.2	0
D1	D2_D2	0	1.45	76.4	0.3↑
		0.17	1.27	66.5	27.8↑
		0.27	1.13	65.1	23.5↑
		0.36	1.00	61.2	23.6↑
		0.46	0.86	56.9	22.7↑
D1	D2_rand	0	1.45	75.8	0.3↓
		0.17	1.27	69.0	30.3↑
		0.27	1.13	67.0	25.4↑
		0.36	1.00	63.6	26.0↑
		0.46	0.86	59.3	25.1↑
D1	D2_D4	0	1.45	76.3	0.2↑
		0.17	1.27	72.8	34.1↑
		0.27	1.13	70.3	28.7↑
		0.36	1.00	66.7	29.1↑
		0.46	0.86	61.3	27.1↑

focuses on dual-path configurations. For clarity, the hyphen symbol "-" represents the connection between the two paths, with the ends indicating the directions of the main paths. The underscore symbol "_" denotes the window-aware scanning mechanism, where the left end indicates the main path direction and the right end reflects the scanning direction within the window. Here, D1-D2 represents Vim [6], while D1-D2_D4 corresponds to AMVim. Additionally, the pruning-free performance discussed in this study is defined as the performance at a reduction ratio of 0.

Effect of asymmetry. Table 5 compares the performance of symmetric (D1-D2) and asymmetric paths (D1-D4 and D1-rand). While D1-D4 exhibits a marginal 0.1% decrease in pruning-free accuracy compared to D1-D2, it achieves consistent improvements of $\geq$10% across all reduction ratios, with a notable 17.2% gain at a reduction ratio of 0.17.

To further validate the importance of asymmetry, we introduce a random scanning direction in the second path (D1-rand). This configuration yields the lowest pruning-free performance, as the random path struggles to capture token relationships and impedes the training process. Despite its reduced pruning robustness relative to D1-D4, D1-rand still significantly outperforms the symmetric D1-D2. These results conclusively demonstrate that asymmetric path designs are essential for enhancing the pruning resilience of Mamba-based architectures.

Effect of window-aware scanning. Table 6 presents the ablation study results for the window-aware scanning mechanism, with global scanning directions aligned with Vim (D1-D2). For simplicity, this study focuses on applying the window-aware scanning mechanism to the second path, as similar trends are observed in the first path (see Table 10 in the Appendix).



(a) reduction ratio = 0.17

(b) reduction ratio = 0.36

Figure 6: Ablation study on the impact of window size on top-1 accuracy (%) during token reduction. The symbol 'r' represents the reduction ratio. Across different reduction ratios, moderate window sizes (i.e., 3×3 and 5×5) consistently demonstrate superior pruning resistance.

Table 7: Ablation study on the impact of multi-scale paths on top-1 accuracy (%) during token reduction. Red text in brackets indicates window sizes. The single-path window-aware design demonstrates superior robustness than the dual-path mechanism, despite both retaining asymmetric multi-scale properties.

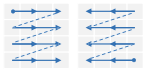
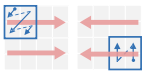
path1	path2	reduction ratio	FLOPs (G)	top-1 acc (%)	△ (%)
		0	1.45	76.1	0
		0.17	1.27	38.7	0
		0.27	1.13	41.6	0
		0.36	1.00	37.6	0
D1	D2	0.46	0.86	34.2	0
		0	1.45	75.2	0.8↓
		0.17	1.27	69.9	31.3↑
		0.27	1.13	66.6	25.0↑
		0.36	1.00	62.8	25.3↑
D1_D5(5)D2_D4(3)		0.46	0.86	58.1	23.9↑
		0	1.45	76.3	0.2↑
		0.17	1.27	72.8	34.1↑
		0.27	1.13	70.3	28.7↑
		0.36	1.00	66.7	29.1↑
D1	D2_D4	0.46	0.86	61.3	27.1↑

Table 8: Ablation study on the impact of window scanning direction on top-1 accuracy (%) during token reduction. The symbol △ quantifies performance differences relative to Vim (i.e., D1-D2). Distinct scanning directions between the window and global path ensure superior resistance to pruning.

path1	path2	reduction ratio	FLOPs (G)	top-1 acc. (%)	△ (%)
		0	1.45	76.1	0
		0.17	1.27	38.7	0
		0.27	1.13	41.6	0
		0.36	1.00	37.6	0
D1	D2	0.46	0.86	34.2	0
		0	1.45	76.4	0.3↑
		0.17	1.27	66.5	27.8↑
		0.27	1.13	65.1	23.5↑
		0.36	1.00	61.2	23.6↑
D1	D2_D2	0.46	0.86	56.9	22.7↑
		0	1.45	76.2	0.1↑
		0.17	1.27	73.0	34.3↑
		0.27	1.13	71.0	29.4↑
		0.36	1.00	67.6	30.1↑
D1	D2_D6	0.46	0.86	62.3	28.1↑
		0	1.45	76.3	0.2↑
		0.17	1.27	72.8	34.1↑
		0.27	1.13	70.3	28.7↑
		0.36	1.00	66.7	29.1↑
D1	D2_D4	0.46	0.86	61.3	27.1↑

The results reveal that introducing a same-direction window-aware scanning mechanism (D1-D2_D2) already significantly enhances pruning robustness, underscoring the importance of local dependency modeling in improving architectural robustness. When the main path and window scanning directions are further diversified (D1-D2_D4), performance improves by an additional 5% compared to D1-D2_D2, reinforcing the effectiveness of asymmetric design in boosting pruning resilience. Notably, the pruning-free performance increases by 0.3% for D1-D2_D2 and 0.2% for D1-D2_D4, highlighting the mechanism's ability to enhance model representation through local dependency capture.

When the window scanning direction is randomized (D1-D2_rand), its pruning robustness lies between D1-D2_D2 and D1-D2_D4, underscoring the irreplaceability of asymmetry and the importance of semantic relationship modeling. Additionally, the pruning-free performance of D1-D2_rand decreases by only 0.3%, demonstrating that the window mechanism effectively mitigates randomness by coordinating global and local information flows.

Effect of window size.

Window size is a critical hyperparameter introduced in this study. To assess its impact on performance, we conducted extensive ablation studies, as shown in Figure 6. In these experiments, the global path remains aligned with Vim (D1-D2), while the second path incorporates the window-aware scanning mechanism. Notably, a window size of 1×1 corresponds to the D1-D2 configuration, whereas a window size of 14×14 corresponds to the D1-D4 configuration.

The results reveal that the window size has a relatively minor effect on pruning-free performance, with optimal performance achieved at 3×3. However, token reduction performance exhibits significant fluctuations as the window size varies, with this trend remaining consistent across all reduction ratios (r = 0.17 in Figure 6a and r = 0.36 in Figure 6b). The optimal pruning robustness is observed at intermediate window sizes (3×3 and 5×5), whereas excessively small or large windows all lead to a performance collapse.

This behavior is expected, as the window-aware mechanism is specifically designed to capture local dependencies. However, larger window sizes exacerbate Mamba's long-range forgetting issue [1, 35], reducing its effectiveness. Furthermore, the poor robustness of the 1×1 and 14×14 window sizes results from their deviation from the multi-scale path design. These findings demonstrate that careful window size selection is essential for harmonizing local dependencies with global context, a key factor in maximizing the efficacy of the multi-scale architecture.

Effect of multi-scale path. The proposed design integrates window-aware scanning into one path

while retaining sequential scanning in the other, forming asymmetric multi-scale paths. This raises a natural question: can a dual-path window-aware mechanism achieve comparable performance by varying window sizes?

As shown in Table 7, the dual-path window-aware configuration (D1_D5(5)-D2_D4(3)) underperforms the single-path window-aware design (D1-D2_D4), despite both maintaining asymmetric multi-scale properties. This performance gap arises from the homogeneous scanning mechanisms in dual window-aware paths. While varying window sizes introduce scale diversity, the lack of receptive field heterogeneity limits their ability to capture hierarchical spatial dependencies. In contrast, the single-path window-aware design achieves superior robustness through heterogeneous scanning mechanisms, balancing local granularity and global continuity.

Effect of scan direction. Table 8 summarizes ablation studies on the scanning direction within windows, with the global scanning direction fixed to align with Vim (D1-D2). The results show that when the window and main path share the same scanning direction (D1-D2_D2), accuracy drops by approximately 7% compared to heterogeneous scanning directions (D1-D2_D6 and D1-D2_D4). Although D1-D2_D2 maintains an asymmetric path design, its diversity in token connections remains significantly lower than that of the heterogeneous configurations, which primarily accounts for its inferior performance.

Additionally, we observe that as long as the scanning direction within the window differs from the main path, the performance remains largely comparable (i.e., D1-D2_D6 vs. D1-D2_D4). While D1-D2_D6 exhibits slightly better pruning robustness, D1-D2_D4 offers a marginal pruning-free performance improvement of 0.1%. In this study, we adopt D1-D2_D4 as our default architecture, but we will release weights for both configurations to allow flexibility in choice. These findings collectively demonstrate that pruning robustness is ensured by heterogeneous scanning directions between the window and main path, regardless of the specific direction chosen.

5 Conclusion

In this work, we propose AMVim, a vision Mamba architecture designed for pruning robustness. By incorporating a window-aware scanning mechanism into one of paths, AMVim enables asymmetric multi-scale scanning. Extensive experiments have verified the effectiveness and high robustness of AMVim, laying a foundation for future research on the stability of SSM-based models.
Limitations. Due to resource constraints, we have not yet fully explored the scalability of AMVim, such as validating its efficacy on multi-task scenarios or integrating it with cross-modal frameworks. Despite these limitations, our experiments conclusively demonstrate AMVim's superiority in pruning robustness and spatial modeling, and we plan to investigate its broader applicability in future work.

Acknowledgments and Disclosure of Funding

This work was supported by National Major Scientific Instrument and Equipment Development Project of National Natural Science Foundation of China under Grant 62427820; in part by National Natural Science Foundation of China under Grant 62306198 and in part by Natural Science Foundation of Sichuan Province under Grant 2024NSFSC1468.

References

[1] Hang Guo, Jinmin Li, Tao Dai, Zhihao Ouyang, Xudong Ren, and Shu-Tao Xia. Mambair: A simple baseline for image restoration with state-space model. In *European conference on computer vision*, pages 222–241. Springer, 2024.

[2] Tianxiang Chen, Zi Ye, Zhentao Tan, Tao Gong, Yue Wu, Qi Chu, Bin Liu, Nenghai Yu, and Jieping Ye. Mim-istd: Mamba-in-mamba for efficient infrared small target detection. *IEEE Transactions on Geoscience and Remote Sensing*, 2024.

[3] Dingkang Liang, Xin Zhou, Wei Xu, Xingkui Zhu, Zhikang Zou, Xiaoqing Ye, Xiao Tan, and Xiang Bai. Pointmamba: A simple state space model for point cloud analysis. *arXiv preprint arXiv:2402.10739*, 2024.

[4] Kunchang Li, Xinhao Li, Yi Wang, Yinan He, Yali Wang, Limin Wang, and Yu Qiao. Videomamba: State space model for efficient video understanding. In *European Conference on Computer Vision*, pages 237–255. Springer, 2024.

[5] Vincent Tao Hu, Stefan Andreas Baumann, Ming Gui, Olga Grebenkova, Pingchuan Ma, Johannes Fischer, and Bjorn Ommer. Zigma: Zigzag mamba diffusion model. *arXiv e-prints*, pages arXiv–2403, 2024.

[6] Lianghui Zhu, Bencheng Liao, Qian Zhang, Xinlong Wang, Wenyu Liu, and Xinggang Wang. Vision mamba: Efficient visual representation learning with bidirectional state space model. *arXiv preprint arXiv:2401.09417*, 2024.

[7] Yue Liu, Yunjie Tian, Yuzhong Zhao, Hongtian Yu, Lingxi Xie, Yaowei Wang, Qixiang Ye, and Yunfan Liu. Vmamba: Visual state space model 2024. *arXiv preprint arXiv:2401.10166*, 2024.

[8] Albert Gu, Isys Johnson, Karan Goel, Khaled Saab, Tri Dao, Atri Rudra, and Christopher Ré. Combining recurrent, convolutional, and continuous-time models with linear state space layers. *Advances in neural information processing systems*, 34:572–585, 2021.

[9] Yuhong Li, Tianle Cai, Yi Zhang, Deming Chen, and Debadeepta Dey. What makes convolutional models great on long sequence modeling? *arXiv preprint arXiv:2210.09298*, 2022.

[10] Antonio Orvieto, Samuel L Smith, Albert Gu, Anushan Fernando, Caglar Gulcehre, Razvan Pascanu, and Soham De. Resurrecting recurrent neural networks for long sequences. In *International Conference on Machine Learning*, pages 26670–26698. PMLR, 2023.

[11] Xuanhua He, Ke Cao, Jie Zhang, Keyu Yan, Yingying Wang, Rui Li, Chengjun Xie, Danfeng Hong, and Man Zhou. Pan-mamba: Effective pan-sharpening with state space model. *Information Fusion*, 115:102779, 2025.

[12] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021.

[13] Alexey Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[14] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International conference on machine learning*, pages 10347–10357. PMLR, 2021.

[15] Yongming Rao, Wenliang Zhao, Benlin Liu, Jiwen Lu, Jie Zhou, and Cho-Jui Hsieh. Dynamicvit: Efficient vision transformers with dynamic token sparsification. *Advances in neural information processing systems*, 34:13937–13949, 2021.

[16] Hongxu Yin, Arash Vahdat, Jose M Alvarez, Arun Mallya, Jan Kautz, and Pavlo Molchanov. A-vit: Adaptive tokens for efficient vision transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10809–10818, 2022.

[17] Zhenglun Kong, Peiyan Dong, Xiaolong Ma, Xin Meng, Wei Niu, Mengshu Sun, Xuan Shen, Geng Yuan, Bin Ren, Hao Tang, et al. Spvit: Enabling faster vision transformers via latency-aware soft token pruning. In *European conference on computer vision*, pages 620–640. Springer, 2022.

[18] Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. Token merging: Your vit but faster. *arXiv preprint arXiv:2210.09461*, 2022.

[19] Youwei Liang, Chongjian Ge, Zhan Tong, Yibing Song, Jue Wang, and Pengtao Xie. Not all patches are what you need: Expediting vision transformers via token reorganizations. *arXiv preprint arXiv:2202.07800*, 2022.

[20] Shufan Li, Harkanwar Singh, and Aditya Grover. Mamba-nd: Selective state space modeling for multi-dimensional data. In *European Conference on Computer Vision*, pages 75–92. Springer, 2024.

[21] Xiaohuan Pei, Tao Huang, and Chang Xu. Efficientvmamba: Atrous selective scan for light weight visual mamba. *arXiv preprint arXiv:2403.09977*, 2024.

[22] Badri N Patro and Vijay S Agneeswaran. Simba: Simplified mamba-based architecture for vision and multivariate time series. *arXiv preprint arXiv:2403.15360*, 2024.

[23] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

[24] Harsh Mehta, Ankit Gupta, Ashok Cutkosky, and Behnam Neyshabur. Long range language modeling via gated state spaces. *arXiv preprint arXiv:2206.13947*, 2022.

[25] Jue Wang, Wentao Zhu, Pichao Wang, Xiang Yu, Linda Liu, Mohamed Omar, and Raffay Hamid. Selective structured state-spaces for long-form video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6387–6397, 2023.

[26] Daniel Y Fu, Tri Dao, Khaled K Saab, Armin W Thomas, Atri Rudra, and Christopher Ré. Hungry hungry hippos: Towards language modeling with state space models. *arXiv preprint arXiv:2212.14052*, 2022.

[27] Jimmy TH Smith, Andrew Warrington, and Scott W Linderman. Simplified state space layers for sequence modeling. *arXiv preprint arXiv:2208.04933*, 2022.

[28] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pvt v2: Improved baselines with pyramid vision transformer. *Computational visual media*, 8(3):415–424, 2022.

[29] Daquan Zhou, Bingyi Kang, Xiaojie Jin, Linjie Yang, Xiaochen Lian, Zihang Jiang, Qibin Hou, and Jiashi Feng. Deepvit: Towards deeper vision transformer. *arXiv preprint arXiv:2103.11886*, 2021.

[30] Jindi Lv, Yuhao Zhou, Yuxin Tian, Qing Ye, Wentao Feng, and Jiancheng Lv. Hypernas: Enhancing architecture representation for nas predictor via hypernetwork. *arXiv preprint arXiv:2509.18151*, 2025.

[31] Eric Nguyen, Karan Goel, Albert Gu, Gordon Downs, Preey Shah, Tri Dao, Stephen Baccus, and Christopher Ré. S4nd: Modeling images and videos as multidimensional signals with state spaces. *Advances in neural information processing systems*, 35:2846–2861, 2022.

[32] Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. *arXiv preprint arXiv:2111.00396*, 2021.

[33] Chenhongyi Yang, Zehui Chen, Miguel Espinosa, Linus Ericsson, Zhenyu Wang, Jiaming Liu, and Elliot J Crowley. Plainmamba: Improving non-hierarchical mamba in visual recognition. *arXiv preprint arXiv:2403.17695*, 2024.

[34] Tao Huang, Xiaohuan Pei, Shan You, Fei Wang, Chen Qian, and Chang Xu. Localmamba: Visual state space model with windowed selective scan. *arXiv preprint arXiv:2403.09338*, 2024.

[35] Yuheng Shi, Minjing Dong, and Chang Xu. Multi-scale vmamba: Hierarchy in hierarchy visual state space model. *arXiv preprint arXiv:2405.14174*, 2024.

[36] Mengzhao Chen, Wenqi Shao, Peng Xu, Mingbao Lin, Kaipeng Zhang, Fei Chao, Rongrong Ji, Yu Qiao, and Ping Luo. Diffrate: Differentiable compression rate for efficient vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 17164–17174, 2023.

[37] Dmitrii Marin, Jen-Hao Rick Chang, Anurag Ranjan, Anish Prabhu, Mohammad Rastegari, and Oncel Tuzel. Token pooling in vision transformers. *arXiv preprint arXiv:2110.03860*, 2021.

[38] Jiarui Xu, Shalini De Mello, Sifei Liu, Wonmin Byeon, Thomas Breuel, Jan Kautz, and Xiaolong Wang. Groupvit: Semantic segmentation emerges from text supervision. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18134–18144, 2022.

[39] Michael S Ryoo, AJ Piergiovanni, Anurag Arnab, Mostafa Dehghani, and Anelia Angelova. Tokenlearner: What can 8 learned tokens do for images and videos? *arXiv preprint arXiv:2106.11297*, 2021.

[40] Zheng Zhan, Zhenglun Kong, Yifan Gong, Yushu Wu, Zichong Meng, Hangyu Zheng, Xuan Shen, Stratis Ioannidis, Wei Niu, Pu Zhao, et al. Exploring token pruning in vision state space models. *arXiv preprint arXiv:2409.18962*, 2024.

[41] Zheng Zhan, Yushu Wu, Zhenglun Kong, Changdi Yang, Yifan Gong, Xuan Shen, Xue Lin, Pu Zhao, and Yanzhi Wang. Rethinking token reduction for state space models. *arXiv preprint arXiv:2410.14725*, 2024.

[42] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Antonio Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. In *IJCV*, 2019.

[43] Tete Xiao, Yingcheng Liu, Bolei Zhou, Masayoshi Tomizuka, and Fisher Yu. Unified perceptual parsing for scene understanding. In *ECCV*, 2018.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction accurately capture the paper's contributions and scope, with claims supported by results.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discusses its limitations in the conclusion, providing transparency about the boundaries of the work and guiding future research directions.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: For each theoretical result, the paper provides the full set of assumptions and a complete and correct proof.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The method pipeline and experimental details are presented along with corresponding reproducible credentials.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: All data sources used in this study are clearly cited in the paper, and the code will be uploaded in a zipped format.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Comprehensive training and testing details are provided in the Appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: The results include the standard deviation calculated from multiple random runs.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide detailed information about computational resources in the experimental settings.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research presented in this paper complies with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The paper addresses both the potential positive contributions and possible negative societal impacts of the research.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper poses no such risks

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The original owners of the data and models used in the paper are properly credited, and the licenses and terms of use are clearly stated and fully respected.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The new assets introduced in the paper are thoroughly documented and made available along with the existing ones.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

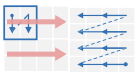
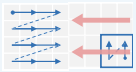
A Implement Details

AMVim is fine-tuned using the pretrained weights from Vim. The detailed training configurations are presented in Table 9. All other settings are aligned with those of Vim.

Table 9: Training settings for AMVim on ImageNet-1K.

finetune config	**AMVim-T**	**AMVim-S**
optimizer	AdamW	AdamW
base learning rate	4e-5	1e-5
minimal learning rate	1e-5	5e-6
weight decay	1e-8	0.05
optimizer momentum	$\beta 1$,$\beta 2$=0.9,0.999	$\beta 1$,$\beta 2$=0.9,0.999
batch size	1024	1024
training epochs	150	150
learning rate schedule	cosine decay	cosine decay
warmup epochs	5	5
warmup learning rate	1e-5	1e-5
warmup schedule	linear	linear
drop path	0	0.3
mixup	0.8	0.8
cutmix	1	1
EMA	None	None

Table 10: Comparison results with window-aware scanning mechanism applied on various paths. The window-aware mechanism on the second path exhibits stronger pruning resistance than on the first path.

path1	path2	red. ratio	FLOPs(G)	top-1 acc(%)	△(%)
		0	1.45	76.1	0
		0.17	1.27	38.7	0
		0.27	1.13	41.6	0
		0.36	1.00	37.6	0
D1	D2	0.46	0.86	34.2	0
		0	1.45	76.4	0.3↑
		0.17	1.27	72.3	33.6↑
		0.27	1.13	70.2	28.6↑
		0.36	1.00	65.4	27.8↑
D1_D3	D2	0.46	0.86	56.8	22.6↑
		0	1.45	76.3	0.2↑
		0.17	1.27	72.8	34.1↑
		0.27	1.13	70.3	28.7↑
		0.36	1.00	66.7	29.1↑
D1	D2_D4	0.46	0.86	61.3	27.1↑

B More Experiments

Impact of window-aware on various paths. To further validate the superiority of our design, we applied the window-aware scanning mechanism to the first path (D1_D3-D2) for comparison. As shown in Table 10, both configurations exhibit comparable performance at reduction ratios of 0.17 and 0.27. However, as the reduction ratio increases, the D1-D2_D4 configuration significantly outperforms D1_D3-D2. This indicates that while the window-aware mechanism benefits either path, its integration into the second path yields optimal performance, particularly under aggressive pruning scenarios. **Impact of the global scanning direction.** In this work, we align the global scanning direction with Vim (i.e., D1-D2). To investigate the impact of global scanning direction, we conducted experiments by changing the global scanning direction of the second path from D2 to D4. As shown in Table 11, when the intra-window scanning direction is set to D2 (D1-D4_D2), pruning robustness improves but remains inferior to configurations with intra-window direction D6 (D1-D4_D6). This performance gap likely stems from the significant overlap in token connections and receptive fields between the D1 and D2 directions, limiting their complementary information flow.

Furthermore, while D1-D4_D6 achieves notable improvements in pruning robustness, it still underperforms our design with global direction D2 (D1-D2_D4) by approximately 2% in accuracy. These results demonstrate that the choice of global scanning direction is critical when combined with the window-aware mechanism.

Impact of sequential scanning path. To further investigate the role of the sequential scanning branch, we implemented the window-aware scanning mechanism on both paths. As shown in Table 12, regardless of the intra-window scanning direction, these configurations achieve comparable pruning robustness. While they significantly outperform Vim (which relies solely on sequential scanning) due to the stability introduced by the window-aware mechanism in capturing local dependencies, they exhibit performance degradation in pruning-free settings compared to Vim. This underscores the importance of sequential scanning for preserving base performance.

The proposed architecture combines sequential scanning with window-aware scanning, inheriting the strengths of both strategies. This combination achieves an optimal balance between pruning robustness and model performance.

Table 11: Ablation study on the impact of global scanning directions. △ denotes the performance gap relative to Vim (i.e., D1-D2). When integrated with the window-aware scanning mechanism, the global scanning direction D2 outperforms D4.

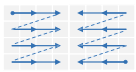
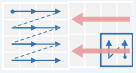
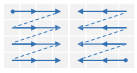
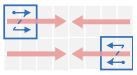
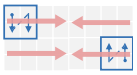
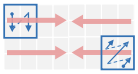
path1	path2	red. ratio	FLOPs(G)	top-1 acc(%)	△(%)
		0	1.45	76.1	0
		0.17	1.27	38.7	0
		0.27	1.13	41.6	0
		0.36	1.00	37.6	0
D1	D2	0.46	0.86	34.2	0
		0	1.45	76.2	0.1↑
		0.17	1.27	59.7	21.1↑
		0.27	1.13	59.2	17.6↑
		0.36	1.00	54.6	17.0↑
D1	D4_D2	0.46	0.86	51.1	17.0↑
		0	1.45	76.0	0.1↓
		0.17	1.27	70.9	32.3↑
		0.27	1.13	68.5	26.9↑
		0.36	1.00	64.0	26.5↑
D1	D4_D6	0.46	0.86	58.7	24.5↑
		0	1.45	76.3	0.2↑
		0.17	1.27	72.8	34.1↑
		0.27	1.13	70.3	28.7↑
		0.36	1.00	66.7	29.1↑
D1	D2_D4	0.46	0.86	61.3	27.1↑

Table 12: Ablation study on the impact of the sequential scanning path. △ represents the performance gap relative to Vim (i.e., D1-D2). While the dual-path window-aware design demonstrates superior pruning robustness compared to Vim, it suffers from performance degradation in pruning-free settings.

path1	path2	red. ratio	FLOPs(G)	top-1 acc(%)	△(%)
		0	1.45	76.1	0
		0.17	1.27	38.7	0
		0.27	1.13	41.6	0
		0.36	1.00	37.6	0
D1	D2	0.46	0.86	34.2	0
		0	1.45	75.8	0.3↓
		0.17	1.27	69.8	31.2↑
		0.27	1.13	67.1	25.5↑
		0.36	1.00	63.2	25.6↑
D1_D1	D2_D2	0.46	0.86	58.3	24.2↑
		0	1.45	75.7	0.4↓
		0.17	1.27	70.3	31.6↑
		0.27	1.13	66.6	25.0↑
		0.36	1.00	61.8	24.3↑
D1_D3	D2_D4	0.46	0.86	56.3	22.1↑
		0	1.45	75.4	0.6↓
		0.17	1.27	70.6	31.9↑
		0.27	1.13	67.5	25.9↑
		0.36	1.00	63.5	25.9↑
D1_D3	D2_D6	0.46	0.86	58.3	24.2↑