PRETRAINED MAMBA-BASED VISION BACKBONES CAN EXHIBIT COLLAPSED TIMESCALE BANKS: CHECKPOINT-AWARE STATE EXPANSION WITH SPANINIT

Anonymous authors
Paper under double-blind review

ABSTRACT

Expanding the recurrent state of a pretrained Mamba requires initializing the decay timescale of each added state dimension. Additional-scan, a parameter-efficient state-expansion method, copies a pretrained entry of the state matrix A, so each added state begins at a decay scale already present in that channel's timescale bank. We show that the effect of this initialization varies with the timescale geometry inherited from pretraining. Because within-channel timescale ratios depend only on A, the bank's relative span can be read directly from a checkpoint without data or a forward pass. Across released Mamba-based vision checkpoints, ImageNet-1k-only models with multiple state dimensions per channel exhibit near-collapsed timescale banks, whereas models exposed to ImageNet-21k have substantially wider spans. Motivated by this diagnosis, we propose **SpanInit**, which fills the largest uncovered gaps within the scan-relevant range $[1, L]$ in log-timescale space. SpanInit changes only the initial A values and uses a single calibration forward pass to set absolute timescales, adding no trainable parameters or inference cost. On VTAB-1k, SpanInit improves Additional-scan on the ImageNet-21k-pretrained MambaVision-B/L checkpoint but degrades it on the ImageNet-1k-only checkpoint of the same architecture. These contrasting outcomes indicate that state-expansion initialization should account for the pretrained checkpoint's timescale geometry rather than follow a single checkpoint-agnostic rule.

1 INTRODUCTION

Vision backbones based on selective state space models (Gu & Dao, 2023; Dao & Gu, 2024), including Vision Mamba (Vim; Zhu et al., 2024), VMamba (Liu et al., 2024b), and MambaVision (Hatamizadeh & Kautz, 2025), are increasingly adapted through parameter-efficient fine-tuning (PEFT) (Yoshimura et al., 2025). Yoshimura et al. (2025) introduced Additional-scan, which appends N' state dimensions to each SSM block while keeping pretrained parameters frozen. Each added state requires an initial entry in the diagonal state matrix A, whose magnitude controls its decay after discretization. Initializing these entries therefore determines the decay scales from which the added states begin adaptation.

Mamba initializes A with the S4D-Real spectrum $A_n = -(n+1)$ for $n = 0, \ldots, N-1$ (Gu et al., 2022a; Gu & Dao, 2023); larger $|A_n|$ yields faster decay. The N state dimensions of a channel therefore begin with distinct decay rates, forming a *timescale bank*. Additional-scan does not restore this spectrum when adding states; instead, it copies a pretrained entry of A, motivated by the fact that pretraining changes the original S4D values (Yoshimura et al., 2025). This raises a basic question: *what timescale structure does a pretrained vision checkpoint actually retain?*

Prior work studies initialization of A for training from scratch (Solozabal et al., 2025; Trockman et al., 2024) or rescales pretrained spectra for long-context language modeling (Lu et al., 2025), but does not characterize the within-channel bank left by pretraining in released vision checkpoints.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

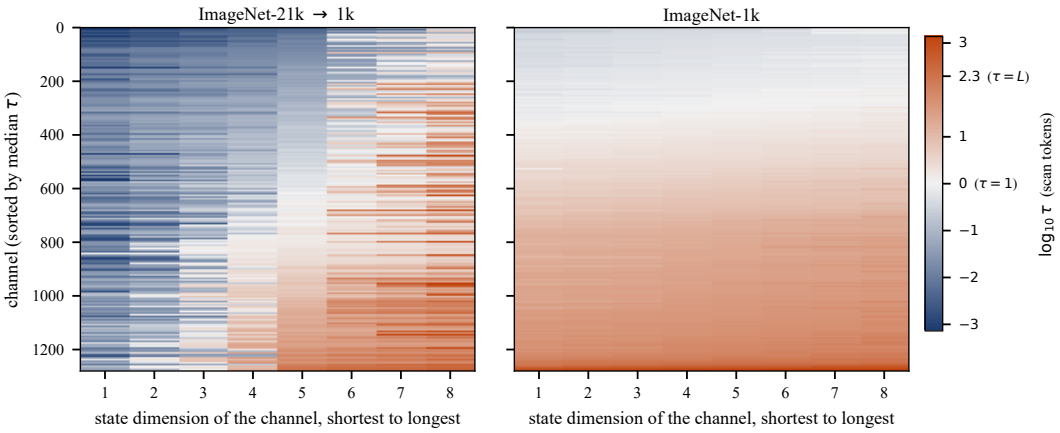


Figure 1: Pretrained timescale banks in five 256-channel MambaVision-B SSM blocks for Caltech101. Color denotes $\log_{10}\tau$, where $\tau = 1/(\bar{\Delta}|A|)$. The ImageNet-21k-pretrained checkpoint has a wide within-channel span, whereas the ImageNet-1k-only checkpoint is nearly collapsed; this relative span is independent of the channel-wise $\bar{\Delta}$.

We characterize this structure across nine released Mamba-based vision checkpoints. Within a channel, timescale ratios depend only on A because the input-dependent step Δ is shared across state dimensions and cancels from the ratio. The *relative span* of a timescale bank can therefore be read directly from a checkpoint without data or a forward pass. Among the eight checkpoints with more than one state dimension per channel, all five ImageNet-1k-only checkpoints have near-collapsed spans, whereas the three exposed to ImageNet-21k have spans exceeding three orders of magnitude. The same separation appears across the B, L, and L2 sizes of MambaVision, so the pattern does not track model size alone. We treat this geometry as a property of the checkpoint, without claiming that a collapsed bank is intrinsically worse or that pretraining data alone causes it.

Copying has a direct limitation: although Additional-scan gives each new state its own input and output projections, the copied A entry starts it at a decay scale already present in the channel. Because Δ is shared within a channel, input dependence can shift the bank along the timescale axis but cannot change its relative span. These timescales characterize recurrent *decay geometry*, not complete functional influence, which also depends on selective input and output terms. Moreover, wide relative span does not imply coverage of the finite scan: in the five $L = 196$ blocks of the ImageNet-21k-pretrained MambaVision-B/L, most pretrained timescales lie below one scan token.

We therefore propose **SpanInit**, which uses pretrained timescales inside the scan-relevant interval $[1, L]$ as anchors and fills the largest uncovered gaps in log-timescale space. Each new state is placed at the geometric midpoint of the widest remaining gap. SpanInit changes only the initial A values of the states Additional-scan adds, adding neither trainable parameters nor inference cost. Absolute placement requires the channel-wise mean step $\bar{\Delta}$, estimated per task from one forward pass over the 1,000 training images. On VTAB-1k with MambaVision-B/L, SpanInit changes the 19-task mean of Additional-scan by $+0.39/+0.59$ percentage points on the ImageNet-21k-pretrained checkpoints and $-0.38/-0.21$ points on their ImageNet-1k-only counterparts, averaged over three seeds. These contrasting outcomes motivate checkpoint-aware state-expansion initialization that accounts for the timescale geometry inherited from pretraining.

Contributions. Our contributions are threefold. (i) We characterize the timescale banks of nine pretrained Mamba-based vision checkpoints and show that their within-channel relative span can be read from A alone, revealing near-collapsed and widely spread banks. (ii) We show that copying a pretrained A entry introduces no new within-channel decay scale and that the shared step Δ cannot change the bank's relative span. Based on this diagnosis, we introduce SpanInit, which fills uncovered regions of the scan-relevant range without additional trainable parameters or inference cost. (iii) Across identical MambaVision-B/L architectures with contrasting pretrained bank geometries, SpanInit yields opposite changes in VTAB-1k performance, motivating checkpoint-aware state-expansion initialization.

2 Related Work

Parameter-efficient fine-tuning for transformers. PEFT adapts pretrained models by updating a small subset of parameters or adding lightweight trainable modules (Lialin et al., 2023; Han et al., 2024). Approaches tune bias terms (Zaken et al., 2022), add adapters (Houlsby et al., 2019; Chen et al., 2022; Luo et al., 2023), optimize visual prompts (Jia et al., 2022), learn low-rank or factorized updates (Hu et al., 2022; Liu et al., 2024a; Jie & Deng, 2023), or scale and shift features (Lian et al., 2022). Closest in spirit is PiSSA (Meng et al., 2024), which initializes LoRA's low-rank factors from principal components of pretrained weights. These methods adapt existing weights, features, or token representations rather than initialize newly added recurrent states and their decay scales.

Parameter-efficient fine-tuning for Mamba. MambaPEFT (Yoshimura et al., 2025) systematically studies PEFT for Mamba and introduces Additional-scan, the state-expansion method on which we build. Other Mamba/SSM-specific approaches adapt different components: ProDiaL adapts pretrained projectors (Ham et al., 2025); Sparse Dimension Tuning updates selected existing SSM dimensions (Galim et al., 2025); State-offset Tuning modifies the recurrent state (Kang et al., 2025); and Memba combines membrane-inspired recurrent gating with low-rank adaptation (Lee et al., 2026). Additional-scan instead creates new recurrent state dimensions, each requiring an initial A. Its default initialization copies a pretrained A entry and outperforms reapplying the original S4D initialization in its reported ablation (Yoshimura et al., 2025), but does not characterize the timescale geometry of the pretrained bank.

Initialization and timescales in state space models. Modern SSM initialization traces to HiPPO (Gu et al., 2020), LSSL (Gu et al., 2021), and S4 (Gu et al., 2022b). Diagonal variants such as DSS and S4D establish effective diagonal state-space parameterizations and highlight the importance of initialization (Gupta et al., 2022; Gu et al., 2022a). In S4-family parameterizations, log-spaced step sizes across channels provide a geometric range of temporal scales (Gu et al., 2021; 2023). Subsequent work studies generalized HiPPO parameterizations (Gu et al., 2023), approximate diagonalization (Yu et al., 2024), frequency-aware initialization of diagonal SSMs (Solozabal et al., 2025), and attention-mimicking initialization for Mamba (Trockman et al., 2024); these methods primarily concern models trained from scratch.

For pretrained Mamba models, MambaExtend calibrates scaling factors in the discretization module (Azizi et al., 2025), while LongMamba and DeciMamba modify effective receptive fields for long-context extrapolation (Ali et al., 2025; Ye et al., 2025; Ben-Kish et al., 2025). Mamba Modulation rescales the pretrained spectrum of A for length generalization (Lu et al., 2025). In contrast, we characterize the within-channel timescale geometry left by vision pretraining and use it to initialize newly added recurrent states.

3 Preliminaries

Selective state space models. A continuous-time state space model maps an input $x(t)$ to an output $y(t)$ through an N-dimensional hidden state $h(t)$:

$$h'(t) = Ah(t) + Bx(t), \qquad y(t) = Ch(t), \tag{1}$$

where A controls state dynamics and B and C write to and read from the state. After discretization with step Δ, $\bar{A} = \exp(\Delta A)$ and $\bar{B} \approx \Delta B$ (Liu et al., 2024b), giving

$$h_t = \bar{A}h_{t-1} + \bar{B}x_t, \qquad y_t = Ch_t. \tag{2}$$

Mamba (Gu & Dao, 2023) makes Δ, B, and C input-dependent while keeping A input-independent and diagonal. Consequently, the homogeneous decay is dimension-wise: each diagonal entry of A sets a continuous-time decay rate, while the discretized decay $\bar{A}_t = \exp(\Delta_t A)$ varies with the input-dependent step Δ_t.

Within-channel parameterization. For a block with D channels and scan length L, omitting batch dimensions,

$$A \in \mathbb{R}^{D\times N}, \qquad \Delta \in \mathbb{R}^{L\times D}, \qquad B, C \in \mathbb{R}^{L\times N}. \tag{3}$$

Table 1: Timescale-bank geometry of nine pretrained checkpoints. The within-channel span is $\max_n \tau_n / \min_n \tau_n = \max_n |A_n| / \min_n |A_n|$, so it needs only A. *Measured* is its geometric mean across channels, *at init.* its initial value, and $\geq 10\times$ the fraction of channels spanning at least $10\times$. The ImageNet-21k MambaVision-L2 is a 512^2 model, whose span is resolution-independent.

					bank span within a channel		
Backbone	Pretraining	N	channels	median $\lvert A\rvert$	at init.	measured	$\geq 10\times$
MambaVision-B	IN-21k $\to$ 1k	8	2,816	6.83	$8\times$	$\mathbf{3{,}527}\times$	$\mathbf{99.2}\%$
MambaVision-L	IN-21k $\to$ 1k	8	4,312	9.87	$8\times$	$\mathbf{3{,}335}\times$	$\mathbf{99.2}\%$
MambaVision-L2	IN-21k $\to$ 1k	8	4,704	10.71	$8\times$	$\mathbf{2{,}590}\times$	$\mathbf{99.6}\%$
MambaVision-B	IN-1k	8	2,816	1.01	$8\times$	$1.67\times$	0.0%
MambaVision-L	IN-1k	8	4,312	1.01	$8\times$	$1.43\times$	0.0%
MambaVision-L2	IN-1k	8	4,704	1.01	$8\times$	$1.37\times$	0.0%
Vim-S	IN-1k	16	36,864	1.00	$16\times$	$1.31\times$	0.0%
Vim-B	IN-1k	16	73,728	1.00	$16\times$	$1.22\times$	0.0%
VMamba-B	IN-1k	1	83,968	0.99	$1\times$	$1.00\times$	0.0%

Crucially, Δ has no state axis: all N state dimensions within a channel share the same step. Therefore, within-channel differences in decay are determined by A, and their relative decay scales are unaffected by Δ.

Vision scan length. Vision SSMs flatten feature maps into token sequences, with scan patterns varying across architectures. Vim uses bidirectional scans (Zhu et al., 2024), VMamba uses four directions (Liu et al., 2024b), and MambaVision uses a forward SSM scan alongside a non-SSM branch (Hatamizadeh & Kautz, 2025). In MambaVision-B/L at 224^2 resolution, the SSM stages operate at 14×14 and 7×7, giving scan lengths $L = 196$ and $L = 49$. We use L as the finite scan horizon when defining the scan-relevant timescale range.

4 Diagnosing the Timescale Banks of Pretrained Checkpoints

We characterize the decay geometry encoded by the pretrained state matrix A. While A fixes continuous-time decay rates, their locations in scan tokens also depend on the input-dependent step Δ. We therefore express decay in timescales relative to the scan length L.

4.1 Effective decay timescales

Conditioned on a realized step sequence $\{\Delta_t\}$, the homogeneous dynamics of Equation 2 satisfy

$$h_k = \left(\prod_{t=1}^{k} \exp(\Delta_t A) \right) h_0 = \exp\left(A \sum_{t=1}^{k} \Delta_t \right) h_0. \tag{4}$$

Let $\bar{\Delta}$ denote the empirical mean step of a channel; the bar here marks a token mean, not discretization. Since Mamba parameterizes $A = -\exp(A_{\log})$, we define the mean-step e-folding timescale of a state dimension as

$$\tau = \frac{1}{\bar{\Delta}|A|}, \tag{5}$$

measured in scan tokens. Under this mean-step approximation, $\tau < 1$ corresponds to decay below $1/e$ within one token, whereas $\tau > L$ retains more than $1/e$ over an L-token scan.

The decay factor depends on the product ΔA (Gu et al., 2023; Gu & Dao, 2023). Because Mamba initializes $A_n = -(n+1)$ (Gu et al., 2022a), the step alone does not determine the state timescales.

Absolute timescales require the input-dependent $\bar{\Delta}$, which we estimate once per channel with a single forward pass over the 1,000 training images. However, Δ has no state axis (Equation 3), so within a channel

$$\frac{\tau_n}{\tau_m} = \frac{|A_m|}{|A_n|}. \tag{6}$$

Hence the relative geometry of a timescale bank is determined by A alone; data are needed only to place the bank on the absolute τ axis. We define its *within-channel span* as $\max_n \tau_n / \min_n \tau_n = \max_n |A_n| / \min_n |A_n|$ and report the geometric mean across channels.

4.2 Collapsed and spread pretrained banks

Table 1 reveals two qualitatively different pretrained bank geometries. All five ImageNet-1k-only checkpoints with multiple state dimensions contract from their initial spectra to geometric-mean spans below $2\times$. In contrast, the three MambaVision checkpoints exposed to ImageNet-21k develop spans exceeding three orders of magnitude, with nearly all channels spanning at least $10\times$. This separation persists across the B, L, and L2 model sizes, indicating that it does not simply follow model scale. We therefore treat the bank geometry as a property of the checkpoint, without attributing the difference solely to the pretraining dataset or interpreting a collapsed bank as intrinsically inferior.

The state-matrix values reinforce this distinction: the five near-collapsed checkpoints concentrate tightly around $|A| = 1$, with median values of 1.00–1.01, indicating that the distinct decay rates present at initialization have become strongly compressed. VMamba has only one state dimension per channel, so its within-channel span is fixed at one by construction and is not informative for this comparison.

4.3 Bank span is not scan coverage

Relative span alone does not reveal where the bank lies within the finite scan. Table 2 shows that the widely spread ImageNet-21k-pretrained MambaVision-B places only about 39% of its pretrained states inside the scan-relevant interval $[1, L]$, with most of the remainder below one token. Its ImageNet-1k-only counterpart places about 75% inside the same interval despite having a nearly collapsed within-channel bank. MambaVision-L exhibits the same qualitative contrast.

Table 2: Placement of pretrained timescales in the five shallower blocks of each backbone, which share $L = 196$ (256 channels in MambaVision-B and 392 in MambaVision-L), averaged over the nineteen VTAB-1k tasks. The last row is the geometric mean of the longest channel-wise timescale.

	MambaVision-B		MambaVision-L	
	IN-21k → 1k	IN-1k	IN-21k → 1k	IN-1k
where the timescales sit				
dimensions with $\tau < 1$	58.1%	22.6%	62.2%	13.2%
dimensions with $1 \le \tau \le L$	38.6%	74.8%	34.0%	82.1%
dimensions with $\tau > L$	3.3%	2.6%	3.8%	4.7%
what the rule finds there				
dimensions in the window, per channel	3.09	5.98	2.72	6.57
longest τ in a channel	20.3	9.3	15.3	9.3

Tables 1 and 2 therefore capture complementary properties: within-channel diversity and placement relative to the finite scan. In the five $L = 196$ blocks, for example, the ImageNet-21k-pretrained MambaVision-B spans about $1{,}380\times$ within a channel, yet much of this spread lies below one token. Conversely, its ImageNet-1k-only counterpart places most states inside the scan window while providing little within-channel diversity. Figure 1 visualizes the same distinction: the ImageNet-21k-pretrained checkpoint varies primarily within channels, whereas the ImageNet-1k-only checkpoint is comparatively compressed within each channel. Thus, neither a large span nor a high fraction of states inside the scan window alone implies good coverage of that window. This motivates placing new states relative to the existing bank rather than reproducing a pretrained decay scale.

4.4 Copying adds no new decay scale

Additional-scan copies the first pretrained column of A for each of the N' added states (Yoshimura et al., 2025), so all added dimensions begin at the same pretrained timescale. Their B and C parameters are initialized separately, but copying introduces no new initial decay scale.

This choice inherits the original S4D ordering, in which the first column corresponds to the longest timescale. Pretraining, however, largely removes this privileged ordering. Across the full ImageNet-21k-pretrained MambaVision-B backbone, the first column is the longest-timescale state in only 17% of channels, only modestly above the 12.5% expected from a uniform choice among eight columns. Thus, copying neither introduces a new decay scale nor reliably selects an extreme of the pretrained bank. Together with the incomplete scan coverage above, this motivates initializing new states according to the geometry of the pretrained bank.

5 METHOD: SPANINIT

Section 4 shows that pretrained checkpoints can have very different within-channel geometries and that copying a pretrained A entry adds no new decay scale. We therefore propose **SpanInit**, which initializes new Additional-scan states by filling uncovered regions of $[1, L]$ relative to the decay scales already present in each channel. SpanInit changes only the initial A values of the added states; all trainable parameters, B and C initialization, and inference computation otherwise follow Additional-scan. Unless stated otherwise, we use $N' = 6$ added states per channel; MambaVision-B/L has $N = 8$ pretrained states.

Scan-relevant window. SpanInit places new states in

$$\mathcal{W} = [1, L], \tag{7}$$

with timescales measured in scan tokens (Section 4.1): below one token the previous state has already decayed past $1/e$, whereas beyond L it retains more than $1/e$ over the scan. We use $[1, L]$ as the finite-horizon range, a design choice rather than a claim that states outside it are unimportant.

Bank-aware placement. For channel d, we retain the pretrained timescales already inside the window as anchors and add the window boundaries as virtual anchors:

$$\mathcal{S}_d^{(0)} = \{1, L\} \cup \{\tau_{d,n} : \tau_{d,n} \in \mathcal{W}\}. \tag{8}$$

At each insertion step, sort the current anchors as $s_1 < \cdots < s_M$ and select the largest adjacent gap in log-timescale space,

$$i^\star = \arg\max_{1 \le i < M} \log \frac{s_{i+1}}{s_i}. \tag{9}$$

The new state is placed at the geometric midpoint,

$$\tau^{\text{new}} = \sqrt{s_{i^\star} s_{i^\star+1}}, \tag{10}$$

and inserted into the anchor set, repeating until N' new states are placed.

The log metric reflects the multiplicative nature of timescales: gaps with the same ratio have equal width. The geometric midpoint is the local minimax point,

$$\sqrt{ab} = \arg\min_{\tau \in (a,b)} \max \left\{ \log \frac{\tau}{a}, \log \frac{b}{\tau} \right\}, \tag{11}$$

halving the maximum log-distance to the gap's endpoints. Combined with Equation 9, each iteration greedily splits the largest uncovered interval. This is a greedy coverage rule; we do not claim global optimality. If no pretrained timescale lies in $[1, L]$, the procedure starts from the two boundary anchors alone. Figure 2 illustrates the six insertions for one channel: panel (a) shows the three pretrained anchors inside $[1, L]$, with five further timescales below the window; panels (b) and (c) show the first insertion at the geometric midpoint of the widest log-gap and the state after six insertions, which narrows that gap from $22\times$ to $2.2\times$.

Mapping back to the state matrix. Relative bank geometry follows from A alone, but absolute placement requires the channel-wise mean step $\bar{\Delta}_d$ (Section 4.1). We estimate it once before fine-tuning with one forward pass over the task's $1{,}000$ training images and use it only for initialization.

For each selected $\tau_{d,j}^{\text{new}}$,

$$|A_{d,j}^{\text{new}}| = \frac{1}{\bar{\Delta}_d \, \tau_{d,j}^{\text{new}}}, \qquad (A_{\log}^{\text{new}})_{d,j} = -\log\big(\bar{\Delta}_d \, \tau_{d,j}^{\text{new}}\big). \tag{12}$$

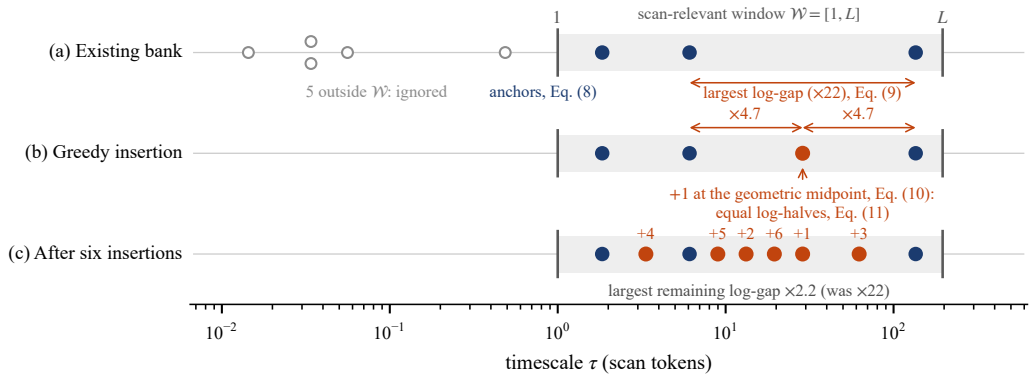


Figure 2: SpanInit on one MambaVision-B channel (ImageNet-21k, Caltech101). Pretrained timescales in $[1, L]$ serve as anchors, and each added state bisects the largest remaining gap in $\log \tau$.

This inverts Equation 5, so the initialized state has the selected mean-step timescale. All other initialization and training choices follow Additional-scan. SpanInit therefore requires one calibration forward pass but adds no trainable parameters or inference-time computation.

6 Experiments

6.1 Experimental setup

We evaluate on VTAB-1k (Zhai et al., 2019): nineteen classification tasks in three groups, 1,000 training images each, scored on the full test split. The backbones, MambaVision-B and MambaVision-L (Hatamizadeh & Kautz, 2025), pretrained on ImageNet-21k and fine-tuned on ImageNet-1k, are held frozen. Additional-scan adds $N' = 6$ state dimensions to each of their eight SSM blocks; those and the classification head are the only trained parameters. SpanInit changes only the initial A values of the added dimensions (Section 5).

We follow the VTAB-1k protocol of Yoshimura et al. (2025), including its optimizer settings, without per-task tuning, and evaluate at the final epoch. Accuracies are means over seeds 42, 43, and 44 with sample standard deviations; each Δ is the seed-paired difference from Additional-scan.

6.2 Experimental results

Main comparison. Table 3 isolates the effect of initialization: Additional-scan and SpanInit share all trainable parameters and differ only in the initial A. SpanInit raises the nineteen-task mean from 69.60 to 69.99 (+0.39) on MambaVision-B and from 70.21 to 70.80 (+0.59) on MambaVision-L. The effect is not uniform across groups: on MambaVision-B, the Natural and Structured means increase, whereas the Specialized mean decreases. We therefore interpret the aggregate difference as evidence that the initialization of added states can affect downstream performance, rather than as uniform superiority across tasks.

The effect varies across checkpoints. Table 4(a) provides an architecture-controlled comparison. With the same MambaVision-B architecture, SpanInit raises the nineteen-task mean on the ImageNet-21k-pretrained checkpoint (+0.39) but lowers it on the ImageNet-1k-only checkpoint (−0.38). These checkpoints exhibit the contrasting timescale-bank geometries identified in Table 1. Thus, SpanInit is not a universal replacement for copying; rather, its aggregate effect differs across checkpoints with contrasting inherited geometries. Because the checkpoints also differ in their pretraining histories, this comparison is associational rather than causal.

What matters in placement. Table 4(b) separates conditioning on the pretrained anchors from simply spreading the added states across $[1, L]$. Uniform log-spacing, which ignores the pretrained

Table 3: VTAB-1k on MambaVision-B and MambaVision-L (MV-B, MV-L), both pretrained on ImageNet-21k. Linear trains only the classification head; Params counts what Additional-scan and SpanInit each train besides it, with its percentage of the frozen backbone. Cells are means over three seeds with the sample standard deviation; Δ is the seed-paired difference from Additional-scan, which differs from SpanInit only in the initial A. *: $p < 0.05$, one-tailed Welch's t-test in the direction of Δ (Jia et al., 2022).

Natural

	Method	CIFAR-100	Caltech101	DTD	Flowers102	Pets	SVHN	Sun397	Mean
MV-B	Linear	58.94 ± 0.12	89.56 ± 0.09	75.18 ± 0.13	99.48 ± 0.03	93.32 ± 0.19	50.06 ± 0.15	54.63 ± 0.04	74.45
	Additional-scan	62.21 ± 0.41	89.67 ± 0.30	75.04 ± 0.16	99.34 ± 0.04	92.87 ± 0.18	73.25 ± 0.60	53.01 ± 0.14	77.91
	SpanInit	63.31 ± 0.56	91.27 ± 0.06	74.61 ± 0.40	99.31 ± 0.08	92.99 ± 0.23	75.90 ± 0.38	53.13 ± 0.36	78.64
	Δ	+1.10*	+1.59*	−0.43	−0.04	+0.12	+2.64*	+0.12	*+0.73*
MV-L	Linear	62.78 ± 0.16	90.02 ± 0.03	76.15 ± 0.37	99.67 ± 0.02	94.28 ± 0.14	53.60 ± 0.14	56.59 ± 0.11	76.16
	Additional-scan	64.56 ± 0.16	90.37 ± 0.48	75.66 ± 0.38	99.40 ± 0.04	93.57 ± 0.24	75.75 ± 0.68	53.12 ± 0.54	78.92
	SpanInit	67.12 ± 0.42	90.99 ± 0.20	75.53 ± 0.51	99.50 ± 0.03	93.68 ± 0.05	77.58 ± 0.54	54.80 ± 0.15	79.89
	Δ	+2.57*	+0.62	−0.12	+0.09*	+0.11	+1.83*	+1.67*	*+0.97*

Specialized

	Method	Camelyon	EuroSAT	Resisc45	Retinopathy	Mean
MV-B	Linear	82.60 ± 0.08	92.91 ± 0.17	81.84 ± 0.06	74.46 ± 0.08	82.95
	Additional-scan	83.80 ± 0.06	93.33 ± 0.20	84.38 ± 0.25	71.41 ± 0.17	83.23
	SpanInit	81.82 ± 0.84	93.99 ± 0.23	84.32 ± 0.10	70.65 ± 0.41	82.70
	Δ	−1.98*	+0.66*	−0.06	−0.76*	*−0.53*
MV-L	Linear	83.44 ± 0.06	93.30 ± 0.13	83.19 ± 0.26	74.51 ± 0.15	83.61
	Additional-scan	83.74 ± 0.16	93.93 ± 0.13	85.40 ± 0.13	69.98 ± 1.31	83.26
	SpanInit	83.15 ± 0.15	94.57 ± 0.16	85.48 ± 0.14	69.91 ± 0.61	83.28
	Δ	−0.59*	+0.65*	+0.08	−0.06	*+0.02*

Structured

	Method	Clevr-Count	Clevr-Dist	DMLab	KITTI-Dist	dSpr-Loc	dSpr-Ori	sNORB-Azim	sNORB-Elev	Mean
MV-B	Linear	46.07 ± 0.13	37.29 ± 0.28	42.96 ± 0.07	69.29 ± 0.21	16.93 ± 0.23	30.56 ± 0.05	18.13 ± 0.31	27.22 ± 0.13	36.06
	Additional-scan	76.88 ± 0.32	61.35 ± 0.33	48.86 ± 0.38	78.67 ± 0.77	67.37 ± 1.52	53.65 ± 0.37	24.43 ± 0.43	32.92 ± 0.66	55.52
	SpanInit	76.45 ± 0.28	61.86 ± 0.80	48.51 ± 0.40	80.08 ± 0.92	71.27 ± 2.03	53.64 ± 0.17	24.41 ± 0.55	32.34 ± 0.55	56.07
	Δ	−0.43	+0.52	−0.35	+1.41	+3.91*	−0.02	−0.01	−0.58	*+0.56*
MV-L	Linear	47.40 ± 0.21	38.04 ± 0.03	43.37 ± 0.08	66.20 ± 1.02	17.16 ± 0.16	32.20 ± 0.01	19.20 ± 0.07	28.10 ± 0.17	36.46
	Additional-scan	75.24 ± 1.43	59.95 ± 0.23	50.59 ± 0.43	78.39 ± 1.60	70.61 ± 0.65	55.46 ± 0.07	25.47 ± 0.28	32.89 ± 0.14	56.07
	SpanInit	74.16 ± 0.91	58.21 ± 1.57	50.22 ± 0.33	80.54 ± 0.43	76.77 ± 0.16	54.26 ± 0.26	25.07 ± 0.38	33.64 ± 0.81	56.61
	Δ	−1.08	−1.74	−0.37	+2.16	+6.16*	−1.20*	−0.40	+0.74	*+0.54*

Mean (19 tasks)	Linear	Additional-scan	SpanInit	Δ	Params
MambaVision-B	**60.07**	**69.60**	**69.99**	**+0.39**	**50.7K (0.05%)**
MambaVision-L	**61.01**	**70.21**	**70.80**	**+0.59**	**77.6K (0.03%)**

bank, already improves over copying in the nineteen-task mean (+0.30), showing that introducing distinct decay scales accounts for a substantial part of the aggregate effect. SpanInit reaches +0.39, providing an additional but modest aggregate advantage over uniform spacing. Thus, the ablation supports the importance of spreading the added decay scales and provides more limited evidence for an additional benefit from conditioning their placement on the pretrained bank.

Window and expansion size. Replacing the ceiling L with the longest pretrained timescale yields a similar nineteen-task difference (Table 4(c), +0.39 versus +0.37), although the group-wise effects differ. The aggregate result is therefore similar under these two tested ceiling choices; we retain L because it provides an architecture-defined finite horizon. The positive aggregate effect also persists when the expansion is reduced to $N' = 3$ (Table 4(d), +0.55 versus +0.39 at $N' = 6$), but is smaller and not significant at $N' = 1$ (+0.26). The group-wise changes differ, however, so the smaller expansion is not uniformly preferable.

Comparison with the original S4D initialization. Table 4(e) is designed to compare SpanInit with reapplying the original S4D initialization to the added state dimensions. This comparison tests whether the effect can be explained simply by restoring the initialization spectrum used before pretraining, rather than by placing new states relative to the geometry of the pretrained bank. Reap-

Table 4: Ablations of (a) the backbone's pretraining, (b) bank-aware placement against uniform log-spacing, (c) the ceiling of the scan window, (d) the number of added dimensions N', (e) the original S4D initialization, and (f) the model size.

Δ	IN-21k $\to$ 1k	IN-1k
Natural	$+0.73$	-1.26
Specialized	-0.53	-0.79
Structured	$+0.56$	$+0.59$
All 19	$+0.39$	-0.38

(a)

Δ	SpanInit	uniform
Natural	$+0.73$	$+0.66$
Specialized	-0.53	-0.45
Structured	$+0.56$	$+0.36$
All 19	$+0.39$	$+0.30$

(b)

Δ	$[1, L]$	$[1, \max \tau]$
Natural	$+0.73$	$+0.26$
Specialized	-0.53	-0.43
Structured	$+0.56$	$+0.85$
All 19	$+0.39$	$+0.37$

(c)

Δ	$N' = 1$	$N' = 3$	$N' = 6$
Natural	$+0.55$	$+0.57$	$+0.73$
Specialized	-0.47	-0.37	-0.53
Structured	$+0.36$	$+1.01$	$+0.56$
All 19	$+0.26$	$+0.55$	$+0.39$

(d)

Δ	SpanInit	S4D
Natural	$+0.73$	-1.37
Specialized	-0.53	-0.93
Structured	$+0.56$	-6.58
All 19	$+0.39$	-3.47

(e)

Δ	MV-B	MV-L
Natural	$+0.73$	$+0.97$
Specialized	-0.53	$+0.02$
Structured	$+0.56$	$+0.54$
All 19	$+0.39$	$+0.59$

(f)

plying that spectrum lowers the nineteen-task mean from 69.60 to 66.13 (-3.47), with decreases in all three groups. The panel therefore does not support restoring the pre-pretraining spectrum as an explanation of the effect.

Model-size transfer. Table 4(f) shows that the positive aggregate change on the ImageNet-21k-pretrained checkpoint is preserved at the larger model size: $+0.59$ for MambaVision-L compared with $+0.39$ for MambaVision-B. Supplementary Table 13 further shows a -0.21 change for the ImageNet-1k-only MambaVision-L. Thus, the aggregate sign contrast between the two checkpoint types recurs across the B and L model sizes, although the underlying task-level effects remain heterogeneous. This recurrence strengthens the empirical association between inherited checkpoint geometry and the effect of state-expansion initialization without establishing either pretraining history or bank geometry as its causal source.

7 Conclusion

We characterized the decay-timescale geometry of pretrained Mamba-based vision checkpoints, whose within-channel relative span can be read directly from A. The examined ImageNet-1k-only checkpoints have near-collapsed banks, whereas the MambaVision checkpoints exposed to ImageNet-21k have widely spread ones. Because copying a pretrained A entry reuses an existing decay scale, **SpanInit** instead initializes added states by filling uncovered log-timescale gaps within the scan-relevant range $[1, L]$, changing only their initial A values. It requires one calibration forward pass but adds no trainable parameters or inference cost. On VTAB-1k, the aggregate effect of SpanInit changes sign between the two checkpoint types, and the same pattern recurs for MambaVision-B and MambaVision-L. These results motivate checkpoint-aware state-expansion initialization rather than a single fixed rule.

Limitations. SpanInit does not improve all tasks, and the source of this task-level heterogeneity remains unclear. The timescale τ captures recurrent decay geometry, not full functional influence: the selective B and C terms can reorder dimensions, particularly when pretrained decay scales are close (Appendix B.3). The window $[1, L]$ and the greedy gap-filling rule are design choices, not claimed to be globally optimal. Uniform log-spacing already accounts for much of the aggregate gain ($+0.30$ versus $+0.39$), so the evidence for bank-aware placement specifically is modest. Finally, the comparisons cover MambaVision-B and MambaVision-L checkpoints whose pretraining data and recipes differ, so the checkpoint dependence cannot be attributed causally to pretraining data or bank geometry alone.

AI USE STATEMENT

In this work, we used generative AI tools to implement methods, to assist with translation, and to compute and interpret statistical results. We have not used generative AI tools to propose or refine hypotheses, or to design or give feedback on the research methodology and experiments. Generating synthetic datasets, developing theoretical models or conceptual frameworks, formulating mathematical claims, providing ingredients for proofs, assisting in the writing of proofs, cleaning or reformatting datasets, and supporting qualitative and thematic data analysis are not applicable to this work. Additionally, we used generative AI tools to create and modify the figures, to create and edit software code, to summarize existing literature and identify relevant work, to search for information, to edit the text for readability, and to format references. We have reviewed all AI-assisted work. AI-assisted implementations were read and tested by the authors and checked against the method descriptions in this paper before any experiment was run. AI-assisted text was checked by the authors for accuracy and originality, and statements about prior work were verified against the cited papers. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

Section 5 and Algorithm 1 specify SpanInit, and Section 4.1 and Appendix A.1 specify how the step $\bar{\Delta}$ that it needs is measured. Every fine-tuning setting is listed in Table 6 and follows the VTAB-1k configuration released by Yoshimura et al. (2025); the pretrained checkpoints are the public releases listed in Table 1, the data splits are the released VTAB-1k splits summarized in Table 5, and all runs use seeds 42, 43 and 44. Appendix A.2 states the error model behind every interval and test. The selective-scan backward kernel accumulates gradients with atomic additions, so a repeated run is not bit-identical to the reported one; this run-to-run variation is contained in the seed-to-seed spread that the tables report.

REFERENCES

Ameen Ali Ali, Itamar Zimerman, and Lior Wolf. The hidden attention of Mamba models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1516–1534, 2025.

Seyedarmin Azizi, Souvik Kundu, Mohammad Sadeghi, and Massoud Pedram. MambaExtend: A training-free approach to improve long context extension of Mamba. In *International Conference on Learning Representations*, volume 2025, pp. 97075–97090, 2025.

Assaf Ben-Kish, Itamar Zimerman, Shady Abu-Hussein, Nadav Cohen, Amir Globerson, Lior Wolf, and Raja Giryes. DeciMamba: Exploring the length extrapolation potential of Mamba. In *International Conference on Learning Representations*, volume 2025, pp. 101148–101170, 2025.

Shoufa Chen, Chongjian Ge, Zhan Tong, Jiangliu Wang, Yibing Song, Jue Wang, and Ping Luo. AdaptFormer: Adapting vision transformers for scalable visual recognition. *Advances in Neural Information Processing Systems*, 35:16664–16678, 2022.

Tri Dao and Albert Gu. Transformers are SSMs: Generalized models and efficient algorithms through structured state space duality. In *International Conference on Machine Learning*, 2024.

Kevin Galim, Wonjun Kang, Yuchen Zeng, Hyung Il Koo, and Kangwook Lee. Parameter-efficient fine-tuning of state space models. In *International Conference on Machine Learning*, 2025.

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

Albert Gu, Tri Dao, Stefano Ermon, Atri Rudra, and Christopher Ré. HiPPO: Recurrent memory with optimal polynomial projections. *Advances in Neural Information Processing Systems*, 33: 1474–1487, 2020.

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.

Albert Gu, Karan Goel, Ankit Gupta, and Christopher Ré. On the parameterization and initialization of diagonal state space models. *Advances in Neural Information Processing Systems*, 35:35971–35983, 2022a.

Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. In *International Conference on Learning Representations*, 2022b.

Albert Gu, Isys Johnson, Aman Timalsina, Atri Rudra, and Christopher Ré. How to train your HiPPO: State space models with generalized orthogonal basis projections. In *International Conference on Learning Representations*, 2023.

Ankit Gupta, Albert Gu, and Jonathan Berant. Diagonal state spaces are as effective as structured state spaces. *Advances in Neural Information Processing Systems*, 35:22982–22994, 2022.

Seokil Ham, Hee-Seon Kim, Sangmin Woo, and Changick Kim. Parameter efficient mamba tuning via projector-targeted diagonal-centric linear transformation. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 30106–30115. IEEE, 2025.

Zeyu Han, Chao Gao, Jinyang Liu, Jeff Zhang, and Sai Qian Zhang. Parameter-efficient fine-tuning for large models: A comprehensive survey. *arXiv preprint arXiv:2403.14608*, 2024.

Ali Hatamizadeh and Jan Kautz. MambaVision: A hybrid Mamba-transformer vision backbone. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 25261–25270. IEEE, 2025.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for NLP. In *International Conference on Machine Learning*, pp. 2790–2799. PMLR, 2019.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *European Conference on Computer Vision*, pp. 709–727. Springer, 2022.

Shibo Jie and Zhi-Hong Deng. FacT: Factor-tuning for lightweight adaptation on vision transformer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 1060–1068, 2023.

Wonjun Kang, Kevin Galim, Yuchen Zeng, Minjae Lee, Hyung Il Koo, and Nam Ik Cho. State-offset tuning: State-based parameter-efficient fine-tuning for state space models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 474–487, 2025.

Donghyun Lee, Yuhang Li, Ruokai Yin, Shiting Xiao, and Priyadarshini Panda. Memba: Membrane-driven parameter-efficient fine-tuning for Mamba. In *International Conference on Learning Representations*, volume 2026, pp. 97118–97141, 2026.

Vladislav Lialin, Vijeta Deshpande, Xiaowei Yao, and Anna Rumshisky. Scaling down to scale up: A guide to parameter-efficient fine-tuning. *arXiv preprint arXiv:2303.15647*, 2023.

Dongze Lian, Daquan Zhou, Jiashi Feng, and Xinchao Wang. Scaling & shifting your features: A new baseline for efficient model tuning. *Advances in Neural Information Processing Systems*, 35: 109–123, 2022.

Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. DoRA: Weight-decomposed low-rank adaptation. In *International Conference on Machine Learning*, 2024a.

Yue Liu, Yunjie Tian, Yuzhong Zhao, Hongtian Yu, Lingxi Xie, Yaowei Wang, Qixiang Ye, Jianbin Jiao, and Yunfan Liu. VMamba: Visual state space model. *Advances in Neural Information Processing Systems*, 37:103031–103063, 2024b.

Peng Lu, Jerry Huang, Qiuhao Zeng, Xinyu Wang, Boxing Chen, Philippe Langlais, and Yufei Cui. Mamba modulation: On the length generalization of Mamba models. *Advances in Neural Information Processing Systems*, 38:19414–19459, 2025.

Gen Luo, Minglang Huang, Yiyi Zhou, Xiaoshuai Sun, Guannan Jiang, Zhiyu Wang, and Rongrong Ji. Towards efficient visual adaption via structural re-parameterization. *arXiv preprint arXiv:2302.08106*, 2023.

Fanxu Meng, Zhaohui Wang, and Muhan Zhang. PiSSA: Principal singular values and singular vectors adaptation of large language models. *Advances in Neural Information Processing Systems*, 37:121038–121072, 2024.

Ruben Solozabal, Velibor Bojkovic, Hilal AlQuabeh, Kentaro Inui, and Martin Takác. Uncovering the spectral bias in diagonal state space models. *Advances in Neural Information Processing Systems*, 38:38687–38713, 2025.

Asher Trockman, Hrayr Harutyunyan, J Zico Kolter, Sanjiv Kumar, and Srinadh Bhojanapalli. Mimetic initialization helps state space models learn to recall. *arXiv preprint arXiv:2410.11135*, 2024.

Zhifan Ye, Kejing Xia, Yonggan Fu, Xin Dong, Jihoon Hong, Xiangchi Yuan, Shizhe Diao, Jan Kautz, Pavlo Molchanov, and Yingyan Celine Lin. LongMamba: Enhancing Mamba's long-context capabilities via training-free receptive field enlargement. In *International Conference on Learning Representations*, volume 2025, pp. 68486–68504, 2025.

Masakazu Yoshimura, Teruaki Hayashi, and Yota Maeda. MambaPEFT: Exploring parameter-efficient fine-tuning for Mamba. In *International Conference on Learning Representations*, volume 2025, pp. 94093–94117, 2025.

Annan Yu, Arnur Nigmetov, Dmitriy Morozov, Michael W Mahoney, and N Benjamin Erichson. Robustifying state-space models for long sequences via approximate diagonalization. In *International Conference on Learning Representations*, volume 2024, pp. 33273–33305, 2024.

Elad Ben Zaken, Yoav Goldberg, and Shauli Ravfogel. BitFit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 1–9, 2022.

Xiaohua Zhai, Joan Puigcerver, Alexander Kolesnikov, Pierre Ruyssen, Carlos Riquelme, Mario Lucic, Josip Djolonga, Andre Susano Pinto, Maxim Neumann, Alexey Dosovitskiy, et al. A large-scale study of representation learning with the visual task adaptation benchmark. *arXiv preprint arXiv:1910.04867*, 2019.

Lianghui Zhu, Bencheng Liao, Qian Zhang, Xinlong Wang, Wenyu Liu, and Xinggang Wang. Vision Mamba: Efficient visual representation learning with bidirectional state space model. In *International Conference on Machine Learning*, 2024.

A EXPERIMENTAL DETAILS

A.1 IMPLEMENTATION DETAILS

Dataset and metric. We evaluate on VTAB-1k (Zhai et al., 2019), comprising nineteen classification tasks grouped into Natural, Specialized, and Structured categories. Each task provides 1,000 training images, all of which we use, and is evaluated on its full test split. Table 5 lists the tasks. We report per-task test accuracy together with group means and the nineteen-task mean. All numbers are averaged over seeds 42, 43, and 44; per-task entries also give the sample standard deviation.

Table 5: The nineteen VTAB-1k tasks, with their group (Natural, Specialized, Structured), number of classes, and test-set size.

Natural							
	CIFAR-100	Caltech101	DTD	Flowers102	Pets	SVHN	Sun397
Classes	100	102	47	102	37	10	397
Train	1,000	1,000	1,000	1,000	1,000	1,000	1,000
Test	10,000	6,084	1,880	6,149	3,669	26,032	21,750

Specialized				
	Camelyon	EuroSAT	Resisc45	Retinopathy
Classes	2	10	45	5
Train	1,000	1,000	1,000	1,000
Test	32,768	5,400	6,300	42,670

Structured								
	Clevr-Count	Clevr-Dist	DMLab	KITTI-Dist	dSpr-Loc	dSpr-Ori	sNORB-Azim	sNORB-Elev
Classes	8	6	6	4	16	16	18	9
Train	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000
Test	15,000	15,000	22,735	711	73,728	73,728	12,150	12,150

Backbone and adaptation. Our main experiments use MambaVision-B (Hatamizadeh & Kautz, 2025), pretrained on ImageNet-21k and subsequently fine-tuned on ImageNet-1k, with the pretrained backbone frozen. Additional-scan expands each of its eight SSM blocks with $N' = 6$ state dimensions; only the added states and the classification head are trained. SpanInit uses the same state-expansion structure and differs only in the initialization of the added A values (Section 5). Ablations with other values of N' are reported separately.

Training and calibration. We follow the VTAB-1k recipe released with MambaPEFT (Yoshimura et al., 2025); Table 6 lists the full settings. The same protocol is used for every task, method, and seed, with no task-specific hyperparameter tuning or model selection; each run is evaluated at the final epoch. For SpanInit, the channel-wise mean step $\bar{\Delta}$ is estimated once from the 1,000 training images with a single forward pass through the frozen backbone and is used only to initialize the added states. It is not re-estimated during fine-tuning, and the rows of x_{proj} that produce Δ receive no gradient. Each run uses a single NVIDIA GeForce RTX 3060 GPU and takes less than two hours; SpanInit otherwise has the same training and inference computation as Additional-scan, with the one-time calibration overhead reported in Section A.3.

Table 6: Training and evaluation settings shared by all VTAB-1k tasks, methods, and seeds. We follow the official MambaPEFT protocol (Yoshimura et al., 2025) without task-specific hyperparameter tuning.

Setting	Value
Optimizer	AdamW, $\beta = (0.9, 0.999)$, $\epsilon = 10^{-8}$
Learning rate	10^{-3}
Weight decay	10^{-4}, none on norms, biases, $A_{\log}$ or the skip parameter
Schedule	ten epochs of linear warm-up, then cosine decay to 10^{-5}
Epochs	100
Batch size	32 to train, 64 to evaluate
Gradient clipping	global norm 5.0
Input	bicubic resize to 224×224, no augmentation

A.2 Seed pairing and statistical testing

All method comparisons are seed-paired. For task t and seed s, we define

$$\delta_{t,s} = \mathrm{Acc}_{t,s}^{\mathrm{SpanInit}} - \mathrm{Acc}_{t,s}^{\mathrm{Add.\text{-}scan}}. \tag{13}$$

The paired runs share the seed, data order, and initialization of the added B and C parameters, so their difference isolates the change in the initial A. Every Δ in the tables is the mean of $\delta_{t,s}$ over the three seeds, and all differences and means are computed before rounding.

For individual tasks, the tables mark significance with the unpaired one-tailed Welch's t-test of Jia et al. (2022), applied to the three runs of each arm; unlike theirs, which fixes the direction in advance, we set the tail by the sign of the observed difference, so a task can be marked in either direction. Because the runs are paired, the three differences $\{\delta_{t,s}\}_{s=1}^{3}$ can also be tested against zero with a two-sided one-sample t-test on two degrees of freedom. Under it five tasks differ from zero rather than the seven marked in the tables: four gains between $+1.10$ and $+3.91$, and a loss of -0.35 on DMLab, where the three seeds agree closely enough to separate a small effect. With only three seeds these per-task tests have little power, so we treat them as supplementary evidence; Table 7 gives both tests, with intervals, for all nineteen tasks, and no task is selected or excluded on the basis of them.

For benchmark-level comparisons we treat the nineteen VTAB-1k tasks as fixed and the seeds as the only source of noise: the mean of $\delta_{t,s}$ over tasks and seeds is tested with the seed variance of each task, pooled over tasks, on 38 degrees of freedom. On the ImageNet-21k checkpoint the mean is $+0.39$ (95% CI $[+0.20, +0.59]$, $p < 0.001$); on the checkpoint pretrained on ImageNet-1k alone it is -0.38 ($[-0.65, -0.11]$, $p = 0.007$); and the contrast between the two checkpoints, formed from the paired seeds, is $+0.77$ ($[+0.47, +1.08]$, $p < 0.001$) (Table 4a, with the per-task values in Table 12). Group means in Table 4 are means of the same $\delta_{t,s}$ over the tasks of a group.

As a sensitivity check we also keep the seed as the unit of replication, since the same three seeds are shared across all tasks. Averaging the paired differences over the nineteen tasks within each seed,

$$\bar{\delta}_s = \frac{1}{19} \sum_{t=1}^{19} \delta_{t,s}, \tag{14}$$

and testing the three values with a two-sided one-sample t-test leaves two degrees of freedom, so the width of the interval depends on the three seeds that were drawn. The $\bar{\delta}_s$ are $+0.35$, $+0.37$ and $+0.46$ on the ImageNet-21k checkpoint and -0.64, -0.33 and -0.18 on the ImageNet-1k one; their agreement in sign is the more robust summary. Under this aggregate the two means are $+0.39$ ($[+0.24, +0.54]$) and -0.38 ($[-0.97, +0.20]$), and their difference is $+0.77$ ($[+0.31, +1.24]$). On the ImageNet-21k checkpoint the three $\bar{\delta}_s$ lie about three times closer together than the per-task seed variances predict, and on the ImageNet-1k one they match it. These benchmark-level tests concern the predefined VTAB-1k task set; they do not imply that the same mean effect will hold on unseen downstream tasks.

Table 7: Per-task significance on the ImageNet-21k MambaVision-B under both tests. Δ is the seed-paired difference of SpanInit from Additional-scan, with its 95% confidence interval over the three seeds. The paired t column is a two-sided one-sample t-test on the three differences; the Welch t column is the unpaired one-tailed test of Jia et al. (2022) that sets the stars in the other tables. Values below 0.05 are in bold.

Task	Δ	95% CI	p	
			paired t	Welch t
Natural				
CIFAR-100	+1.10	[+0.71, +1.50]	**0.007**	**0.028**
Caltech101	+1.59	[+0.71, +2.48]	**0.016**	**0.005**
DTD	−0.43	[−1.82, +0.97]	0.319	0.099
Oxford-Flowers102	−0.04	[−0.32, +0.25]	0.626	0.252
Oxford-Pets	+0.12	[−0.82, +1.05]	0.641	0.264
SVHN	+2.64	[+0.70, +4.59]	**0.028**	**0.003**
Sun397	+0.12	[−1.10, +1.33]	0.713	0.315
Specialized				
Patch-Camelyon	−1.98	[−4.04, +0.08]	0.054	**0.027**
EuroSAT	+0.66	[−0.30, +1.63]	0.099	**0.011**
Resisc45	−0.06	[−0.87, +0.75]	0.787	0.367
Retinopathy	−0.76	[−1.61, +0.09]	0.061	**0.034**
Structured				
Clevr-Count	−0.43	[−1.84, +0.97]	0.318	0.078
Clevr-Dist	+0.52	[−1.46, +2.49]	0.376	0.193
DMLab	−0.35	[−0.42, −0.29]	**0.002**	0.163
KITTI-Dist	+1.41	[−2.79, +5.60]	0.286	0.057
dSprites-Loc	+3.91	[+0.16, +7.65]	**0.046**	**0.030**
dSprites-Ori	−0.02	[−0.73, +0.69]	0.917	0.470
sNORB-Azim	−0.01	[−0.37, +0.35]	0.909	0.490
sNORB-Elev	−0.58	[−3.48, +2.32]	0.483	0.156

A.3 Computational cost

Once initialized, SpanInit has the same trainable parameters, fine-tuning computation, and inference cost as Additional-scan. Its only additional computation is the one-time estimation of $\bar{\Delta}$ before fine-tuning, obtained with a forward-only pass over the task's training set. The subsequent placement procedure uses only the measured step and pretrained A and is negligible relative to fine-tuning. Table 8 reports the measured overhead.

Table 8: Computational overhead of SpanInit relative to Additional-scan, measured from the MambaVision-B runs of Table 3 on a single NVIDIA GeForce RTX 3060. The calibration time includes the data loading before it and is therefore an upper bound. † Parameter counts exclude the classification head.

	Add.-scan	SpanInit
Trainable parameters†	50,688	50,688
Estimating $\bar{\Delta}$ before fine-tuning	–	3–10 s
as a share of the run	–	0.06–0.51%
Added at inference	none	none

A.4 SpanInit pseudocode

Algorithm 1 summarizes SpanInit for one SSM block. It is applied once before fine-tuning and changes only the initial A values of the added states. In paired experiments, SpanInit and Additional-

scan use identical initializations of the added B and C parameters, so the two runs differ only in the initial A. We sort the selected timescales before assigning them to the added state slots, providing a deterministic assignment independent of the order in which the gaps are filled.

Algorithm 1 SpanInit for one SSM block

Input: pretrained $|A| \in \mathbb{R}^{D\times N}$, measured $\bar{\Delta} \in \mathbb{R}^{D}$, scan length L, added dimensions N'
Output: initial $A_{\log}^{\text{new}} \in \mathbb{R}^{D\times N'}$
for each channel $d = 1, \dots, D$ **do**
  $\tau_{d,n} \leftarrow 1/(\bar{\Delta}_d |A_{d,n}|)$ for $n = 0, \dots, N-1$ ▷ Equation 5
  $\mathcal{S} \leftarrow \{1, L\} \cup \{\tau_{d,n} : 1 \le \tau_{d,n} \le L\}$ ▷ Equation 8
  for $j = 1, \dots, N'$ **do**
    sort $\mathcal{S}$ as $s_1 < \cdots < s_M$
    $i^\star \leftarrow \arg\max_{1\le i<M} \log(s_{i+1}/s_i)$ ▷ Equation 9
    $\tau_{d,j}^{\text{new}} \leftarrow \sqrt{s_{i^\star} s_{i^\star+1}}$
    $\mathcal{S} \leftarrow \mathcal{S} \cup \{\tau_{d,j}^{\text{new}}\}$ ▷ Equation 10
  end for
  sort $\tau_{d,1}^{\text{new}}, \dots, \tau_{d,N'}^{\text{new}}$ increasingly
  $(A_{\log}^{\text{new}})_{d,j} \leftarrow -\log\left(\bar{\Delta}_d \tau_{d,j}^{\text{new}}\right)$ for $j = 1, \dots, N'$ ▷ Equation 12
end for

B Interpretation and Validation of the Timescale Diagnosis

B.1 Interpreting timescales in image scans

Section 4.1 measures timescales in scan tokens rather than in image-space distance. Figure 3 makes this unit concrete by placing four timescales on the 14×14 feature map scanned by the five 256-channel MambaVision-B blocks, for which $L = 196$. Under the mean-step approximation of Equation 5, a state with timescale τ retains a token from τ steps earlier with a recurrent decay factor of $1/e$, so the reach follows the scan order rather than the spatial layout of the image.

The four examples are a short and a long representative pretrained timescale together with the two bounds of SpanInit's scan window, $\tau = 1$ and $\tau = L = 196$. They visualize the decay geometry that τ encodes; τ is a proxy for recurrent reach, not a measure of complete functional influence, which also depends on the selective B and C terms.

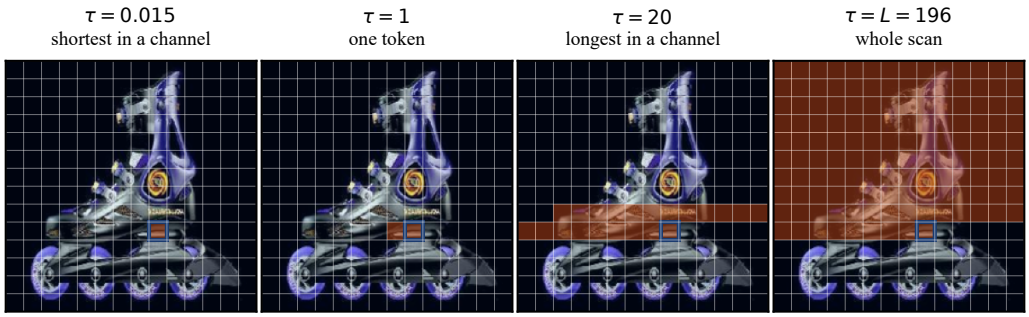


Figure 3: Illustration of four decay timescales on the 14×14 feature map scanned by the five 256-channel MambaVision-B blocks. The blue square marks the current token, and shading follows the scan order. Under the mean-step exponential decay model, a token τ steps in the past has a recurrent decay factor of $1/e$. Thus, τ measures distance in scan tokens rather than a spatial radius in the image.

B.2 Absolute bank placement across tasks

The within-channel span of Section 4 is independent of $\bar{\Delta}$, whereas the bank's absolute position on the τ axis depends on the downstream input through $\bar{\Delta}$. Table 9 reports, per VTAB-1k task, the fraction of pretrained state dimensions below, inside, and above the scan-relevant interval $[1, L]$ in the five 256-channel blocks. On average, 38.6% of the ImageNet-21k-pretrained dimensions fall inside $[1, L]$ against 74.8% for the ImageNet-1k-only checkpoint; 58.1% versus 22.6% lie below one token, and only 3.3% versus 2.6% exceed L. The fractions inside $[1, L]$ move by under two percentage points across the nineteen tasks (37.6–39.2% and 73.8–75.5%), so the qualitative gap between the two checkpoints does not come from a particular downstream dataset.

Table 9: Task-wise placement of pretrained timescales in the five 256-channel MambaVision-B blocks with $L = 196$. Entries are the fractions of state dimensions below, inside, and above the scan-relevant interval $[1, L]$ for the ImageNet-21k-pretrained and ImageNet-1k-only checkpoints.

	$\tau < 1$		$1 \le \tau \le L$		$\tau > L$	
Task	21k	1k	21k	1k	21k	1k
Natural						
CIFAR-100	57.6	22.9	39.0	74.5	3.5	2.6
Caltech101	58.9	23.1	38.7	74.5	2.4	2.4
DTD	57.8	23.3	38.6	74.0	3.6	2.7
Oxford-Flowers102	59.0	23.4	38.7	74.1	2.3	2.5
Oxford-Pets	59.3	23.8	38.7	73.8	2.0	2.4
SVHN	56.7	23.1	38.7	74.1	4.6	2.8
Sun397	59.4	23.6	38.5	74.0	2.1	2.4
Natural mean	58.4	23.3	38.7	74.1	2.9	2.5
Specialized						
Patch-Camelyon	57.7	22.5	38.9	74.9	3.4	2.6
EuroSAT	57.0	22.2	39.2	75.2	3.8	2.6
Resisc45	58.3	23.0	39.0	74.4	2.7	2.6
Retinopathy	58.4	22.5	38.9	75.3	2.7	2.2
Specialized mean	57.9	22.5	39.0	74.9	3.1	2.5
Structured						
Clevr-Count	58.6	22.2	38.5	75.4	2.9	2.3
Clevr-Dist	58.6	22.2	38.5	75.4	2.9	2.3
DMLab	58.8	22.5	38.6	75.2	2.7	2.4
KITTI-Dist	59.5	23.8	38.5	73.8	2.1	2.4
dSprites-Loc	56.5	21.0	37.6	75.4	5.9	3.6
dSprites-Ori	56.5	21.0	37.6	75.4	5.9	3.6
sNORB-Azim	57.4	21.9	38.6	75.5	4.0	2.6
sNORB-Elev	57.4	21.9	38.6	75.5	4.0	2.6
Structured mean	57.9	22.1	38.3	75.2	3.8	2.7
Mean (19 tasks)	58.1	22.6	38.6	74.8	3.3	2.6

Table 10 takes a channel-level view through the longest pretrained timescale in each channel. For the ImageNet-21k-pretrained checkpoint its median is 22.4 tokens (interquartile range 2.6–127.9) and it exceeds L in 15.8% of channels; for the ImageNet-1k-only checkpoint the median is 8.8 (interquartile range 1.5–46.2) and only 3.3% of channels reach beyond L. The ImageNet-21k-pretrained distribution is thus far broader and extends past the scan, whereas the ImageNet-1k-only one stays concentrated at short scales; the separation is stable across the nineteen tasks.

This also motivates anchoring SpanInit at L. Even for the broader checkpoint the median longest timescale (22.4 tokens, against 8.8 for the collapsed one) sits well below the horizon $L = 196$, leaving the upper part of $[1, L]$ uncovered by the pretrained bank; including both boundaries lets the gap-filling rule reach that region.

Table 10: Distribution across channels of the longest pretrained timescale in the five 256-channel MambaVision-B blocks with $L = 196$. Quartiles, median, and geometric mean summarize its location across channels; the final columns report the fraction of channels whose longest timescale exceeds the scan length.

	Q_1		median		Q_3		mean		$\tau > L$ (%)	
Task	21k	1k	21k	1k	21k	1k	21k	1k	21k	1k
Natural										
CIFAR-100	2.82	1.46	21.3	8.5	125.0	45.2	20.9	9.1	16.4	3.4
Caltech101	2.43	1.49	20.0	8.3	104.3	40.4	18.0	8.7	12.4	2.8
DTD	2.60	1.45	22.3	8.7	138.9	48.9	20.8	9.4	17.0	3.6
Oxford-Flowers102	2.40	1.44	21.1	8.3	104.5	40.3	18.1	8.6	11.6	2.9
Oxford-Pets	2.36	1.43	20.0	8.1	96.3	38.0	17.1	8.3	10.9	2.6
SVHN	2.93	1.48	26.1	8.9	155.9	51.6	23.6	9.7	20.4	3.8
Sun397	2.30	1.48	18.3	7.8	99.3	37.1	17.0	8.2	11.2	2.7
Natural mean	2.55	1.46	21.3	8.4	117.7	43.1	19.3	8.9	14.3	3.1
Specialized										
Patch-Camelyon	2.64	1.50	22.0	9.0	123.6	49.5	20.3	9.6	15.7	3.5
EuroSAT	2.81	1.55	23.7	8.8	137.1	47.9	22.1	9.6	18.2	3.5
Resisc45	2.50	1.48	20.0	8.2	113.4	41.1	18.9	8.8	13.1	3.4
Retinopathy	2.50	1.55	22.3	8.4	115.4	43.1	19.5	9.0	13.4	2.7
Specialized mean	2.61	1.52	22.0	8.6	122.4	45.4	20.2	9.2	15.1	3.3
Structured										
Clevr-Count	2.53	1.55	22.0	8.9	121.9	42.8	19.5	9.1	15.0	3.0
Clevr-Dist	2.53	1.55	22.0	8.9	121.9	42.8	19.5	9.1	15.0	3.0
DMLab	2.50	1.52	19.7	8.0	108.5	40.6	18.5	8.7	13.1	2.9
KITTI-Dist	2.32	1.38	19.4	8.0	95.2	36.7	16.9	8.1	10.5	2.8
dSprites-Loc	2.99	1.62	27.6	11.0	191.1	65.3	25.6	11.3	24.5	4.9
dSprites-Ori	2.99	1.62	27.6	11.0	191.1	65.3	25.6	11.3	24.5	4.9
sNORB-Azim	2.78	1.56	25.0	9.1	143.4	50.5	22.0	9.8	18.3	3.4
sNORB-Elev	2.78	1.56	25.0	9.1	143.4	50.5	22.0	9.8	18.3	3.4
Structured mean	2.68	1.54	23.6	9.2	139.6	49.3	21.2	9.7	17.4	3.5
Mean (19 tasks)	2.62	1.51	22.4	8.8	127.9	46.2	20.3	9.3	15.8	3.3

B.3 Validation against measured contribution horizons

Equation 5 derives τ from A and the channel-wise mean step $\bar{\Delta}$, omitting the selective B and C terms and the token-wise variation in Δ_t. We therefore compare it with a contribution-based temporal horizon measured directly from the pretrained model. For state dimension n of channel d, the magnitude of the contribution from a token at lag k is

$$W_{d,n}(k) = \mathbb{E}_t \left[|C_{t,n}\, \Delta_{t-k,d}\, B_{t-k,n}| \exp\left(-|A_{d,n}| \sum_{u=t-k+1}^{t} \Delta_{u,d} \right) \right], \tag{15}$$

combining the magnitude written at the earlier token, that read at the current token, and the recurrent decay between them. Taking magnitudes removes sign cancellation, so $W_{d,n}$ measures contribution magnitude rather than net signed effect. After normalizing $W_{d,n}(k)$ over the available lags, we define the measured horizon as the lag at which cumulative contribution reaches $1 - 1/e$; for an untruncated pure exponential this recovers the e-folding scale up to discretization. We estimate it from 64 images per task, and on Caltech101 increasing the sample from 16 to 128 images leaves the median horizon-to-τ ratio essentially unchanged.

Figure 4 compares the measured horizon with the A-derived τ for states inside $[1, L]$. The two stay close over most of the scan-relevant range—at the median the measured horizon is $1.07\,\tau$ on the ImageNet-21k-pretrained checkpoint and $1.11\,\tau$ on the ImageNet-1k-only one—supporting τ as a compact proxy for recurrent reach while exposing the deviations introduced by the selective terms.

Because SpanInit fills gaps between neighboring timescales, relative ordering matters more than exact numerical agreement. Table 11 reports the fraction of within-channel state pairs ordered con-

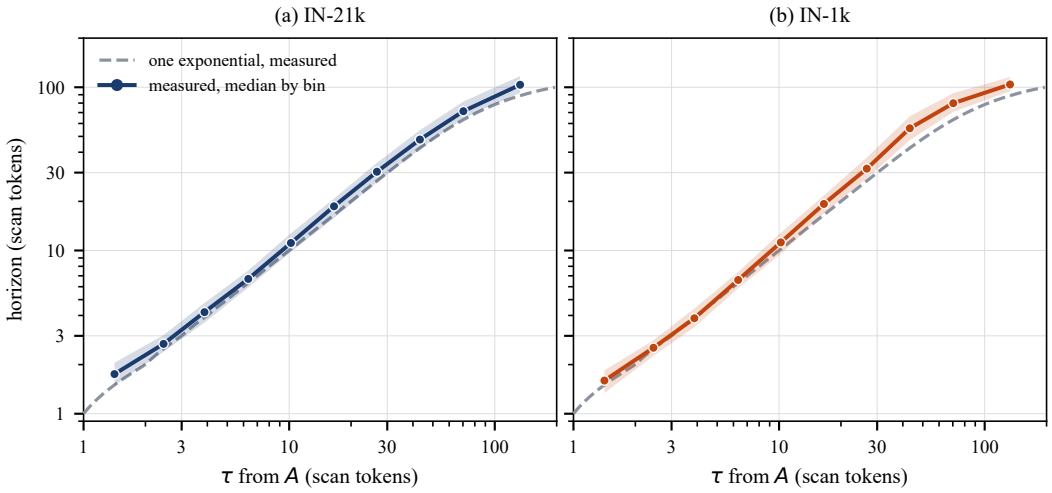


Figure 4: Measured contribution horizon versus the A-derived timescale τ for state dimensions inside $[1, L]$ in the five 256-channel MambaVision-B blocks ($L = 196$), evaluated on 64 Caltech101 images. The dashed curve shows the same measurement applied to a pure exponential; solid curves and bands show the median and interquartile range within bins of τ.

sistently by τ and by the measured horizon; since the token step is shared across the eight dimensions of a channel, any pair the two order differently is reversed by B and C. Agreement stays at or above 95.5% for the MambaVision-B and MambaVision-L checkpoints exposed to ImageNet-21k at both scan lengths, but falls for the ImageNet-1k-only checkpoints—to 89.7% (MambaVision-B) and 85.1% (MambaVision-L) at $L = 196$, and further to 73.9% and 73.2% at $L = 49$. This is consistent with their collapsed banks: when decay scales within a channel are close, small differences in the selective B and C terms can more easily reverse the functional ordering. The validation concerns pretrained states; the added states receive their own B and C parameters and are then trained.

B.4 Consistency across scan lengths

The main analysis uses the five shallower MambaVision-B blocks with $L = 196$; Figure 5 shows the two checkpoints' banks there, with SpanInit's six added dimensions. Figure 6 repeats it on the three deeper 512-channel blocks with $L = 49$, where the same separation persists: the ImageNet-21k-pretrained bank stays widely spread within channels and the ImageNet-1k-only bank stays near-collapsed, matching the lower ordering agreement of these checkpoints at $L = 49$ (73.9% for MambaVision-B and 73.2% for MambaVision-L; Table 11).

The shorter scan lowers the horizon, so a larger fraction of pretrained timescales falls beyond L. This reinforces why raw within-channel span is not itself scan coverage: it also counts decay scales outside $[1, L]$, and restricting the comparison to the scan-relevant window gives a more meaningful view across the two hierarchy levels. Since SpanInit is applied to all SSM blocks, this check confirms that the diagnostic pattern is not specific to the $L = 196$ level; it is not intended as a per-level ablation.

C Per-Task VTAB-1k Results

C.1 Per-task backbone comparison

Table 12 expands Table 4(a) to all nineteen tasks for the two MambaVision-B checkpoints. On the ImageNet-21k-pretrained checkpoint SpanInit raises the nineteen-task mean from 69.60 to 69.99 (+0.39), with group means of +0.73 (Natural), −0.53 (Specialized), and +0.56 (Structured); the largest task gains are on CIFAR-100 (+1.10), Caltech101 (+1.59), SVHN (+2.64), and dSprites-Loc (+3.91), and the largest loss is on Patch-Camelyon (−1.98). On the ImageNet-1k-only check-

Table 11: Agreement between the within-channel ordering induced by the A-derived timescale τ and by the measured contribution horizon. Results cover all nineteen VTAB-1k tasks, MambaVision-B and MambaVision-L, their ImageNet-21k-pretrained and ImageNet-1k-only checkpoints, and both scan lengths. Only state dimensions with $\tau \in [1, L]$ are included.

	$L = 196$ (%)				$L = 49$ (%)			
Task	B 21k	B 1k	L 21k	L 1k	B 21k	B 1k	L 21k	L 1k
Natural								
CIFAR-100	96.3	90.7	96.7	86.7	96.9	76.9	96.6	76.0
Caltech101	96.3	91.8	96.6	88.6	95.5	75.2	95.3	73.7
DTD	96.4	91.9	95.6	86.2	97.3	75.0	96.7	76.9
Oxford-Flowers102	96.5	92.3	96.7	88.3	96.5	73.4	95.9	71.4
Oxford-Pets	96.3	91.4	96.2	88.5	96.2	74.5	95.8	74.2
SVHN	96.0	91.0	96.8	85.4	97.2	75.1	96.7	76.7
Sun397	96.3	91.7	96.4	88.4	96.1	75.3	95.7	75.5
Natural mean	96.3	91.5	96.4	87.4	96.5	75.1	96.1	74.9
Specialized								
Patch-Camelyon	96.7	91.6	96.8	86.5	97.1	75.5	97.3	77.6
EuroSAT	96.4	91.0	96.6	86.0	96.8	73.9	96.6	75.1
Resisc45	96.4	92.4	96.5	87.6	97.2	74.6	96.5	77.5
Retinopathy	95.8	91.3	96.0	85.5	96.7	73.4	96.2	73.2
Specialized mean	96.3	91.6	96.5	86.4	97.0	74.3	96.7	75.8
Structured								
Clevr-Count	94.3	87.2	95.1	82.3	94.6	73.3	93.8	71.3
Clevr-Dist	94.3	87.2	95.1	82.3	94.6	73.3	93.8	71.3
DMLab	95.4	88.1	96.2	85.2	95.2	76.6	95.2	75.2
KITTI-Dist	96.1	92.3	96.4	88.9	95.4	72.8	95.3	73.5
dSprites-Loc	93.2	83.8	94.1	79.6	94.9	70.3	94.9	66.0
dSprites-Ori	93.2	83.8	94.1	79.6	94.9	70.3	94.9	66.0
sNORB-Azim	94.2	87.1	95.0	81.0	94.0	72.8	95.6	70.0
sNORB-Elev	94.2	87.1	95.0	81.0	94.0	72.8	95.6	70.0
Structured mean	94.4	87.1	95.1	82.5	94.7	72.8	94.9	70.4
Mean (19 tasks)	95.5	89.7	95.9	85.1	95.8	73.9	95.7	73.2

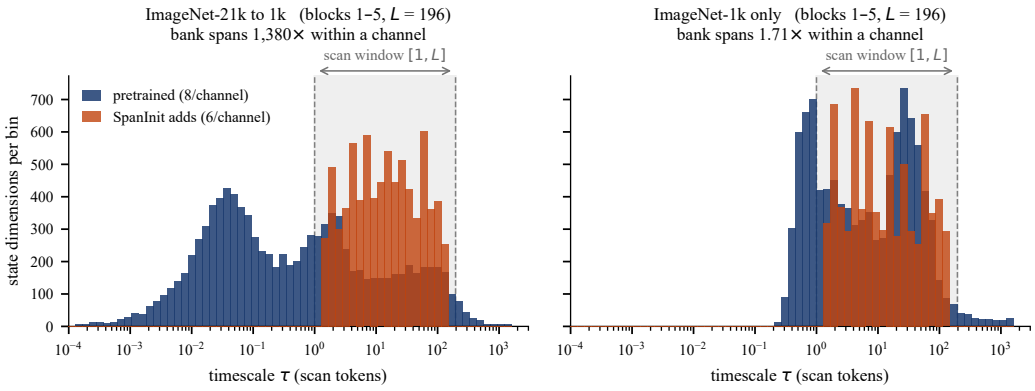


Figure 5: Pretrained timescale banks of two MambaVision-B checkpoints with SpanInit's six added dimensions on Caltech101. The checkpoints share architecture and fine-tuning settings but differ in pretraining data and schedule. Absolute placement varies with $\bar{\Delta}$ across tasks, whereas the within-channel span is determined by A alone.

point the mean instead falls from 68.04 to 67.66 (-0.38), with negative Natural and Specialized group means (-1.26 and -0.79) against a positive Structured mean ($+0.59$). dSprites-Loc im-

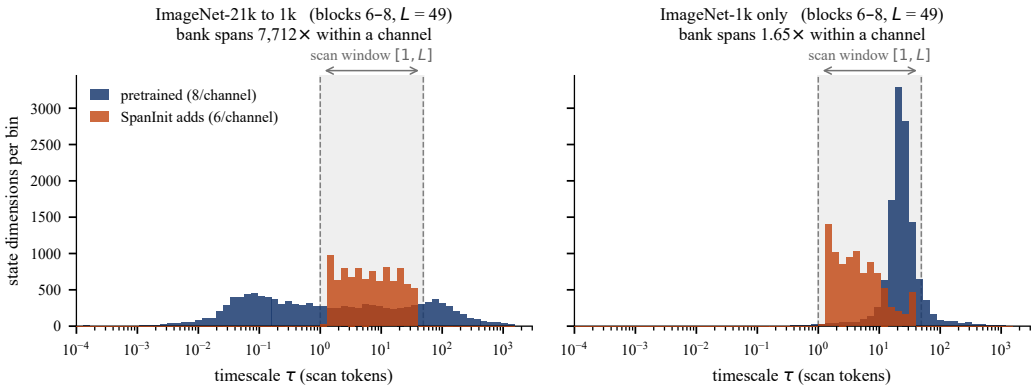


Figure 6: Pretrained and SpanInit-added timescales in the three deeper 512-channel MambaVision-B blocks with $L = 49$, shown for Caltech101. The spread-versus-collapse pattern observed at $L = 196$ persists at the shorter scan length. Within-channel bank spans are determined by A and therefore do not depend on $\bar{\Delta}$.

proves on both checkpoints ($+3.91$ and $+3.95$), so the sign reversal is a shift in the aggregate rather than a uniform flip across tasks.

Table 12: VTAB-1k, MambaVision-B pretrained on ImageNet-21k and on ImageNet-1k alone, the two checkpoints of Table 4a, with and without a surviving timescale bank. Cells are means over three seeds with the sample standard deviation; Δ and * are as in Table 3.

Natural

Method	CIFAR-100	Caltech101	DTD	Flowers102	Pets	SVHN	Sun397	Mean
21k Add.-scan	62.21 ± 0.41	89.67 ± 0.30	75.04 ± 0.16	99.34 ± 0.04	92.87 ± 0.18	73.25 ± 0.60	53.01 ± 0.14	77.91
21k SpanInit	63.31 ± 0.56	91.27 ± 0.06	74.61 ± 0.40	99.31 ± 0.08	92.99 ± 0.23	75.90 ± 0.38	53.13 ± 0.36	78.64
Δ *(21k)*	$+1.10^*$	$+1.59^*$	-0.43	-0.04	$+0.12$	$+2.64^*$	$+0.12$	$+0.73$
1k Add.-scan	53.44 ± 0.44	89.43 ± 0.34	69.91 ± 0.51	90.93 ± 0.33	92.28 ± 0.24	80.86 ± 0.42	43.61 ± 0.24	74.35
1k SpanInit	53.01 ± 0.27	89.43 ± 0.34	69.15 ± 0.28	88.01 ± 0.23	91.73 ± 0.02	78.37 ± 1.80	41.91 ± 0.30	73.09
Δ *(1k)*	-0.43	-0.01	-0.76	-2.92^*	-0.55^*	-2.49	-1.70^*	-1.26

Specialized

Method	Camelyon	EuroSAT	Resisc45	Retinopathy	Mean
21k Add.-scan	83.80 ± 0.06	93.33 ± 0.20	84.38 ± 0.25	71.41 ± 0.17	83.23
21k SpanInit	81.82 ± 0.84	93.99 ± 0.23	84.32 ± 0.10	70.65 ± 0.41	82.70
Δ *(21k)*	-1.98^*	$+0.66^*$	-0.06	-0.76^*	-0.53
1k Add.-scan	83.19 ± 0.49	92.30 ± 0.14	79.69 ± 0.41	71.53 ± 0.31	81.68
1k SpanInit	82.02 ± 0.79	92.54 ± 0.10	77.88 ± 0.24	71.11 ± 0.44	80.89
Δ *(1k)*	-1.17	$+0.24^*$	-1.81^*	-0.42	-0.79

Structured

Method	Clevr-Count	Clevr-Dist	DMLab	KITTI-Dist	dSpr-Loc	dSpr-Ori	sNORB-Azim	sNORB-Elev	Mean
21k Add.-scan	76.88 ± 0.32	61.35 ± 0.33	48.86 ± 0.38	78.67 ± 0.77	67.37 ± 1.52	53.65 ± 0.37	24.43 ± 0.43	32.92 ± 0.66	55.52
21k SpanInit	76.45 ± 0.28	61.86 ± 0.80	48.51 ± 0.40	80.08 ± 0.92	71.27 ± 2.03	53.64 ± 0.17	24.41 ± 0.55	32.34 ± 0.55	56.07
Δ *(21k)*	-0.43	$+0.52$	-0.35	$+1.41$	$+3.91^*$	-0.02	-0.01	-0.58	$+0.56$
1k Add.-scan	76.24 ± 0.23	60.43 ± 0.64	45.03 ± 0.15	76.79 ± 1.41	76.22 ± 2.09	53.23 ± 1.34	24.64 ± 0.07	32.98 ± 0.59	55.70
1k SpanInit	73.64 ± 1.78	61.69 ± 0.41	43.52 ± 0.95	79.32 ± 0.98	80.18 ± 1.10	51.91 ± 0.21	25.93 ± 0.77	34.11 ± 0.87	56.29
Δ *(1k)*	-2.60	$+1.26^*$	-1.51	$+2.53^*$	$+3.95^*$	-1.32	$+1.29^*$	$+1.13$	$+0.59$

	21k Add.-scan	21k SpanInit	Δ *(21k)*	1k Add.-scan	1k SpanInit	Δ *(1k)*
Mean (19 tasks)	**69.60**	**69.99**	**+0.39**	**68.04**	**67.66**	**−0.38**

C.2 PER-TASK RESULTS ON MAMBAVISION-L

Table 13 expands Table 4(f) to all nineteen tasks and adds the ImageNet-1k-only MambaVision-L beside it. The sign contrast seen for MambaVision-B recurs at the larger size: SpanInit raises the nineteen-task mean from 70.21 to 70.80 (+0.59) on the ImageNet-21k-pretrained checkpoint—group means +0.97 (Natural), +0.02 (Specialized), +0.54 (Structured)—but lowers it from 68.30 to 68.09 (−0.21) on the ImageNet-1k-only one, where all three group means lie near zero (−0.43, −0.19, −0.02). The negative aggregate on the collapsed checkpoint is thus reproduced, though its magnitude does not track model size in a simple way.

Table 13: VTAB-1k, MambaVision-L pretrained on ImageNet-21k and on ImageNet-1k alone. Cells are means over three seeds with the sample standard deviation; Δ and * are as in Table 3.

Natural

Method	CIFAR-100	Caltech101	DTD	Flowers102	Pets	SVHN	Sun397	Mean
21k Add.-scan	64.56 ± 0.16	90.37 ± 0.48	75.66 ± 0.38	99.40 ± 0.04	93.57 ± 0.24	75.75 ± 0.68	53.12 ± 0.54	78.92
21k SpanInit	67.12 ± 0.42	90.99 ± 0.20	75.53 ± 0.51	99.50 ± 0.03	93.68 ± 0.05	77.58 ± 0.54	54.80 ± 0.15	79.89
Δ *(21k)*	+2.57*	+0.62	−0.12	+0.09*	+0.11	+1.83*	+1.67*	+0.97
1k Add.-scan	53.30 ± 0.33	89.94 ± 0.77	68.92 ± 0.65	92.23 ± 0.13	92.38 ± 0.60	79.72 ± 0.37	43.16 ± 0.25	74.23
1k SpanInit	53.25 ± 0.46	90.44 ± 0.23	67.04 ± 0.52	90.74 ± 0.52	92.89 ± 0.17	80.52 ± 0.77	41.78 ± 0.19	73.81
Δ *(1k)*	−0.05	+0.51	−1.88*	−1.50*	+0.51	+0.80	−1.38*	−0.43

Specialized

Method	Camelyon	EuroSAT	Resisc45	Retinopathy	Mean
21k Add.-scan	83.74 ± 0.16	93.93 ± 0.13	85.40 ± 0.13	69.98 ± 1.31	83.26
21k SpanInit	83.15 ± 0.15	94.57 ± 0.16	85.48 ± 0.14	69.91 ± 0.61	83.28
Δ *(21k)*	−0.59*	+0.65*	+0.08	−0.06	+0.02
1k Add.-scan	82.00 ± 0.34	92.29 ± 0.26	80.38 ± 0.36	70.37 ± 0.30	81.26
1k SpanInit	82.56 ± 0.06	92.39 ± 0.12	79.00 ± 0.40	70.35 ± 0.79	81.07
Δ *(1k)*	+0.55	+0.10	−1.38*	−0.02	−0.19

Structured

Method	Clevr-Count	Clevr-Dist	DMLab	KITTI-Dist	dSpr-Loc	dSpr-Ori	sNORB-Azim	sNORB-Elev	Mean
21k Add.-scan	75.24 ± 1.43	59.95 ± 0.23	50.59 ± 0.43	78.39 ± 1.60	70.61 ± 0.65	55.46 ± 0.07	25.47 ± 0.28	32.89 ± 0.14	56.07
21k SpanInit	74.16 ± 0.91	58.21 ± 1.57	50.22 ± 0.33	80.54 ± 0.43	76.77 ± 0.16	54.26 ± 0.26	25.07 ± 0.38	33.64 ± 0.81	56.61
Δ *(21k)*	−1.08	−1.74	−0.37	+2.16	+6.16*	−1.20*	−0.40	+0.74	+0.54
1k Add.-scan	78.12 ± 1.43	62.07 ± 0.57	46.90 ± 0.32	80.26 ± 0.77	77.84 ± 1.02	53.46 ± 0.22	23.99 ± 0.24	30.30 ± 0.50	56.62
1k SpanInit	74.26 ± 0.75	62.38 ± 0.58	45.19 ± 1.01	81.62 ± 2.18	78.84 ± 1.18	52.40 ± 0.33	25.90 ± 0.42	32.19 ± 1.41	56.60
Δ *(1k)*	−3.86*	+0.31	−1.71*	+1.36	+1.01	−1.06*	+1.91*	+1.88	−0.02

	21k Add.-scan	21k SpanInit	Δ *(21k)*	1k Add.-scan	1k SpanInit	Δ *(1k)*
Mean (19 tasks)	**70.21**	**70.80**	**+0.59**	**68.30**	**68.09**	**−0.21**

C.3 BOTH CEILINGS OF THE SCAN WINDOW

Table 14 expands Table 4(c) to all nineteen tasks, comparing SpanInit's default ceiling L with the alternative ceiling $\max \tau$ (each channel's longest pretrained timescale). The two give almost the same nineteen-task mean (+0.39 for $[1, L]$ versus +0.37 for $[1, \max \tau]$) but redistribute it across groups: $\max \tau$ helps Structured more (+0.85 versus +0.56) and Natural less (+0.26 versus +0.73). The aggregate result is therefore insensitive to this boundary, and we keep L as an architecture-defined horizon.

C.4 THE NUMBER OF ADDED DIMENSIONS

Table 15 gives the nineteen per-task Δ behind Table 4(d) for every N' from 1 to 8. Reducing the expansion from six to three added dimensions keeps the aggregate positive—the nineteen-task mean rises by +0.55 at $N' = 3$ (from 68.94 to 69.49) against +0.39 at $N' = 6$—though the group pattern shifts toward Structured (+1.01 versus +0.56). Four tasks reach $p < 0.05$ at $N' = 3$: CIFAR-100, Caltech101, and dSprites-Loc for SpanInit, Patch-Camelyon against it. Without dSprites-Loc the means are +0.15 at $N' = 3$ and +0.20 at $N' = 6$. Applied to the seed-paired Δ values of the two settings, the same test separates them on only two tasks: SVHN in favor of $N' = 6$ and dSprites-Loc in favor of $N' = 3$.

Table 14: VTAB-1k, MambaVision-B (ImageNet-21k), with SpanInit under both ceilings of the scan window: the scan length L and each channel's longest pretrained timescale $\max \tau$. Cells are means over three seeds with the sample standard deviation; Δ and * are as in Table 3.

Natural

Method	CIFAR-100	Caltech101	DTD	Flowers102	Pets	SVHN	Sun397	Mean
Additional-scan	62.21 ± 0.41	89.67 ± 0.30	75.04 ± 0.16	99.34 ± 0.04	92.87 ± 0.18	73.25 ± 0.60	53.01 ± 0.14	77.91
SpanInit $[1, L]$	63.31 ± 0.56	91.27 ± 0.06	74.61 ± 0.40	99.31 ± 0.08	92.99 ± 0.23	75.90 ± 0.38	53.13 ± 0.36	78.64
Δ *(L)*	$+1.10^*$	$+1.59^*$	-0.43	-0.04	$+0.12$	$+2.64^*$	$+0.12$	$+0.73$
SpanInit $[1, \max \tau]$	62.38 ± 0.10	89.74 ± 0.55	74.98 ± 0.31	99.29 ± 0.12	92.75 ± 0.21	74.99 ± 0.24	53.11 ± 0.20	78.18
Δ *(max τ)*	$+0.17$	$+0.07$	-0.05	-0.05	-0.12	$+1.74^*$	$+0.10$	$+0.26$

Specialized

Method	Camelyon	EuroSAT	Resisc45	Retinopathy	Mean
Additional-scan	83.80 ± 0.06	93.33 ± 0.20	84.38 ± 0.25	71.41 ± 0.17	83.23
SpanInit $[1, L]$	81.82 ± 0.84	93.99 ± 0.23	84.32 ± 0.10	70.65 ± 0.41	82.70
Δ *(L)*	-1.98^*	$+0.66^*$	-0.06	-0.76^*	-0.53
SpanInit $[1, \max \tau]$	82.85 ± 0.19	93.57 ± 0.19	83.77 ± 0.11	71.01 ± 0.62	82.80
Δ *(max τ)*	-0.96^*	$+0.24$	-0.60^*	-0.40	-0.43

Structured

Method	Clevr-Count	Clevr-Dist	DMLab	KITTI-Dist	dSpr-Loc	dSpr-Ori	sNORB-Azim	sNORB-Elev	Mean
Additional-scan	76.88 ± 0.32	61.35 ± 0.33	48.86 ± 0.38	78.67 ± 0.77	67.37 ± 1.52	53.65 ± 0.37	24.43 ± 0.43	32.92 ± 0.66	55.52
SpanInit $[1, L]$	76.45 ± 0.28	61.86 ± 0.80	48.51 ± 0.40	80.08 ± 0.92	71.27 ± 2.03	53.64 ± 0.17	24.41 ± 0.55	32.34 ± 0.55	56.07
Δ *(L)*	-0.43	$+0.52$	-0.35	$+1.41$	$+3.91^*$	-0.02	-0.01	-0.58	$+0.56$
SpanInit $[1, \max \tau]$	78.67 ± 0.41	62.24 ± 0.49	49.61 ± 0.08	78.39 ± 0.35	70.62 ± 1.95	53.67 ± 0.26	24.99 ± 0.66	32.75 ± 0.15	56.37
Δ *(max τ)*	$+1.78^*$	$+0.90^*$	$+0.75^*$	-0.28	$+3.25^*$	$+0.02$	$+0.57$	-0.17	$+0.85$

	Additional-scan	SpanInit $[1, L]$	Δ *(L)*	SpanInit $[1, \max \tau]$	Δ *(max τ)*
Mean (19 tasks)	**69.60**	**69.99**	**+0.39**	**69.97**	**+0.37**

C.5 The placement rule

Table 16 expands Table 4(b) to all nineteen tasks, placing uniform log-spacing beside SpanInit. Both improve over copying on the nineteen-task mean—$+0.30$ for uniform and $+0.39$ for SpanInit—so most of the gain comes from introducing distinct decay scales rather than from conditioning on the pretrained anchors. SpanInit's advantage over uniform is small and concentrated in the Structured group ($+0.56$ versus $+0.36$; e.g., dSprites-Loc $+3.91$ versus $+2.86$), while the Natural and Specialized groups are nearly tied ($+0.73$ versus $+0.66$ and -0.53 versus -0.45).

C.6 The original S4D initialization

Table 17 gives the nineteen per-task values behind the S4D column of Table 4(e), with SpanInit beside it. Continuing the S4D-Real spectrum to $|A| = 9, \ldots, 14$ lowers the nineteen-task mean from 69.60 to 66.13 (-3.47), with group means of -1.37 (Natural), -0.93 (Specialized) and -6.58 (Structured). The losses concentrate on tasks that require integrating across tokens: Clevr-Count (-15.13), dSprites-Loc (-12.14), SVHN (-8.81) and Clevr-Dist (-8.73); the only gains above half a point are Sun397 ($+0.87$) and Retinopathy ($+0.40$). Thirteen tasks separate from Additional-scan under the test of Table 3, eleven of them against the spectrum. Under the step measured on the same images, it places the added dimensions at a median τ of 0.3–0.7 scan tokens against 2.7–3.8 for copying, so between 64 and 78% of them decay within a single scan token.

C.7 Vim pretrained on ImageNet-1k

Table 18 gives the nineteen per-task values on the two sizes of Vim (Zhu et al., 2024), whose banks are collapsed (Table 1). Vim-S is the backbone Yoshimura et al. (2025) report Additional-scan on.

On Vim-B SpanInit moves the nineteen-task mean from 66.74 to 66.92 ($+0.17$), with group means of $+0.00$ (Natural), -0.14 (Specialized) and $+0.48$ (Structured). Under the pooled model of Appendix A.2 the interval covers zero ($[-0.31, +0.65]$, $p = 0.47$), and the three seed means range from -0.19 to $+0.40$. Five tasks separate from Additional-scan under the test of Table 3, four of

Table 15: VTAB-1k, MambaVision-B (ImageNet-21k), with N' from 1 to 8 added dimensions. In the three task blocks each cell is a seed-paired Δ, marked as in Table 3. The last block gives, for each N', the parameters trained besides the head with their percentage of the frozen backbone, and the nineteen-task means; its $N' = 6$ row is that of Table 3.

Natural

Δ	CIFAR-100	Caltech101	DTD	Flowers102	Pets	SVHN	Sun397	Mean
$N'=1$	+0.37	+1.57*	−0.35	+0.02	+0.34*	+1.33	+0.60*	*+0.55*
$N'=2$	+1.04*	+1.59*	+0.02	+0.04	−0.13	−0.42	+0.08	*+0.32*
$N'=3$	+0.83*	+1.94*	−0.02	+0.03	+0.05	+1.09	+0.04	*+0.57*
$N'=4$	+0.37	+1.85*	−0.55*	+0.03	+0.22*	+1.49	+0.09	*+0.50*
$N'=5$	+1.25*	+1.86*	−0.37	+0.05	+0.24	+2.34*	−0.18	*+0.74*
$N'=6$	+1.10*	+1.59*	−0.43	−0.04	+0.12	+2.64*	+0.12	*+0.73*
$N'=7$	+0.90	+1.55*	−0.37	+0.11	+0.12	+2.73*	−0.05	*+0.71*
$N'=8$	+0.82*	+1.39*	−0.28	+0.09*	+0.05	+2.18*	+0.10	*+0.62*

Specialized

Δ	Camelyon	EuroSAT	Resisc45	Retinopathy	Mean
$N'=1$	−1.99*	+0.28*	+0.13	−0.32	*−0.47*
$N'=2$	−2.23*	+0.50*	+0.03	−0.57	*−0.57*
$N'=3$	−1.61*	+0.37	+0.21	−0.47	*−0.37*
$N'=4$	−1.89*	+0.53*	+0.37	−0.90*	*−0.47*
$N'=5$	−1.89*	+0.64*	+0.22	−1.24*	*−0.57*
$N'=6$	−1.98*	+0.66*	−0.06	−0.76*	*−0.53*
$N'=7$	−1.43*	+0.51	−0.03	−1.26*	*−0.56*
$N'=8$	−1.56*	+0.40	−0.03	−0.76*	*−0.49*

Structured

Δ	Clevr-Count	Clevr-Dist	DMLab	KITTI-Dist	dSpr-Loc	dSpr-Ori	sNORB-Azim	sNORB-Elev	Mean
$N'=1$	−0.13	−0.83	−0.20	+1.03	+4.11	−1.79*	−0.18	+0.91*	*+0.36*
$N'=2$	−1.21	+0.67*	+0.11	+1.83	+7.92*	−0.06	+0.12	−1.70*	*+0.96*
$N'=3$	−0.76	+0.12	−0.70	+1.45	+7.90*	−0.33	+0.30	+0.09	*+1.01*
$N'=4$	−0.67	+0.08	−0.31	+1.27	+7.39*	+0.05	+0.46	−0.11	*+1.02*
$N'=5$	−0.50	+0.61*	−0.11	+1.31	+8.58*	+0.23	−0.01	−0.68*	*+1.18*
$N'=6$	−0.43	+0.52	−0.35	+1.41	+3.91*	−0.02	−0.01	−0.58	*+0.56*
$N'=7$	+0.11	+0.78*	+0.25	+1.31	+4.55*	+0.01	+0.52	−0.29	*+0.91*
$N'=8$	+0.54	+0.42	−0.38	+0.19	+3.20*	−0.01	+0.19	−0.33	*+0.48*

Mean (19 tasks)	Params	Additional-scan	SpanInit	Δ
$N'=1$	8.4K (0.01%)	67.20	67.46	+0.26
$N'=2$	16.9K (0.02%)	68.42	68.82	+0.40
$N'=3$	25.3K (0.03%)	68.94	69.49	+0.55
$N'=4$	33.8K (0.03%)	69.23	69.75	+0.51
$N'=5$	42.2K (0.04%)	69.38	70.03	+0.65
$N'=6$	50.7K (0.05%)	69.60	69.99	+0.39
$N'=7$	59.1K (0.06%)	69.61	70.13	+0.53
$N'=8$	67.6K (0.07%)	69.78	70.11	+0.33

them in favour of SpanInit: Clevr-Dist (+4.66), KITTI-Dist (+2.53), SVHN (+1.86) and Clevr-Count (+1.80), against Oxford-Flowers102 (−1.45). Vim enters fine-tuning with every state within a factor of two of $|A| = 1$ (Table 1); the aggregate effect on this bank is not distinguishable from zero, rather than negative as on the ImageNet-1k-only MambaVision-B (−0.38). The expansion is also much larger here: Vim runs an SSM in both directions of all 24 layers where MambaVision-B has eight SSM blocks, so the same $N' = 6$ adds 1,327,104 parameters instead of 50,688, twenty-six times as many. The two frozen backbones are nearly the same size, 96.8M and 96.7M, so the added parameters are 1.37% of the backbone here against 0.05% there. The larger expansion does not produce a larger aggregate effect.

On Vim-S the linear probe reaches 52.72 over the nineteen tasks, within 0.03 of the 52.75 that Yoshimura et al. (2025) report for the same backbone and protocol. SpanInit moves the nineteen-task mean from 67.74 to 67.49 (−0.25), with group means of −0.37 (Natural), −0.58 (Specialized) and +0.02 (Structured); the pooled interval covers zero ($[-0.59, +0.09]$, $p = 0.15$). Six tasks separate from Additional-scan under the test of Table 3, five of them against SpanInit, the exception being dSprites-Loc (+3.09), which also gains on Vim-B and on both MambaVision checkpoints.

Table 16: VTAB-1k, MambaVision-B (ImageNet-21k), with the added timescales spaced uniformly in $\log \tau$ over $[1, L]$, ignoring the pretrained bank. Cells are means over three seeds with the sample standard deviation; Δ and * are as in Table 3.

Natural

Method	CIFAR-100	Caltech101	DTD	Flowers102	Pets	SVHN	Sun397	Mean
Add.-scan	62.21 ± 0.41	89.67 ± 0.30	75.04 ± 0.16	99.34 ± 0.04	92.87 ± 0.18	73.25 ± 0.60	53.01 ± 0.14	77.91
SpanInit	63.31 ± 0.56	91.27 ± 0.06	74.61 ± 0.40	99.31 ± 0.08	92.99 ± 0.23	75.90 ± 0.38	53.13 ± 0.36	78.64
Δ *SpanInit*	$+1.10^*$	$+1.59^*$	-0.43	-0.04	$+0.12$	$+2.64^*$	$+0.12$	$+0.73$
uniform	63.18 ± 0.44	91.01 ± 0.09	74.77 ± 0.19	99.36 ± 0.03	92.98 ± 0.09	75.61 ± 0.39	53.09 ± 0.13	78.57
Δ *uniform*	$+0.97^*$	$+1.34^*$	-0.27	$+0.02$	$+0.11$	$+2.36^*$	$+0.08$	$+0.66$

Specialized

Method	Camelyon	EuroSAT	Resisc45	Retinopathy	Mean
Add.-scan	83.80 ± 0.06	93.33 ± 0.20	84.38 ± 0.25	71.41 ± 0.17	83.23
SpanInit	81.82 ± 0.84	93.99 ± 0.23	84.32 ± 0.10	70.65 ± 0.41	82.70
Δ *SpanInit*	-1.98^*	$+0.66^*$	-0.06	-0.76^*	-0.53
uniform	82.08 ± 0.42	94.08 ± 0.03	84.63 ± 0.24	70.33 ± 0.63	82.78
Δ *uniform*	-1.72^*	$+0.75^*$	$+0.26$	-1.08^*	-0.45

Structured

Method	Clevr-Count	Clevr-Dist	DMLab	KITTI-Dist	dSpr-Loc	dSpr-Ori	sNORB-Azim	sNORB-Elev	Mean
Add.-scan	76.88 ± 0.32	61.35 ± 0.33	48.86 ± 0.38	78.67 ± 0.77	67.37 ± 1.52	53.65 ± 0.37	24.43 ± 0.43	32.92 ± 0.66	55.52
SpanInit	76.45 ± 0.28	61.86 ± 0.80	48.51 ± 0.40	80.08 ± 0.92	71.27 ± 2.03	53.64 ± 0.17	24.41 ± 0.55	32.34 ± 0.55	56.07
Δ *SpanInit*	-0.43	$+0.52$	-0.35	$+1.41$	$+3.91^*$	-0.02	-0.01	-0.58	$+0.56$
uniform	76.84 ± 0.53	60.87 ± 0.39	48.97 ± 0.44	78.81 ± 0.57	70.23 ± 0.88	53.61 ± 0.51	24.73 ± 0.79	32.97 ± 0.33	55.88
Δ *uniform*	-0.04	-0.48	$+0.11$	$+0.14$	$+2.86^*$	-0.05	$+0.30$	$+0.06$	$+0.36$

	Add.-scan	SpanInit	Δ *SpanInit*	uniform	Δ *uniform*
Mean (19 tasks)	**69.60**	**69.99**	**+0.39**	**69.90**	**+0.30**

Table 17: VTAB-1k, MambaVision-B (ImageNet-21k), with the added A continuing the S4D-Real spectrum $A_n = -(n+1)$ of the pretrained states, $|A| = 9, \ldots, 14$. Cells are means over three seeds with the sample standard deviation; Δ and * are as in Table 3.

Natural

Method	CIFAR-100	Caltech101	DTD	Flowers102	Pets	SVHN	Sun397	Mean
Add.-scan	62.21 ± 0.41	89.67 ± 0.30	75.04 ± 0.16	99.34 ± 0.04	92.87 ± 0.18	73.25 ± 0.60	53.01 ± 0.14	77.91
SpanInit	63.31 ± 0.56	91.27 ± 0.06	74.61 ± 0.40	99.31 ± 0.08	92.99 ± 0.23	75.90 ± 0.38	53.13 ± 0.36	78.64
Δ *SpanInit*	$+1.10^*$	$+1.59^*$	-0.43	-0.04	$+0.12$	$+2.64^*$	$+0.12$	$+0.73$
S4D	60.97 ± 0.46	89.78 ± 0.19	74.34 ± 0.51	99.35 ± 0.07	93.02 ± 0.16	64.44 ± 0.74	53.88 ± 0.12	76.54
Δ *S4D*	-1.23^*	$+0.11$	-0.69	$+0.01$	$+0.15$	-8.81^*	$+0.87^*$	-1.37

Specialized

Method	Camelyon	EuroSAT	Resisc45	Retinopathy	Mean
Add.-scan	83.80 ± 0.06	93.33 ± 0.20	84.38 ± 0.25	71.41 ± 0.17	83.23
SpanInit	81.82 ± 0.84	93.99 ± 0.23	84.32 ± 0.10	70.65 ± 0.41	82.70
Δ *SpanInit*	-1.98^*	$+0.66^*$	-0.06	-0.76^*	-0.53
S4D	81.15 ± 0.16	93.34 ± 0.13	82.89 ± 0.19	71.81 ± 0.20	82.30
Δ *S4D*	-2.65^*	$+0.01$	-1.48^*	$+0.40^*$	-0.93

Structured

Method	Clevr-Count	Clevr-Dist	DMLab	KITTI-Dist	dSpr-Loc	dSpr-Ori	sNORB-Azim	sNORB-Elev	Mean
Add.-scan	76.88 ± 0.32	61.35 ± 0.33	48.86 ± 0.38	78.67 ± 0.77	67.37 ± 1.52	53.65 ± 0.37	24.43 ± 0.43	32.92 ± 0.66	55.52
SpanInit	76.45 ± 0.28	61.86 ± 0.80	48.51 ± 0.40	80.08 ± 0.92	71.27 ± 2.03	53.64 ± 0.17	24.41 ± 0.55	32.34 ± 0.55	56.07
Δ *SpanInit*	-0.43	$+0.52$	-0.35	$+1.41$	$+3.91^*$	-0.02	-0.01	-0.58	$+0.56$
S4D	61.75 ± 0.27	52.62 ± 0.35	45.60 ± 0.28	77.68 ± 0.92	55.23 ± 1.16	48.96 ± 0.21	21.04 ± 0.85	28.58 ± 0.30	48.93
Δ *S4D*	-15.13^*	-8.73^*	-3.26^*	-0.98	-12.14^*	-4.69^*	-3.39^*	-4.34^*	-6.58

	Add.-scan	SpanInit	Δ *SpanInit*	S4D	Δ *S4D*
Mean (19 tasks)	**69.60**	**69.99**	**+0.39**	**66.13**	**−3.47**

Additional-scan reaches 67.74 against the 68.65 that Yoshimura et al. (2025) report for this backbone.

Table 18: VTAB-1k, Vim-B and Vim-S (ImageNet-1k), under the Additional-scan configuration that Yoshimura et al. (2025) release for each: $N' = 6$ in both directions of all 24 layers, with $L = 197$. Params counts what Additional-scan and SpanInit each train besides the head, with its percentage of the frozen backbone. Cells are means over three seeds with the sample standard deviation; Δ and * are as in Table 3.

Natural

	Method	CIFAR-100	Caltech101	DTD	Flowers102	Pets	SVHN	Sun397	Mean
Vim-B	Linear	46.00 ± 0.09	87.97 ± 0.06	62.59 ± 0.63	78.40 ± 0.09	89.92 ± 0.17	42.26 ± 0.08	40.43 ± 0.03	63.94
	Additional-scan	56.21 ± 0.63	90.19 ± 0.31	66.74 ± 0.45	86.73 ± 0.54	90.62 ± 0.70	82.31 ± 0.51	41.28 ± 0.24	73.44
	SpanInit	56.34 ± 0.34	90.10 ± 0.03	66.35 ± 0.24	85.28 ± 0.17	90.29 ± 0.56	84.16 ± 0.40	41.58 ± 0.48	73.44
	Δ	+0.13	−0.09	−0.39	−1.45*	−0.34	+1.86*	+0.29	*+0.00*
Vim-S	Linear	47.97 ± 0.02	86.49 ± 0.05	58.92 ± 0.17	75.28 ± 0.20	88.74 ± 0.19	41.08 ± 0.18	39.45 ± 0.11	62.56
	Additional-scan	57.88 ± 0.15	90.17 ± 0.56	66.29 ± 0.51	87.59 ± 0.31	90.14 ± 0.52	83.14 ± 0.93	41.96 ± 0.27	73.88
	SpanInit	57.26 ± 0.68	90.29 ± 0.20	65.25 ± 0.13	85.94 ± 0.16	89.82 ± 0.22	84.55 ± 0.58	41.44 ± 0.18	73.51
	Δ	−0.62	+0.13	−1.05*	−1.65*	−0.33	+1.42	−0.52*	*−0.37*

Specialized

	Method	Camelyon	EuroSAT	Resisc45	Retinopathy	Mean
Vim-B	Linear	81.54 ± 0.13	89.48 ± 0.07	71.72 ± 0.14	72.69 ± 0.11	78.86
	Additional-scan	82.39 ± 0.87	92.95 ± 0.19	79.07 ± 0.45	70.99 ± 0.80	81.35
	SpanInit	82.55 ± 0.66	92.76 ± 0.48	78.54 ± 0.30	70.99 ± 0.55	81.21
	Δ	+0.16	−0.19	−0.53	+0.00	*−0.14*
Vim-S	Linear	78.30 ± 0.01	88.67 ± 0.20	68.49 ± 0.20	73.12 ± 0.09	77.15
	Additional-scan	82.12 ± 0.28	93.41 ± 0.26	79.74 ± 0.28	71.36 ± 0.40	81.66
	SpanInit	81.74 ± 0.65	92.98 ± 0.21	78.58 ± 0.43	71.01 ± 0.38	81.08
	Δ	−0.39	−0.43*	−1.16*	−0.34	*−0.58*

Structured

	Method	Clevr-Count	Clevr-Dist	DMLab	KITTI-Dist	dSpr-Loc	dSpr-Ori	sNORB-Azim	sNORB-Elev	Mean
Vim-B	Linear	37.15 ± 0.09	38.32 ± 0.12	34.78 ± 0.16	60.53 ± 0.57	18.86 ± 0.01	27.82 ± 0.04	16.60 ± 0.23	23.26 ± 0.20	32.17
	Additional-scan	70.06 ± 0.67	58.75 ± 1.65	43.10 ± 0.32	76.47 ± 0.63	73.37 ± 3.29	44.78 ± 0.32	28.12 ± 0.92	34.04 ± 0.93	53.58
	SpanInit	71.85 ± 0.64	63.41 ± 0.90	43.37 ± 0.45	79.00 ± 0.72	74.79 ± 1.69	43.73 ± 0.70	23.70 ± 6.55	32.63 ± 1.57	54.06
	Δ	+1.80*	+4.66*	+0.27	+2.53*	+1.43	−1.05	−4.41	−1.41	*+0.48*
Vim-S	Linear	37.07 ± 0.23	36.31 ± 0.09	33.54 ± 0.03	65.73 ± 0.59	17.57 ± 0.11	24.54 ± 0.12	15.71 ± 0.05	24.74 ± 0.13	31.90
	Additional-scan	70.41 ± 1.30	59.80 ± 2.15	42.38 ± 0.37	80.40 ± 0.94	74.55 ± 0.67	47.70 ± 0.56	27.53 ± 0.75	40.55 ± 4.11	55.41
	SpanInit	67.40 ± 3.33	61.33 ± 2.31	42.28 ± 0.35	78.86 ± 1.20	77.64 ± 1.03	48.45 ± 0.94	27.74 ± 0.34	39.79 ± 2.18	55.44
	Δ	−3.01	+1.54	−0.10	−1.55	+3.09*	+0.75	+0.22	−0.77	*+0.02*

Mean (19 tasks)	Linear	Additional-scan	SpanInit	Δ	Params
Vim-B	**53.70**	**66.74**	**66.92**	**+0.17**	**1,327,104 (1.37%)**
Vim-S	**52.72**	**67.74**	**67.49**	**−0.25**	**663,552 (2.61%)**